## Contents

### About script steps
- Creating scripts to automate tasks .................................................. 7
- Using this script steps reference ......................................................... 7
- Learning about scripts ........................................................................ 7

### Script steps reference (alphabetical list)

#### Control script steps
- **Perform Script** ................................................................. 12
- **Pause/Resume Script** ............................................................. 14
- **Exit Script** ............................................................................. 15
- **Halt Script** .............................................................................. 16
- **If** .......................................................................................... 17
- **Else If** ................................................................................... 18
- **Else** ......................................................................................... 19
- **End If** ..................................................................................... 19
- **Loop** ......................................................................................... 20
- **Exit Loop If** ........................................................................... 21
- **End Loop** ................................................................................ 22
- **Allow User Abort** ..................................................................... 22
- **Set Error Capture** .................................................................... 23
- **Set Variable** ............................................................................ 24
- **Install OnTimer Script** ............................................................. 25

#### Navigation script steps
- **Go to Layout** .......................................................................... 27
- **Go to Record/Request/Page** ....................................................... 28
- **Go to Related Record** .............................................................. 29
- **Go to Portal Row** ....................................................................... 30
- **Go to Object** ............................................................................ 31
- **Go to Field** ............................................................................... 32
- **Go to Next Field** ........................................................................ 33
- **Go to Previous Field** ................................................................. 34
- **Enter Browse Mode** ................................................................. 34
- **Enter Find Mode** ........................................................................ 35
- **Enter Preview Mode** ................................................................. 36

#### Editing script steps
- **Undo/Redo** ............................................................................... 37
- **Cut** ............................................................................................ 37
- **Copy** .......................................................................................... 38
- **Paste** ........................................................................................... 39
- **Clear** ............................................................................................ 40
- **Set Selection** .............................................................................. 40
- **Select All** .................................................................................... 41
- **Perform Find/Replace** ............................................................... 42
### Fields script steps

- Set Field ................................................................. 43
- Set Field By Name ...................................................... 44
- Set Next Serial Value .................................................. 45
- Insert Text ................................................................. 46
- Insert Calculated Result ............................................... 47
- Insert From Index ....................................................... 48
- Insert From Last Visited .............................................. 49
- Insert Current Date .................................................... 50
- Insert Current Time ................................................... 51
- Insert Current User Name ............................................. 51
- Insert Picture ............................................................ 52
- Insert QuickTime ....................................................... 53
- Insert Object (Windows) .............................................. 54
- Insert File ................................................................. 55
- Update Link (Windows) ............................................... 55
- Replace Field Contents ............................................... 56
- Relookup Field Contents ............................................. 57
- Export Field Contents ................................................ 58

### Records script steps

- New Record/Request .................................................. 60
- Duplicate Record/Request .......................................... 60
- Delete Record/Request ............................................... 61
- Delete Portal Row ........................................................ 62
- Delete All Records .................................................... 63
- Open Record/Request .................................................. 63
- Revert Record/Request .............................................. 64
- Commit Records/Requests ........................................... 65
- Copy Record/Request .................................................. 66
- Copy All Records/Requests .......................................... 67
- Import Records .......................................................... 68
- Export Records .......................................................... 70
- Save Records As Excel ............................................... 72
- Save Records As PDF .................................................... 73

### Found Sets script steps

- Perform Find ............................................................ 75
- Constrain Found Set ................................................... 76
- Extend Found Set ........................................................ 77
- Modify Last Find ........................................................ 77
- Show All Records ........................................................ 78
- Show Omitted Only ..................................................... 79
- Omit Record ............................................................... 79
- Omit Multiple Records ............................................... 80
- Sort Records ............................................................. 81
- Unsort Records .......................................................... 82

### Windows script steps

- New Window ............................................................. 83
- Select Window ........................................................... 84
Close Window ................................................................. 85
Adjust Window ............................................................. 86
Move/Resize Window ......................................................... 87
Arrange All Windows ......................................................... 88
Freeze Window ............................................................ 89
Refresh Window .......................................................... 90
Scroll Window .............................................................. 90
Show/Hide Status Area ....................................................... 91
Show/Hide Text Ruler ......................................................... 92
Set Window Title ........................................................... 92
Set Zoom Level ............................................................ 93
View As ................................................................. 94

Files script steps

New File ........................................................................... 95
Open File ....................................................................... 96
Close File ...................................................................... 97
Convert File ................................................................... 97
Set Multi-User ............................................................ 99
Set Use System Formats .................................................. 100
Save a Copy as ............................................................. 101
Recover File ................................................................. 101
Print Setup ..................................................................... 102
Print ............................................................................. 103

Accounts script steps

Add Account ................................................................. 105
Delete Account .......................................................... 106
Reset Account Password .............................................. 107
Change Password ....................................................... 108
Enable Account .......................................................... 109
Re-Login ................................................................. 110

Spelling script steps

Check Selection ........................................................... 111
Check Record ............................................................... 111
Check Found Set .......................................................... 112
Correct Word ............................................................... 113
Spelling Options .......................................................... 114
Select Dictionaries ........................................................ 114
Edit User Dictionary ..................................................... 115

Open Menu Item script steps

Open Preferences .......................................................... 117
Open Edit Saved Finds ................................................... 118
Open File Options ........................................................ 119
Open Manage Database ............................................... 119
Open Manage Data Sources .......................................... 120
Open Manage Value Lists .............................................. 121
Open Find/Replace ....................................................... 122
Open Help . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 123
Open Remote . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 124
Open Manage Scripts . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 124
Open Sharing . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 125

**Miscellaneous script steps**

    Show Custom Dialog. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 127
    Allow Toolbars. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 129
    Beep . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 130
    Speak (Mac OS) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 131
    Dial Phone . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 132
    Install Menu Set . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 133
    Set Web Viewer . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 134
    Open URL . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 135
    Send Mail . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 137
    Send DDE Execute (Windows) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 138
    Perform AppleScript (Mac OS) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 139
    Execute SQL . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 141
    Send Event (Mac OS) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 142
    Send Event (Windows) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 144
    Comment . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 145
    Flush Cache to Disk . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 146
    Exit Application . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 146
About script steps

Creating scripts to automate tasks

Scripts can do simple tasks like setting print orientation or complex tasks like preparing a customized mailing to each client.

For example, you could define a complex set of scripts that creates a thank you letter to clients in your Client database who have made a purchase in the last week. The script composes a letter tailored to each client. The script switches to Preview mode and pauses so you can see what the letter looks like before it prints. The whole task is initiated by clicking a button on the Sales Entry layout.

You use the Manage Scripts feature in FileMaker Pro to build scripts by selecting from a list of FileMaker Pro commands, called script steps, specifying options (if necessary), and arranging the steps in the correct order to perform the task.

Using this script steps reference

The content in this document was originally written for the FileMaker Pro and FileMaker Pro Advanced Help. It has been collected in this format to allow solution developers to read the information independent of the help system. Links to help topics may not work in this format.

Script step topics are organized by category. Each script step topic describes what the script step does, options, and parameters. Each topic also shows the format for the script step and provides a usage example.

Learning about scripts

If you’re new to scripting, use the following resources to learn about scripts.

To use a step-by-step tutorial:

• In FileMaker Pro, choose Help menu > Product Documentation > Tutorial.

To view help topics about scripting and the Manage Scripts feature:

• In FileMaker Pro, choose Help menu > FileMaker Pro Help. In the help window, choose Designing and creating databases, then choose Creating scripts to automate tasks. You can also search help for information about scripting.

To view all help topics about individual script steps:

• In FileMaker Pro, choose Help > FileMaker Pro Help. In the help window, choose Reference > Script steps reference.

To view answers to frequently asked questions, tips, troubleshooting advice, and more, visit the FileMaker Knowledge Base at www.filemaker.com/kb/.
## Script steps reference (alphabetical list)

### A, B, C

<table>
<thead>
<tr>
<th>Script Step</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add Account</td>
<td>105</td>
</tr>
<tr>
<td>Adjust Window</td>
<td>86</td>
</tr>
<tr>
<td>Allow Toolbars</td>
<td>129</td>
</tr>
<tr>
<td>Allow User Abort</td>
<td>22</td>
</tr>
<tr>
<td>Arrange All Windows</td>
<td>88</td>
</tr>
<tr>
<td>Beep</td>
<td>130</td>
</tr>
<tr>
<td>Change Password</td>
<td>108</td>
</tr>
<tr>
<td>Check Found Set</td>
<td>112</td>
</tr>
<tr>
<td>Check Record</td>
<td>111</td>
</tr>
<tr>
<td>Check Selection</td>
<td>111</td>
</tr>
<tr>
<td>Clear</td>
<td>40</td>
</tr>
<tr>
<td>Close File</td>
<td>97</td>
</tr>
<tr>
<td>Close Window</td>
<td>85</td>
</tr>
<tr>
<td>Comment</td>
<td>145</td>
</tr>
<tr>
<td>Commit Records/Requests</td>
<td>65</td>
</tr>
<tr>
<td>Constrain Found Set</td>
<td>76</td>
</tr>
<tr>
<td>Convert File</td>
<td>97</td>
</tr>
<tr>
<td>Copy</td>
<td>38</td>
</tr>
<tr>
<td>Copy All Records/Requests</td>
<td>67</td>
</tr>
<tr>
<td>Copy Record/Request</td>
<td>66</td>
</tr>
<tr>
<td>Correct Word</td>
<td>113</td>
</tr>
<tr>
<td>Cut</td>
<td>37</td>
</tr>
</tbody>
</table>

### D

<table>
<thead>
<tr>
<th>Script Step</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete Account</td>
<td>106</td>
</tr>
<tr>
<td>Delete All Records</td>
<td>63</td>
</tr>
<tr>
<td>Delete Portal Row</td>
<td>62</td>
</tr>
<tr>
<td>Delete Record/Request</td>
<td>61</td>
</tr>
<tr>
<td>Dial Phone</td>
<td>132</td>
</tr>
<tr>
<td>Duplicate Record/Request</td>
<td>60</td>
</tr>
</tbody>
</table>

### E

<table>
<thead>
<tr>
<th>Script Step</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit User Dictionary</td>
<td>115</td>
</tr>
<tr>
<td>Else</td>
<td>19</td>
</tr>
<tr>
<td>Else If</td>
<td>18</td>
</tr>
<tr>
<td>Enable Account</td>
<td>109</td>
</tr>
<tr>
<td>End If</td>
<td>19</td>
</tr>
<tr>
<td>End Loop</td>
<td>22</td>
</tr>
<tr>
<td>Enter Browse Mode</td>
<td>34</td>
</tr>
<tr>
<td>Enter Find Mode</td>
<td>35</td>
</tr>
<tr>
<td>Enter Preview Mode</td>
<td>36</td>
</tr>
<tr>
<td>Execute SQL</td>
<td>141</td>
</tr>
<tr>
<td>Exit Application</td>
<td>146</td>
</tr>
<tr>
<td>Exit Loop If</td>
<td>21</td>
</tr>
<tr>
<td>Exit Script</td>
<td>15</td>
</tr>
<tr>
<td>Export Field Contents</td>
<td>58</td>
</tr>
<tr>
<td>Export Records</td>
<td>70</td>
</tr>
</tbody>
</table>
### Script steps reference (alphabetical list)

<table>
<thead>
<tr>
<th>Step</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extend Found Set</td>
<td>77</td>
</tr>
<tr>
<td><strong>F</strong></td>
<td></td>
</tr>
<tr>
<td>Flush Cache to Disk</td>
<td>146</td>
</tr>
<tr>
<td>Freeze Window</td>
<td>89</td>
</tr>
<tr>
<td><strong>G</strong></td>
<td></td>
</tr>
<tr>
<td>Go to Field</td>
<td>32</td>
</tr>
<tr>
<td>Go to Layout</td>
<td>27</td>
</tr>
<tr>
<td>Go to Next Field</td>
<td>33</td>
</tr>
<tr>
<td>Go to Object</td>
<td>31</td>
</tr>
<tr>
<td>Go to Portal Row</td>
<td>30</td>
</tr>
<tr>
<td>Go to Previous Field</td>
<td>34</td>
</tr>
<tr>
<td>Go to Record/Request/Page</td>
<td>28</td>
</tr>
<tr>
<td>Go to Related Record</td>
<td>29</td>
</tr>
<tr>
<td><strong>H, I, J, K</strong></td>
<td></td>
</tr>
<tr>
<td>Halt Script</td>
<td>16</td>
</tr>
<tr>
<td>If</td>
<td>17</td>
</tr>
<tr>
<td>Import Records</td>
<td>68</td>
</tr>
<tr>
<td>Insert Calculated Result</td>
<td>47</td>
</tr>
<tr>
<td>Insert Current Date</td>
<td>50</td>
</tr>
<tr>
<td>Insert Current Time</td>
<td>51</td>
</tr>
<tr>
<td>Insert Current User Name</td>
<td>51</td>
</tr>
<tr>
<td>Insert File</td>
<td>55</td>
</tr>
<tr>
<td>Insert From Index</td>
<td>48</td>
</tr>
<tr>
<td>Insert From Last Visited</td>
<td>49</td>
</tr>
<tr>
<td>Insert Object (Windows)</td>
<td>54</td>
</tr>
<tr>
<td>Insert Picture</td>
<td>52</td>
</tr>
<tr>
<td>Insert QuickTime</td>
<td>53</td>
</tr>
<tr>
<td>Insert Text</td>
<td>46</td>
</tr>
<tr>
<td>Install Menu Set</td>
<td>133</td>
</tr>
<tr>
<td>Install OnTimer Script</td>
<td>25</td>
</tr>
<tr>
<td><strong>L, M, N</strong></td>
<td></td>
</tr>
<tr>
<td>Loop</td>
<td>20</td>
</tr>
<tr>
<td>Modify Last Find</td>
<td>77</td>
</tr>
<tr>
<td>Move/Resize Window</td>
<td>87</td>
</tr>
<tr>
<td>New File</td>
<td>95</td>
</tr>
<tr>
<td>New Record/Request</td>
<td>60</td>
</tr>
<tr>
<td>New Window</td>
<td>83</td>
</tr>
<tr>
<td><strong>O</strong></td>
<td></td>
</tr>
<tr>
<td>Omit Multiple Records</td>
<td>80</td>
</tr>
<tr>
<td>Omit Record</td>
<td>79</td>
</tr>
<tr>
<td>Open Edit Saved Finds</td>
<td>118</td>
</tr>
<tr>
<td>Open File</td>
<td>96</td>
</tr>
<tr>
<td>Open File Options</td>
<td>119</td>
</tr>
<tr>
<td>Open Find/Replace</td>
<td>122</td>
</tr>
<tr>
<td>Open Help</td>
<td>123</td>
</tr>
</tbody>
</table>
## Script steps reference (alphabetical list)

<table>
<thead>
<tr>
<th>Action</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open Manage Data Sources</td>
<td>120</td>
</tr>
<tr>
<td>Open Manage Database</td>
<td>119</td>
</tr>
<tr>
<td>Open Manage Scripts</td>
<td>124</td>
</tr>
<tr>
<td>Open Manage Value Lists</td>
<td>121</td>
</tr>
<tr>
<td>Open Preferences</td>
<td>117</td>
</tr>
<tr>
<td>Open Record/Request</td>
<td>63</td>
</tr>
<tr>
<td>Open Remote</td>
<td>124</td>
</tr>
<tr>
<td>Open Sharing</td>
<td>125</td>
</tr>
<tr>
<td>Open URL</td>
<td>135</td>
</tr>
<tr>
<td>Paste</td>
<td>39</td>
</tr>
<tr>
<td>Pause/Resume Script</td>
<td>14</td>
</tr>
<tr>
<td>Perform AppleScript (Mac OS)</td>
<td>139</td>
</tr>
<tr>
<td>Perform Find</td>
<td>75</td>
</tr>
<tr>
<td>Perform Find/Replace</td>
<td>42</td>
</tr>
<tr>
<td>Perform Script</td>
<td>12</td>
</tr>
<tr>
<td>Print</td>
<td>103</td>
</tr>
<tr>
<td>Print Setup</td>
<td>102</td>
</tr>
<tr>
<td>Recover File</td>
<td>101</td>
</tr>
<tr>
<td>Refresh Window</td>
<td>90</td>
</tr>
<tr>
<td>Re-Login</td>
<td>110</td>
</tr>
<tr>
<td>Relookup Field Contents</td>
<td>57</td>
</tr>
<tr>
<td>Replace Field Contents</td>
<td>56</td>
</tr>
<tr>
<td>Reset Account Password</td>
<td>107</td>
</tr>
<tr>
<td>Revert Record/Request</td>
<td>64</td>
</tr>
<tr>
<td>Save a Copy as</td>
<td>101</td>
</tr>
<tr>
<td>Save Records As Excel</td>
<td>72</td>
</tr>
<tr>
<td>Save Records As PDF</td>
<td>73</td>
</tr>
<tr>
<td>Scroll Window</td>
<td>90</td>
</tr>
<tr>
<td>Select All</td>
<td>41</td>
</tr>
<tr>
<td>Select Dictionaries</td>
<td>114</td>
</tr>
<tr>
<td>Select Window</td>
<td>84</td>
</tr>
<tr>
<td>Send DDE Execute (Windows)</td>
<td>138</td>
</tr>
<tr>
<td>Send Event (Mac OS)</td>
<td>142</td>
</tr>
<tr>
<td>Send Event (Windows)</td>
<td>144</td>
</tr>
<tr>
<td>Send Mail</td>
<td>137</td>
</tr>
<tr>
<td>Set Error Capture</td>
<td>23</td>
</tr>
<tr>
<td>Set Field</td>
<td>43</td>
</tr>
<tr>
<td>Set Field By Name</td>
<td>44</td>
</tr>
<tr>
<td>Set Multi-User</td>
<td>99</td>
</tr>
<tr>
<td>Set Next Serial Value</td>
<td>45</td>
</tr>
<tr>
<td>Set Selection</td>
<td>40</td>
</tr>
<tr>
<td>Set Use System Formats</td>
<td>100</td>
</tr>
<tr>
<td>Set Variable</td>
<td>24</td>
</tr>
<tr>
<td>Set Web Viewer</td>
<td>134</td>
</tr>
<tr>
<td>Set Window Title</td>
<td>92</td>
</tr>
<tr>
<td>Script Steps</td>
<td>Page</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Set Zoom Level</td>
<td>93</td>
</tr>
<tr>
<td>Show All Records</td>
<td>78</td>
</tr>
<tr>
<td>Show Custom Dialog</td>
<td>127</td>
</tr>
<tr>
<td>Show Omitted Only</td>
<td>79</td>
</tr>
<tr>
<td>Show/Hide Status Area</td>
<td>91</td>
</tr>
<tr>
<td>Show/Hide Text Ruler</td>
<td>92</td>
</tr>
<tr>
<td>Sort Records</td>
<td>81</td>
</tr>
<tr>
<td>Speak (Mac OS)</td>
<td>131</td>
</tr>
<tr>
<td>Spelling Options</td>
<td>114</td>
</tr>
<tr>
<td><strong>U, V, W, X, Y, Z</strong></td>
<td></td>
</tr>
<tr>
<td>Undo/Redo</td>
<td>37</td>
</tr>
<tr>
<td>Unsort Records</td>
<td>82</td>
</tr>
<tr>
<td>Update Link (Windows)</td>
<td>55</td>
</tr>
<tr>
<td>View As</td>
<td>94</td>
</tr>
</tbody>
</table>
Control script steps

Control script steps control the progression of the script by letting you tell FileMaker Pro exactly what to do when and if specific conditions occur.

Use these script steps to:

- call scripts and sub-scripts
- pause and resume a script, based on defined conditions
- conditionally perform script steps using if/then/else logic
- stop a script before it's finished, if a specific condition is met
- create loops that repetitively carry out a sequence of steps in a script, until a condition is met

Note When you perform a script that uses the Get(LastError) function with control script steps, the control script steps do not clear the last error condition reported by FileMaker Pro.

Perform Script

Purpose
Performs a script that is defined in the current file or in another FileMaker Pro file.

Format
Perform Script ["<script name>"; Parameter: <parameter>]

Options
To select a script, click Specify and choose the script from the list.
For Optional script parameter, type the text you want to use or click Edit and use the Specify Calculation dialog box to build a more complex parameter.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
There is no need to open an external file when using a script in it—FileMaker Pro opens it for you.
You can use separate scripts to break complex tasks into smaller segments, which are often easier to troubleshoot. You can also save time by putting common tasks like printing or exporting records into separate scripts, which can then be accessed by more than one script. For example, you might want to use the same page setup and printing options in several scripts. By defining those steps once and saving them in separate scripts, you make it easy to access those steps many times.

Use script parameters to pass text or numbers into a script. When a parameter is specified, that parameter may be accessed within a script or passed to other script steps using the Get(ScriptParameter) function.
Script parameters:

- exist within the parent script only unless explicitly passed to the sub-script, as shown in examples three and four below. If your script calls another script, parameters in the called script are independent of parameters in the parent script.

- exist only for the duration of the script. Script parameters are reset each time a script is performed.

- can be used (but not modified) within a script and can be passed along to sub-scripts by using the Get(ScriptParameter) function as the parameter for the sub-script. You can also specify different parameters each time the sub-script is called using Perform Script. Changing the parameters passed to a sub-script does not modify the value of the parameters returned from Get(ScriptParameter) in the parent script.

- can contain multiple values separated by carriage returns. Use the LeftValues function, MiddleValues function, and RightValues function to isolate a specific parameter.

Tip If you've performed an external script and you want to return to the original file, add an Enter Browse Mode step or Go to Layout step right after the Perform Script step in the original file, so that the script returns to the original file.

Notes

- If you are using FileMaker Pro Advanced to define a custom menu item that performs a script, select the script and optional script parameters. For more information, see Defining custom menus (FileMaker Pro Advanced).

- A runtime solution can perform an external script only if the external file is bound to the solution.

Examples

#Example 1: This is a simple example with no parameters
Go to Layout ["Detailed Report"]
Perform Script ["Print in Landscape"]

#Example 2: This example uses the script parameter to set the title of the report
Go to Layout ["Detailed Report"]
Perform Script ["Sort by Date"; Parameter: "Month End Report"]
Set Field [Sales::Report Title; Get (ScriptParameter)]
Perform Script ["Print in Landscape"]

#Example 3: This example uses a field value as a parameter
Go to Layout ["Detailed Report"]
Perform Script ["Find by Salesperson"; Parameter: Sales::Salesperson]
Set Field [Sales::Report Title; "Sales by " & Get (ScriptParameter)]

#Example 4: This script uses script parameters to pass text entered by the user to another script, which creates a new account

Fields
account_name Text Global Storage
password Text Global Storage
Script: Get New Account Info
#Clear the Account Name and Password fields for new information
Set Field [ first_table::account_name; "" ]
Set Field [ first_table::password; "" ]
Show Custom Dialog [ Title: "Create A New Administrator Account";
Message: "Please enter an account name and password for your new user.";
Buttons: "OK", "Cancel"; Input #1: first_table::account_name; Input #2: first_table::password ]
Perform Script [ "Make New Account"; Parameter:
first_table::account_name & ¶ & first_table::password ]

Script: Make New Account
Add Account [Account Name: LeftWords( Get( ScriptParameter ); 1);
Password: LeftWords( Get( ScriptParameter ); 2); Privilege Set: "[Data Entry Only]"

Pause/Resume Script

Purpose
Pauses a script indefinitely or for a specified length of time so the user can perform other tasks in the current window.

Format
Pause/Resume Script [Duration (seconds) <n>]

Options
Click Specify to display the Pause/Resume Options dialog box, where you can set the following options.
- Select Indefinitely to pause the script until the user clicks Continue (a button created by FileMaker Pro in the status toolbar) or presses Enter.
- Select For duration and enter the number of seconds to pause the script.
- Select For duration and click Specify to create a calculation to determine the number of seconds to pause the script.

Compatibility
This script step is:
- supported in web publishing only if the script is paused indefinitely
- not supported in a FileMaker Server scheduled script

Description
This script step can, for example, wait for a user to enter data, and then guide the user from screen to screen, prompting for data entry as needed. You can also use Pause/Resume Script to help
debug your scripts, for example, to see what value is in a field at a particular point in a script or to evaluate the progress of a script.

The Pause/Resume Script script step operates on the foreground window of the file from which the script is performed. If the script's current window is hidden, Pause/Resume Script brings that window to the front and shows it.

The **For duration** value must evaluate as a number, which is the duration of the pause in seconds. If you use the Specify Calculation dialog box to determine the duration, the calculation result must be a number or your script will not pause.

To give you more control over a user's actions when a script pauses, FileMaker Pro makes some menu commands unavailable to users. You can define buttons to let users perform actions not available on the menus.

**Examples**

The following script displays a welcoming screen for three seconds when the database is opened, then displays a data entry layout. The user cannot stop this script. The default state is off.

Allow User Abort [Off]
Go to Layout ["Welcome"]
Pause/Resume Script [Duration (seconds): 3]
Go to Layout ["Data Entry"]

*Tip* You can create a button with the Resume step to allow the user to resume a script if it's paused.

**Exit Script**

**Purpose**

Forces sub-scripts or external scripts to stop immediately and return to the main script, with or without an optional script result.

**Format**

Exit Script [Result]

**Options**

Click **Specify** to specify a value in the Specify Calculation dialog box. The value is typically based on the state of the current script. The value is passed back to the calling script, where it can control the script's logic.

**Compatibility**

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**

If the optional script result is specified, the script result is returned and stored in the main script until another sub-script that returns a value is called. The script result gets deleted when you exit the main script. The script result can be stored in a local or global variable for later use.
Examples

#Example 1: Master Mailing Label Script
Go to Layout ["Mailing Labels"]
Perform Script ["Find records for California mailing"]
Go to Layout ["List View"]

#Find records for California mailing sub-script
Perform Find [Restore]
If [Get (ScriptResult)=0]
    Show All Records
    Go to Layout ["Data Entry"]
    Exit Script
Else
    Print []
End If

#Example 2: Assign Letter Grade
If [Get(ScriptParameter) >= 90]
    Exit Script [Result: "A"]
Else If [Get(ScriptParameter) >= 80]
    Exit Script [Result: "B"]
Else If [Get(ScriptParameter) >= 70]
    Exit Script [Result: "C"]
Else If [Get(ScriptParameter) >= 60]
    Exit Script [Result: "D"]
Else
    Exit Script [Result: "F"]
End If

Halt Script

Purpose
Forces all scripts (including any sub-scripts or external scripts) to stop immediately.

Format
Halt Script

Options
None.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.
Description
While pressing Esc will also halt a script, Halt Script is useful if you are also using the Allow User Abort script step to disable the Esc key. Halt Script stops the script when Allow User Abort is set to Off.

Examples
You can use the Halt Script script step with the Show Custom Dialog script step to present a user with options. If the user clicks Cancel, the script stops; otherwise, the script continues.

Show Custom Dialog ["Delete all records?"]
If [Get (LastMessageChoice) = 2]
    Halt Script
End If
Delete All Records [No Dialog]

Purpose
Evaluates a Boolean calculation and performs a conditional action based on that evaluation.

Format
If [<Boolean calculation>]

Options
Click Specify to define the Boolean calculation. In the Specify Calculation dialog box, type the calculation you want evaluated, or use the field list (on the left) and the functions list (on the right) with the mathematical and text operators to build the calculation.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
If the calculation result is any number except zero, the calculation evaluates to True and the subsequent script steps are executed. If the calculation result is zero, no data, or does not resolve into a number, then the calculation evaluates to False and the subsequent script steps are not executed.

Every If step must have a corresponding End If script step somewhere after the If step and at the same indentation. Whenever you use an If script step, the Manage Scripts feature automatically enters an End If step.

You can also add additional conditions by using the Else If script step and Else script step.

Note If you do not specify a calculation or if the calculation is unsuccessful, it will evaluate as False. Use the Get(LastError) function to capture these errors.
Examples

If [Get (AccountName) = "Jim Davis"]
   Go to Layout ["File One"]
   Perform Find [Restore]
End If

Else If

Purpose
Evaluates a Boolean calculation and performs a conditional action based on that evaluation, like the If script step.

Format
Else If [<Boolean calculation>]

Options
Click Specify to define the Boolean calculation. In the Specify Calculation dialog box, type the calculation you want evaluated, or use the field list (on the left) and the functions list (on the right) with the mathematical and text operators to build the calculation.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
Must follow an If step or another Else If step; it can’t follow an Else script step.

When an If script step calculation evaluates to True (not zero), FileMaker Pro performs the steps that are indented under it. When an If script step evaluates to False (zero), script execution moves to any following Else If steps as long as they evaluate to False. When an Else If statement evaluates to True, the block of steps under the Else If is executed. An evaluation of True ends the Else If process, and upon reaching the next Else If or Else statements, execution will skip ahead to End If.

Examples

If [Get (AccountName) = "Jim Davis"]
   Go to Layout ["File One"]
   Perform Find [Restore]
Else If [Get (AccountName) = "Julia Vargas"]
   Go to Layout ["File Two"]
   Perform Find [Restore]
Else If [Get (AccountName) = "Gerard LeFranc"]
   Go to Layout ["File Three"]
   Perform Find [Restore]
Else
   Go to Layout ["File Four"]
   Show All Records
End If
Note In this example, Get(AccountName) only returns usable values if database users log in using assigned account names. If no accounts are created or required, Get (AccountName) returns the default value 'Admin'.

Else

Purpose
Performs one set of steps if a calculation is True (not zero), and another set of steps if the calculation is False (zero).

Format
Else

Options
None.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
Use with the If script step.

Examples
If [Table1::State = "CA"]
   Perform Script ["Compute CA Tax and Total"]
Else
   Perform Script ["Compute Total"]
End If

End If

Purpose
Marks the end of an If script step structure.

Format
End If

Options
None.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.
Control script steps

Description

Every If script step must have a corresponding End If script step somewhere after it at the same indentation. Whenever you use an If script step, the Manage Scripts feature automatically enters an End If step.

Examples

If [Table1::State = "CA"]
    Perform Script ["Compute CA Tax and Total"]
Else
    Perform Script ["Compute Total"]
End If

Loop

Purpose

Repeats a set of script steps.

Format

Loop

Options

None.

Compatibility

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description

This script step and the End Loop script step mark the beginning and end of a repeating structure of script steps. The Manage Scripts feature performs the script steps that are enclosed within the loop structure until it encounters one of the following:

- an Exit Loop If script step
- a Go to Record/Request/Page script step or Go to Portal Row script step when the Exit Loop After First/Last option is selected

Every Loop step must have a corresponding End Loop step somewhere after the Loop step and at the same indentation. Whenever you use a Loop script step, the Manage Scripts feature automatically enters an End Loop step.

Examples

Go to Record/Request/Page [First]
Loop
    Set Field [Table1::Bonus; 2500.00]
    Go to Record/Request/Page [Next]
    Exit Loop if [Get (RecordNumber) = 25]
End Loop
Exit Loop If

**Purpose**
Exits a loop if the specified calculation is True (not zero).

**Format**
Exit Loop If [<Boolean calculation>]

**Options**
Click Specify to define the Boolean calculation. In the Specify Calculation dialog box, type the calculation you want evaluated, or use the field list (on the left) and the functions list (on the right) with the mathematical and text operators to build the calculation.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
If the calculation is True (not zero), the script continues with the script step that follows the End Loop script step.
If the calculation if False (zero), the loop is not exited and the script continues with the script step that follows the Exit Loop If script step.

**Examples**
This example calculates a bonus for the top 10 sales agents based on their sales. The script finds all records and sorts by a field called "Sales" (descending) to arrange the agents by sales performance. The script starts at the first record and sets the loop counter variable to 1 (in this example the loop counter field use is a number field with global storage enabled -- this is the Count field in the example below). Then, the script calculates the bonus as one quarter of the agent’s salary. The script then goes to the next record and exits the loop if the counter reaches 10. The loop counter increments by 1 and the loop repeats.

```
Show All Records
Sort Records[Restore; No dialog]
Set Field [Table1::Count; 1]
Loop
  Set Field [Table1::Bonus; Table1::Salary * .25]
  Go to Record/Request/Page[Next]
  Exit Loop If [Table1::Count = 10]
  Set Field [Table1::Count; Table1::Count + 1]
End Loop
```
End Loop

**Purpose**
Marks the end of a Loop script structure.

**Format**
End Loop

**Options**
None.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
Whenever you use a Loop script step, the Manage Scripts feature automatically enters an End Loop step. The steps between Loop and End Loop are automatically indented and the End Loop step is placed at the same indentation level as its corresponding Loop step.

**Examples**
Go to Record/Request/Page[First]
Loop
  Set Field [Table1::Salary; Table1::Salary * 1.1]
  Go to Record/Request/Page[Exit after last; Next]
End Loop

Allow User Abort

**Purpose**
Allows users to stop a script if set to **On** (default), or prevents users from stopping a script if set to **Off**.

**Format**
Allow User Abort [<on or off>]

**Options**
- **On** to allow users to stop a script by pressing Esc or Command-.(period) (Mac OS).
- **Off** to prevent users from stopping a script by pressing Esc or Command-.(period) (Mac OS).

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.
Description

By default, Allow User Abort is turned on for an entire script. In other words, users can stop a script by pressing Esc or Command-. (period) (Mac OS). To prevent users from stopping a script, use the Allow User Abort script step and set it to Off. FileMaker Pro automatically turns Allow User Abort on after the script finishes running.

Examples

To set up a kiosk solution, use:

Allow User Abort [Off]

Then, create the rest of the script as a looping script that never ends. Use the Allow User Abort script step as the first step to prevent people from closing the file or exiting the application.

The next script steps display a “welcome” screen for three seconds when the database is opened, then continues to a data entry layout. The user cannot cancel this “welcome” screen display.

Allow User Abort [Off]
Go to Layout ["Welcome"]
Pause/Resume Script [Duration (seconds): 3]
Go to Layout ["Data Entry"]
Allow User Abort [On]

The following script verifies that all records have client numbers entered, audibly notifies the user when it finds a missing number, and waits so the user can enter the number. This script runs when the database closes and can’t be cancelled by the user.

Allow User Abort [Off]
Go to Record/Request/Page [First]
Loop
  If [IsEmpty(Table1::clientNumberField)]
    Speak ["Client number is missing"]
    Pause/Resume Script [Indefinitely]
  End If
  Go to Record/Request/Page [Exit after last; Next]
End Loop
Allow User Abort [On]

Set Error Capture

Purpose

Suppresses or enables normal FileMaker Pro alert messages.

Format

Set Error Capture [<on or off>]

Options

- **On** suppresses FileMaker Pro alert messages and some dialog boxes. If the error result is 100 or 803, then certain standard file dialog boxes are suppressed, such as the Open dialog box.
- **Off** re-enables the alert messages.
Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Description
Use this script step to handle errors in a manner consistent with the functions the script performs. By using the Get(LastError) function immediately after a script step, you can verify that the step was performed properly.

By using the Get (LastError) function with the control script steps, you can make sure your script performs correctly. When you decide to suppress alerts, it is important that you anticipate as many problems as possible and include clear instructions for what to do when an error condition is encountered.

Note Use Get (LastError) immediately after the script step you intend to test; a successful intervening step may clear the error code you were intending to capture.

Examples
Set Error Capture [On]
Perform Find [Restore]
Set Field [Table1::gErrorCaptureField; Get (LastError)]
#The field Table1::gErrorCaptureField is a global text field
If[Table1::gErrorCaptureField <> 0]
    Show Custom Dialog ["Couldn't find the record..."]
End If
Set Error Capture [Off]

Set Variable

Purpose
Sets a local or global variable to a specified value.

Format
Set Variable [<variable name> {[<repetition number>]}; Value:<value or formula>]

Options
Click Specify to set the variable options:
• Name is the name of the variable you want to create. Prefix the name with $ for a local variable or $$ for a global variable. If no prefix is added, $ is automatically added to the beginning of the name.
• Value is the value the variable is set to. You can enter text or specify a calculation.
• Repetition is the repetition (index) number of the variable you want to create. If no repetition is specified, the default is 1.
Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
If a variable doesn’t exist, this script step will create one. A variable name has the same naming restrictions as a field name. For more information, see About naming fields.

Local and global variables can be created and used in calculations.

• A local variable can only be used in script steps in the currently executing script. The value in a local variable is cleared when the script exits.

• A global variable can be used in a calculation or script anywhere in a file, for example, other scripts or file paths. The value of a global variable is not cleared until the file is closed.

• Local and global variables (or even two local variables in different scripts) can have the same name but they are treated as different variables and can store different values.

Examples
The following example sets a local variable.
Set Variable [$commission; Value:.05]
The following example sets a global variable.
Set Variable [$$commission; Value:.1]
The following example sets a global variable at a specified repetition number.
Set Variable [$$bonus[2]; Value:3 * 4]

Install OnTimer Script

Purpose
Runs a specified script at the specified interval. Installs a single timer per window. After the specified interval has passed, the next time the application is idle, runs the specified script.

Format
Install OnTimer Script ["<script name>"; Interval: <number>]

Options
Click Specify to set the variable options:

• Specify script is the name of the script you want to run.

• Interval seconds is the value in seconds that the timer waits before running the script.

Compatibility
This script step is:

• not supported in web publishing

• not supported in a FileMaker Server scheduled script
Description
This step repeats until the window in which it is running closes or its options change. You can also halt the step by specifying another Install OnTimer Script script step for the window with no script specified.

The default interval is zero; you must specify a script name and an interval.

Examples
The following example runs the script MyScript every 10 minutes.

Install OnTimer Script ["MyScript"; Interval: 600]
Navigation script steps

Navigation script steps move to different areas of a database. Use Navigation script steps to:

• go to a specific record or find request
• switch to a specific layout
• move among fields on a layout
• emulate pressing Enter or Tab
• switch to Browse mode to work with contents of a file
• switch to Find mode to fill out find requests
• switch to Preview mode to see how records, forms, or reports will look when they’re printed

Go to Layout

**Purpose**
Switches to the specified layout.

**Format**
Go to Layout ["<layout name or layout number>"]

**Options**
**Specify** lets you choose the target layout. For **Specify**, you can:

• choose the original layout. The original layout is the active layout when the script is initiated.
• choose a specific layout by name from the list of layouts you’ve defined in your file.
• choose **Layout Name by calculation**, and use the Specify Calculation dialog box to create a formula whose result is a valid layout name.
• choose **Layout Number by calculation**, and use the Specify Calculation dialog box to create a formula whose result is a valid layout number. (Layout numbers correspond to the order of the layouts in the file.)

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
This script step is useful when you begin a script to make sure the user starts with the correct layout displayed.

**Notes**
• The Go to Layout script step can only take you to layouts defined in the same file as the script itself. To go to layouts in an external file, define a script in that file using the Go to Layout script step, and call that script from the first file using the **Perform Script** script step.
• Layout Name by calculation recognizes layouts with the same names in the order in which they were created. If you have multiple layouts with the same name, either select the specific layout you want from the layout list or use Layout Number by calculation.

• Define an unstored calculation field with the function `Get(LayoutNumber)` function and place it on your layouts to verify the layout numbers of your layouts.

• When a file has two or more tables, FileMaker Pro appends the name of the source table to the layout name for clarity. For example, Go to Layout ["Layout #2" (TableB)].

**Examples**

Enter Browse Mode []
Go to Layout ["Contacts"]
New Record/Request
Go to Field [Table1::First Name]

**Go to Record/Request/Page**

**Purpose**

In Browse mode, moves to a record in the found set. In Find mode, displays a find request. In Preview mode, moves to a page in a report.

**Format**

Go to Record/Request/Page [<first/last/previous/next/by calculation>]

**Options**

Specify lets you choose from the following options.

• **First** moves to the first record in the file or found set, displays the first find request, or moves to the first page in a report.

• **Last** moves to the last record in the file or found set, displays the last find request, or moves to the last page in a report.

• **Previous** moves to the previous record in the file or found set, displays the previous find request, or moves to the previous page in a report. **Exit after last** tells FileMaker Pro to stop the script when it reaches the last record in the found set. You can use **Exit after last** with the Loop script step to exit out of a loop after the last record.

• **Next** moves to the next record in the file or found set, displays the next find request, or moves to the next page in a report. **Exit after last** tells FileMaker Pro to exit the script or control structure (like a loop) when it reaches the last record in the found set.

• **By Calculation** lets you create a calculation to determine the index number for the desired record, find request, or page in Preview mode.

**Note** **Exit after last** will exit a loop but does not stop or exit a script.

**Compatibility**

This script step is also supported in web publishing and in a FileMaker Server scheduled script.
Examples

Go to Record/Request/Page [First]
Go to Record/Request/Page [Last]
Go to Record/Request/Page [Previous]
Go to Record/Request/Page [Next]

This command uses a calculation to determine the correct number:

Go to Record/Request/Page [MyDatabase::Number Field]

Go to Related Record

**Purpose**

Goes to the current related record(s) in a related table, except when this script step is used from an active portal row.

**Format**

Go to Related Record [From table: "<table name>"; Using layout "<layout name>"]

**Options**

- **Get related record from** lets you select the source relationship from a list of tables in the current database. If the table is not in the list or if you need to add or change a relationship, [Manage Database](#) displays the Manage Database dialog box, where you can create or edit relationships.
- **Show record using layout** lets you choose a layout in the current file that will be used to display the related record(s).
- **Use external table’s layouts** opens the file containing the external table you specify and lets you choose a layout from that file in which to display the related record(s). This option is only available if the source relationship you selected references a table in another file.
- **Show in new window** allows you to show the related records in a new window and lets you specify the settings for the new window. For more information, see [New Window script step](#).
- **Specify** lets you edit the settings you have previously chosen for the new window.
- **Show only related records** creates a new found set in the related table with the options either to match the current record or to match the entire found set.
- **Match current record only** creates a new found set in the related table containing a set of records that match the current record. For example, if there are three records in the related invoice table that match the customer record in the customer table you are currently viewing, and you want to see all three invoices, use **Match current record only** to display the three invoices.
- **Match all records in the current found set** creates a new found set in the related table that matches all records in the current found set. For example, if you have a found set of ten customers and there are forty invoices in the invoice table that match any of these ten customers, use **Match all records in the current found set** to display the forty matching invoices.
**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
If this script step is used from an active portal row, and the portal’s table is the related table, then the related record in that table is made current. If the portal’s table is not the related table, the first related record in the found set is made current.

You have the option to view the related records in a new window. This script step requires a relationship to be in effect.

For example, suppose you have an Invoices table that’s related to a Customers table and you’re currently looking at a particular invoice. In the Invoices table, you have a button that’s set to Go to Related Record [Customers]. Clicking this button determines which customer record your current invoice is related to and immediately goes to that record in the Customers table. If the record in question is not currently in the Customers table’s found set, the script can perform a find to make the record current.

This script step also works with portal rows. If the currently active portal row is row #3 and you execute a Go to Related Record script step using the same relationship as the portal, then FileMaker Pro goes to that particular related record in the related table.

When used with a table in an external file, this script step will open the file containing the external table and, if selected, go to the external layout you specify.

**Note** To bring the new window with the related record(s) to the front automatically, add the Select Window script step.

**Examples**
The following goes to a related record in the table "MyRelatedTable".

Go to Related Record [From table: "MyRelatedTable";
Using layout: “MyLayout”]

The following goes to a related record in the table "MyRelatedTable" and shows a found set of related records only.

Go to Related Record [Show only related records;
From table: "MyRelatedTable"; Using layout: “MyLayout”]

**Go to Portal Row**

**Purpose**
Navigates among the rows in the active portal.

**Format**

Go to Portal Row [<first/last/previous/next/by calculation>]
Options
Specify lets you choose from the following options.

- **First** moves to the first row in the portal.
- **Last** moves to the last row in the portal.
- **Previous** moves to the previous row in the portal. **Exit after last** tells FileMaker Pro to exit the loop or the enclosing script.
- **Next** moves to the next row in the portal. **Exit after last** tells FileMaker Pro to exit the loop or the enclosing script.
- **By Calculation** moves to the row number that is the result of the calculation you create.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
When no portal is active, the script uses the first portal in the layout stacking order.

This script step tries to keep the same related field selected when changing rows. If no field in a portal is selected, the script selects the first related field it can enter.

When a related field in a portal is selected, you can use this step to move to the same field in another portal row. For example, if the third field in the second portal row is selected, Go to Portal Row [Next] goes to the third field in the third portal row.

Examples
The following selects the first portal row in the current record.

Go to Portal Row [Select, First]

Go to Object

Purpose
Moves to the specified object on the current layout.

Format

Go to Object [Object Name: "<object name>";
Repetition: <repetition number>]

Options
Click Specify to display the "Go to Object" Options dialog box, where you can set the following options:

- **Object Name** is the named object to make active on the current layout. To assign an object name, choose View menu > Object Info in Layout mode and enter a name.
- **Repetition** (optional) lets you choose a field's repetition to go to. Otherwise defaults to 1. This option is ignored if the object is not a field.
Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
This script step uses an object name to identify an object, so you must assign a unique object name to each object on a layout that you want to go to.

Note If the object is a repeating field, you can also specify which repetition that you want to go to.

Examples
The following example navigates to a text field named "First name":
Go to Object [Object Name: "First Name"]
The following example navigates to a button named "Launch button":
Go to Object [Object Name: "Launch button"]
The following example navigates to the repetition (2) of a field named "Value":
Go to Object [Object Name: "Value"; Repetition: 2]
The following example navigates to an object whose name is the current day of the week:
Go to Object [Object Name: DayName (Get(CurrentDate))]

Go to Field

Purpose
Moves to the specified field on the current layout.

Format
Go to Field [Select/perform; <table::field>]

Options
- Select/perform tells FileMaker Pro to perform an action on the contents of a field. All text in a field is selected with this option. The contents of the field (sound, movie, or OLE) determine what action is performed. If the field contains a sound, then the sound is played. If the field contains a movie, the movie is played. If the field holds an OLE object, FileMaker Pro activates the application that created the object, and then performs the primary command for that application. For example, if the primary command is Edit, Select/perform opens the document in a state ready for editing.

- Select Go to target field or click Specify to specify the target field.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.
Description

This script step can also perform an action on an object in that field. For instance, in a container field, it can play a sound or a movie, or activate an OLE object (Windows).

To play a sound or movie, or perform the primary command for an OLE object (Windows), use the Select/perform option.

Note  If this script step specifies a field in a tab panel that is not in front, the specified field is selected and the tab panel it is in moves to the front of the tab control. If, however, the same field appears elsewhere on the layout and the script finds that field first, the tab panel with that same field will not come forward. Instead, assign an object name to the field and use the Go to Object script step to move to that instance of the field.

Examples

Enter Browse Mode []
Go to Layout ["Contacts"]
New Record/Request
Go to Field [Table1::First Name]

Go to Next Field

Purpose

Moves to the next field in the tab order of the current layout.

Format

Go to Next Field

Options

None.

Compatibility

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description

If no field is selected when this script step is performed, this script step moves to the first field in the tab order of the current layout. If a field is formatted as a button, the field object is selected, not the button object.

Examples

Go to Layout ["Layout #2"]
Go to Record/Request/Page [First]
Go to Field [Table1::First Name]
Pause/Resume Script [Indefinitely]
Go to Next Field
Go to Record/Request/Page [Next]
Go to Previous Field

**Purpose**
Moves to the previous field in the tab order of the current layout.

**Format**
Go to Previous Field

**Options**
None.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
If no field is selected when this script step is performed, this script step moves to the last field of the tab order of the current layout. If a field is formatted as a button, the field object is selected, not the button object.

**Examples**
Go to Layout ["Layout #2"]
Go to Field [Table1::FAX]
Pause/Resume Script [Indefinitely]
Go to Previous Field

Enter Browse Mode

**Purpose**
Switches to Browse mode, where you can enter or edit data.

**Format**
Enter Browse Mode [Pause]

**Options**
Pause temporarily stops the script so the user can enter data.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Examples**
Enter Browse Mode []
Go to Layout ["Layout #1"]]
Show All Records
Sort [Restore, No dialog]

Enter Find Mode

**Purpose**
Switches to Find mode, where you can search for sets of records.

**Format**
Enter Find Mode [Restore; Pause]

**Options**
- **Pause** temporarily stops the script to allow the user to enter a find request.
- **Specify find requests** allows you to create and edit requests for use with this script step.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
When you select **Specify find requests** at the time the script step is defined, FileMaker Pro displays the current find request(s), if any. Additional find requests can be defined, and existing find requests can be duplicated, edited, or deleted.

To edit a find request, select it from the list in the Specify Find Requests dialog box and click **Edit**.

In the Edit Find Requests dialog box, select the action (**Find Records** or **Omit Records**) you want the find request to perform. A single find request can either find records or omit records, but cannot perform both actions at the same time. Use multiple find requests to find and omit records from within a single script step.

For each field in your request, specify the criteria for which you want FileMaker Pro to search.

**Examples**
Go to Layout ["Article View"]
Enter Find Mode [Restore; Pause]
Perform Find []
Enter Preview Mode

**Purpose**
Switches to Preview mode, where you can see how records, forms, or reports will look when they're printed.

**Format**
Enter Preview Mode [Pause]

**Options**
*Pause* temporarily stops the script to allow you to examine the preview result before proceeding with the next step in the script.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Examples**
Sort [Restore, No dialog]
Go to Layout ["Sales Summary"]
Enter Preview Mode [Pause]
Go to Layout [original layout]
Enter Browse Mode []
Unsort
Go to Record/Request/Page [First]
Editing script steps

With editing script steps, you can cut, copy, paste, or clear the contents of a field; undo or redo previous actions; or select the contents of a field.

Undo/Redo

Purpose
Reverses, restores, or switches between the most recently performed actions in the file.

Format
Undo/Redo [Undo; Redo; Toggle]

Options
Undo reverses the previously performed action in the file.
Redo restores the previously undone action in the file.
Toggle switches between the two most recently performed actions in the file.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
Multiple actions can be undone by repeatedly executing this script step with the Undo option. The number of actions that can be undone or redone is limited only by the amount of available memory.

Examples
Undo/Redo [Undo]
Undo/Redo [Redo]
Undo/Redo [Toggle]

Cut

Purpose
Deletes the contents of the specified field in the current record and saves the contents to the Clipboard.

Format
Cut [Select; <table::field>]
Options

- **Select entire contents** deletes the contents of a field and saves it to the Clipboard. If you do not use Select entire contents, only the selected portion of the field's data is cut.
- Select **Go to target field** or click **Specify** to select the field whose contents you want to cut.

Compatibility

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description

This script step removes the contents of the field. To duplicate the field information, use the `Copy` script step.

**Note** In a web-published database, use a Commit Record/Request script step after a Cut script step to update the record in the browser window.

Examples

The following example takes notes from a "Recent Notes" field and pastes them at the end of a "Previous Notes" field (creating a history of notes).

Enter Browse Mode [ ]
Cut [Select, Table1::Recent Notes]
Paste [Table1::Previous Notes]

Copy

**Purpose**

Copies the contents of the specified field in the current record and saves them to the Clipboard.

**Format**

Copy [Select; <table::field>]

**Options**

- **Select entire contents** copies the entire contents of a field to the Clipboard. If you do not use Select entire contents, only the selected portion of the field's data is copied.
- Select **Go to target field** or click **Specify** to select the field whose contents you want to copy. When no field is specified and nothing is selected, FileMaker Pro copies the values from all fields of the current record.

Compatibility

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**

This script step duplicates the contents of the field. To remove the field information, use the `Cut` script step.
Paste

**Purpose**

Pastes the contents of the Clipboard into the specified field in the current record.

**Format**

```plaintext
Paste [Select; No style; <table::field>]
```

**Options**

- **Select entire contents** replaces the contents of a field with the contents of the Clipboard. If you do not use **Select entire contents**, Paste copies the contents of the Clipboard to the currently selected portion of the field.
- **Paste without style** tells FileMaker Pro to ignore all text style and formatting associated with the Clipboard contents.
- **Select Go to target field** or click **Specify** to specify the field to paste into.
- **Link if Available** (Windows) tells FileMaker Pro to choose a link over other formats on the clipboard. If both a link and an embedded object are present on the clipboard, the link is selected. If a link is available, it is selected over other formats.

**Compatibility**

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**

If you try to paste data that doesn’t match the field type of the target field, FileMaker Pro pastes the data but displays a validation alert when you attempt to exit the field. If the field is not on the current layout, FileMaker Pro returns an error code which can be captured with the `Get(LastError)` function.

If the item being pasted is an OLE object, you can choose a link over other possible formats in the Clipboard. The **Link if Available** option has no effect when executed on a Mac OS system.

**Note** In a web-published database, use a Commit Record/Request script step after a Paste script step to update the record in the browser window.

**Examples**

- Go to Record/Request/Page [First]
- Copy [Select; Invoices::Company Name]
- Go to Record/Request/Page [Next; Exit after last]
- Paste [Select; No style; Invoices::Company Name]
Clear

**Purpose**
Deletes the contents of the specified field in the current record.

**Format**
Clear [Select; <table::field>]

**Options**
- **Select entire contents** deletes the entire contents of a field. If you do not use Select entire contents, only the selected portion of the field's data is deleted.
- **Select Go to target field** or click Specify to specify the field whose contents you want to delete.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
This script step removes the contents of the field without placing the contents on the Clipboard. If you want to cut and paste the field information, use the Cut script step. Use Undo immediately to restore the contents.

**Note** In a web-published database, use a Commit Record/Request script step after a Clear script step to update the record in the browser window.

**Examples**
The following example clears the values in a repeating field with three repetitions.

Clear [Select, table::field[3]]
Clear [Select, table::field[2]]
Clear [Select, table::field]

Set Selection

**Purpose**
Allows the user to specify the starting and ending position of a selection in a field.

**Format**
Set Selection [Start Position: <n>; End Position: <n>]

**Options**
- **Select Go to target field** or click Specify to specify the field whose contents you want to select.
- **Specify** lets you set the starting and ending positions of a selection, either by entering the start and end numbers directly or by using a calculation to determine them.
Editing script steps

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
The starting and ending values can be typed in directly or determined via a calculation. If no target field is specified the current active field is used.

Notes
- Field position is determined by the number of characters, including spaces, beginning with position 1, the first character in the field. The selection includes all values beginning with the Start Position and concluding with the End Position. For example, if Field1 has the value "abcdefgh", and the script parameters specify a start position of "3" and an end position of "6", the script step will return the selection "cdef".
- If the start position is valid and the end position out of range, everything from the start position to the end of the field is selected. If the end position only is valid, the cursor moves to the end position, and nothing is selected. If both start and end positions are invalid, the cursor moves to the end of the field contents.
- No action is taken if the user attempts to perform a selection on a container field.
- Data selected out of visual range is scrolled into view.

Examples
Go to Layout ["Data Entry"]
Set Selection [Table1::Account Number; Start Position: 5; End Position: 10]
Copy []

Select All

Purpose
Selects the entire contents of the current field.

Format
Select All

Options
None.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.
Examples
Go to Field [Table1::Performance Review]
Select All
Copy []
Go to Record/Request/Page [Next]
Go to Field [Performance Review]
Paste []

Perform Find/Replace

Purpose
Finds/replaces data according to the options in the Specify Find/Replace dialog box.

Format
Perform Find/Replace [No dialog; "<text to be found>"; "<replacement text>"; Find Next/Replace & Find/Replace/Replace All]

Options
• **Perform without dialog** prevents display of the Find/Replace Summary dialog box at the end of the find/replace operation. This option also prevents display of the confirmation dialog box when a Replace All operation is executed.
  
  If you want the user to be able to enter find or replace criteria, use the [Open Find/Replace script step].

• **Specify** displays the Specify Find/Replace dialog box, where you can set search options and the type of find/replace operation to be performed.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Examples
Perform Find/Replace ["Houston"; "Dallas"; Replace & Find]
Fields script steps

Fields script steps operate on specific fields. With these script steps, you can:

• paste data into fields
• import information into fields
• evaluate a calculation and assign the result to a field

Set Field

**Purpose**

Replaces the entire contents of the specified field in the current record with the result of a calculation.

**Format**

Set Field [ \textit{table::field}; \textit{value or formula}]  

**Options**

• Select Specify target field or click Specify to specify the field whose contents you want to replace. If no field is specified and a field is selected in Browse mode or Find mode, that field is used.

• For Calculated result, click Specify to define the calculation. In the Specify Calculation dialog box, type the calculation you want evaluated, or use the field list (on the left) and the functions list (on the right) with the mathematical and text operators to build the calculation.

**Compatibility**

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**

• The result of the calculation must match the field type. For example, you can't assign a date calculation to a container field.

• The specified field doesn’t have to be on the current layout.

• If the result of the calculation doesn’t match the target field type, and the validate option for the field is set to Always, the field will not be set and an error code is returned (which can be captured with the \texttt{Get(LastError) function}).

• When possible, the Set Field script step makes the record active and leaves it active until the record is exited or committed. Scripts that use a series of Set Field script steps should group these steps together if possible, so that subsequent Set Field script steps can act on the record without having to lock the record, download and upload data, index the field, and so on, after each individual Set Field script step. These functions and record level validation are performed after the record has been exited or committed.
Examples
The following example calculates 7 percent of the Salary field and assigns the result to the field SDI. Note that Set Field works whether or not the field is on the active layout.

Set Field [Table1::SDI; Salary * .07]

Set Field By Name

Purpose
Replaces the entire contents of a calculated target field in the current record with the result of the calculated value.

Format
Set Field By Name[<calculated target field>; <calculated value>]

Options
- Select Specify target field or click Specify to create a calculation to specify the field whose contents you want to replace. In the Specify Calculation dialog box, use the field list (on the left) and the functions list (on the right) with the mathematical and text operators to build the calculation.
- For Calculated result, click Specify to define the calculated value.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
The Set Field By Name script step lets you create a calculation to specify a field name, then change the value of the field either literally or based on a second calculation.

The calculated target field must return a text result.

If quotation marks are not included around the fully qualified field name, the target field name is obtained from the named field.

If no field is specified and a field is selected in Browse mode or Find mode, that field is used.

Notes
- The specified field doesn’t have to be on the current layout.
- Set Field By Name ignores validation checking.
- When possible, the Set Field By Name script step makes the record active and leaves it active until the record is exited or committed. Scripts that use a series of Set Field By Name script steps should group these steps together if possible, so that subsequent Set Field By Name script steps can act on the record without having to lock the record, download and upload data, index the field, and so on, after each individual Set Field By Name script step. These functions and record-level validation are performed after the record has been exited or committed.
**Example 1**

In the following example, a calculation identifies the target field in which a value should be replaced based on the geographical location in which the data was entered. Then the script enters a calculated value in the target field based on the appropriate currency exchange rate.

```
Set Field By Name
[If (Members::Country = "USA" ;
   Members::Fee Paid-USD ; Members::Fee Paid-GBP);
If (Members::Country = "USA" ;
   Members::Fee*<USCurrencyRate> ;Members::Fee*<GBCurrencyRate>);
```

**Example 2**

The following example replaces the contents of the Country field located in Table1 with the data located in the Name field in related Table2.

```
Set Field By Name["Table1::Country";Table2::Name]
```

**Note** To ensure proper evaluation, FileMaker Pro must treat the target field as a literal string. Therefore, you must include quotation marks around the calculated target field.

**Example 3**

The following example evaluates the global target field Location::fieldName, then replaces the contents of fieldName with the data value located in the Name field in related Table2.

```
Set Field By Name[Location::fieldName;Table2::Name]
```

**Note** The context for the calculation (determined in the upper left corner of the Specify Calculation dialog box) is essential for proper evaluation.

**Example 4**

The following example uses the GetFieldName function to ensure that FileMaker Pro retrieves the fully qualified name of the target field and the Evaluate function to extract the value stored in the target field, then replaces the contents of fieldName with the data value located in the Name field in related Table2.

```
Set Field By Name
    [GetFieldName(Evaluate(Location::fieldName));Table2::Name]
```

Set Next Serial Value

**Purpose**

Resets the next serial value in an auto-entry serial number field.

**Format**

```
Set Next Serial Value [<table::field>; <value or formula>]
```
Options

- Select Specify target field or click Specify to specify the serial number field on which the script step will operate. The field you specify must be defined as an auto-entry serial number field.
- Calculated result: Click Specify to enter the next serial value or create a calculation to determine the next serial value.

Compatibility

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description

Allows you to use the Manage Scripts feature to update the value. You can define this script step to use any calculation expression to determine the next serial value of a field that has been defined as an auto-entry serial number field. The calculation always evaluates to a text result.

For example, you might want to reset the next serial value after you do one of the following:

- import records into a FileMaker Pro database with an auto-entry serial number field
- delete multiple serialized numbers from a FileMaker Pro database

Notes

- This script step affects the definition of the field you specify instead of the actual contents of the field that you specify.
- This script step can operate on multiple files. If you specify a field in another file, then FileMaker Pro attempts to update the serial number for the specified field in the other file. To specify a field in another file, define a relationship to that file and use Specify target field to select a field from that file.

Examples

The following example calculates the number of the next available invoice ID. If the invoice ID contains non-numeric data, then the calculation would need to be more sophisticated to maintain the numeric and non-numeric data.

**Go to Record/Request/Page [Last]**
**Set Next Serial Value [Table1::Invoice ID; Table1::InvoiceID + 1]**

Insert Text

**Purpose**

Pastes a text value into a field in the current record.

**Format**

Insert Text [Select; <table::field>; "<text>"}
Options

• **Select entire contents** replaces the contents of a field. If you do not select this option, Insert Text inserts the specified value at the end of the field's data.

• Select **Go to target field** or click **Specify** to specify the field to receive the pasted information. If no field is selected, the Insert Text command will place the specified text after the insertion point. If no field is active at the time the command executes, it has no effect. Also, if the selected field is not present on the current Layout, the Insert Text command has no effect.

• **Specify** displays the Specify dialog box where you can enter the exact value you want to paste.

Compatibility

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description

This script step pastes text that is specified in the script, instead of text provided by the user.

Notes

• This script step is intended to paste text into text field types. To insert other types of data into other types of fields, use either the **Insert Calculated Result** script step or the **Set Field** script step.

• If the specified field does not exist on the layout where the script is being performed, Insert Text returns an error code which can be captured with the **Get(LastError)** function.

• In a web-published database, use a Commit Record/Request script step after an Insert Text script step to update the record in the browser window.

Examples

Insert Text [Select; Table1::LastName; "Smith"]

Insert Calculated Result

**Purpose**

Pastes the result of a calculation into the current field in the current record.

**Format**

Insert Calculated Result [Select; <table::field>; <formula>]
Options

- **Select entire contents** replaces the contents of a field. If you don't select this option, Insert Calculated Result replaces only the selected portion of the current field, or inserts the result at the insertion point. The default insertion point is at the end of the field's data.

- Select **Go to target field** or click **Specify** to specify the field to paste the calculated results into. The target field must be present on the layout to paste successfully. If no field is active when the script is performed, the step has no effect.

- **Calculated result**: Click **Specify** to define the calculation whose results will be inserted by this script step. In the Specify Calculation dialog box, type the calculation you want evaluated, or use the field list (on the left) and the functions list (on the right) with the mathematical and text operators to build the calculation.

**Note** If the specified field does not exist on the layout where the script is being performed, Insert Calculated Result returns an error code which can be captured with the `Get(LastError)` function.

Compatibility

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Examples

The following script switches to a specific layout, pastes the current user name, then switches to a different layout. Note that Insert Calculated Result only works when the field is on the active layout.

```
Go to Layout ["User Name Layout"]
Insert Calculated Result [Table1::UserNameField; Get(AccountName)]
Go to Layout ["Layout Number based on User Name"]
```

Insert From Index

**Purpose**

Pastes a value from the index into a field.

**Format**

```
Insert From Index [Select; <table::field>]
```

Options

- **Select entire contents** replaces the contents of a field. If you do not select this option, Insert From Index inserts the result at the cursor position or at the end of the field's data.

- Select **Go to target field** or click **Specify** to specify the field to paste the index value into. The field you specify may be of any type.

Compatibility

This script step is:

- not supported in web publishing
- not supported in a FileMaker Server scheduled script
**Description**

This script step displays the index for the active field. You can use Insert From Index in Browse or Find modes.

FileMaker Pro displays the View Index dialog box so the user can select an entry. This method promotes consistent data entry and correct spelling. For this script step, you can specify a field and select the entire contents of the field.

FileMaker Pro uses what was last selected for the **Show individual words** option in the View Index dialog box (it is not remembered in the script).

**Notes**

- If the specified field does not exist on the layout where the script is being performed, Insert From Index returns an error code which can be captured with the `Get(LastError)` function.
- If you have turned indexing off for a field and deselected the option to **Automatically turn indexing on if needed**, the index will not display and an error code is returned. For more information on field indexing see [Defining field indexing options](#).

**Examples**

The following example displays the index for a field in Find mode, so the user will select a preexisting value.

Enter Find Mode [ ]
Insert From Index [Table1::User Name Field]
Perform Find [ ]

---

**Insert From Last Visited**

**Purpose**

Pastes information from a field in the last active record into the specified field in the current record or find request.

**Format**

`Insert From Last Visited [Select; <table::field>]`

**Options**

- **Select entire contents** replaces the contents of a field. If you do not use Select Entire Contents, Paste From Last Record inserts the result at the insertion point or at the end of the field’s existing data.
- Select **Go to target field** or click **Specify** to specify the field to paste into.
Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
Use this script step to ensure consistency and automate data entry. The active record is the last record where FileMaker Pro performed some activity, such as pasting text or moving into a field with the Go to Field script step.

Note If the specified field does not exist on the layout where the script is being performed, Insert From Last Visited returns an error code which can be captured with the Get(LastError) function.

Examples
Go to Record/Request/Page [First]
Enter Find Mode []
Go to Field [Company Name]
Insert From Last Visited []

Insert Current Date

Purpose
Pastes the current system date into the specified field.

Format
Insert Current Date [Select; <table::field>]

Options
- Select entire contents replaces the contents of a field with the current date. If you do not select this option, Insert Current Date adds the value of the current date to the end of the field's existing data.
- Select Go to target field or click Specify to specify the field to paste the date into.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Notes
- To perform a calculation with the current date, make sure the receiving field is defined as a date field.
- If the specified field does not exist on the layout where the script is being performed, Insert Current Date returns an error code which can be captured with the Get(LastError) function.
- In a web-published database, use a Commit Record/Request script step after an Insert Current Date script step to update the record in the browser window.
Examples
Go to Record/Request/Page [First]
Enter Find Mode []
Insert Current Date [Select; Invoices::Invoice Date]

Insert Current Time

Purpose
Pastes the current system time into the specified field.

Format
Insert Current Time [Select; <table::field>]

Options
- **Select entire contents** replaces the contents of a field with the current time. If you do not select this option, Insert Current Time adds the value of the current time to the end of the field's existing data.
- Select **Go to target field** or click **Specify** to specify the field to paste into.

Note If the specified field does not exist on the layout where the script is being performed, Insert Current Time returns an error code which can be captured with the `Get(LastError) function`.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Examples
Go to Layout [Call Log::Data Entry]
Insert Current Time [Select; Call Log::Call Time]

Insert Current User Name

Purpose
Pastes the current user name into the specified field in the current record.

Format
Insert Current User Name [Select; <table::field>]

Options
- **Select entire contents** replaces the contents of a field with the current user name. If you do not select this option, Insert Current User Name adds the value of the current user name to the end of the field's existing data.
- Select **Go to target field** or click **Specify** to specify the field to paste into.
Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Notes
• If the specified field does not exist on the layout where the script is being performed, Insert Current User Name returns an error code which can be captured with the Get(LastError) function.
• User name is the FileMaker Pro user, an editable field displayed on the General tab of the Preferences dialog box. To track user access to a database with more security, use the Get(AccountName) function to return the current user’s account name.

Examples
Go to Layout [“Sales”]
Insert Current User Name [Select; Sales::Sales Person]

Insert Picture

Purpose
Imports a graphic from another file into the current container field.

Format
Insert Picture [Select; <table::field>]

Options
• Store only a reference to the file instructs FileMaker Pro to store only a link to the graphic file instead of the entire file. This option may reduce the size of your FileMaker Pro file, but if you move or delete the graphic, FileMaker Pro won’t be able to display it.
• Select Specify source file or click Specify to identify the file that contains the graphic. For more information on creating file paths in FileMaker Pro, see Creating file paths.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script
Notes

• You must specify a field, click in a field, or use the Go to Field script step before performing this script step. If you have not specified a graphic to be imported, FileMaker Pro displays a dialog box, where the user can select the graphic file to import.

• If there is no active container field on the layout where the script is being performed, Insert Picture returns an error code that can be captured with the Get(LastError) function.

Examples
Go to Field [Table1::Photo]
Insert Picture ["My_Photo.tif"]

Insert QuickTime

Purpose
Imports a QuickTime movie or sound file into the current container field.

Format
Insert QuickTime [“<filename>”]

Options
Select Specify source file or click Specify to display the Open dialog box where you can specify the name, file type, and location of the QuickTime file you intend to insert. For more information, see Creating file paths.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Description
You must specify a field, click in a field, or use the Go to Field script step before performing this script step.

When Insert QuickTime is performed, unless a file is specified with the Specify source file option, FileMaker Pro displays a dialog box where the user can select and preview the file to be imported.

Notes

• If there is no active container field on the layout where the script is being performed, Insert QuickTime returns an error code that can be captured with the Get(LastError) function.

• This script step requires that QuickTime extension for Windows be installed.

• For a list of media types that QuickTime supports, see Working with data in container fields.
**Insert Object (Windows)**

**Purpose**
Embeds or links an OLE object in the current container field.

**Format**
Insert Object ["<object type>"]

**Options**
- Click **Specify** to display the Insert Object dialog box where you can set the following options.
  - **Object Type** - choose the type of object you want to embed or link from the list of available file and application types.
  - **Create New** - lets you embed a blank object of the specified object type. When the object is activated, it can then be created with the associated application.
  - **Create from File** - lets you enter the name of an existing file as the object to be embedded or linked. You can use the **Browse** button to select a file that is on the computer or the network.
  - When **Create from File** has been selected, you can select **Link** to create indicate that the object is to be a linked object. When **Link** is not selected, the object is embedded.
  - **Display As Icon** tells FileMaker Pro not to display the embedded or linked object completely, but to display an icon that represents the object. You can use the **Change Icon** button to select a different icon for display. When **Display As Icon** is not selected, the complete object is displayed in the container field.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
If the specified object/file does not exist on the computer the script is being run on (or if it’s run on a Mac OS system), this script step returns an error code that can be captured with the **Get(LastError)** function.

**Examples**
Go to Field [Table1::Field1]
Insert Object [“Video Clip”]
Insert File

**Purpose**
Imports a file or a reference to a file into the current container field.

**Format**
Insert File [Reference; <table::field>; "<filename>”]

**Options**
- **Store only a reference** instructs FileMaker Pro to store only a link to a file in the container field instead of the entire file. This option may reduce the size of your FileMaker Pro file, but if you move or delete the file being referenced, FileMaker Pro won't be able to display it.
- Select **Go to target field** or click **Specify** to specify the container field to insert the file into.
- Select **Specify source file** or click **Specify** to specify the file to be inserted. For information on creating file paths in FileMaker Pro, see [Creating file paths](#).

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
You must specify a target field, click in a field, or use the [Go to Field script step](#) before performing this script step. If you do not specify a file to insert, FileMaker Pro displays a dialog box where the user can select a file.

**Note** If there is no active container field on the layout where the script is being performed, Insert File returns an error code that can be captured with the Get(LastError) function.

**Examples**
Go to Field [Last Correspondence]
Insert File ["My_Letter.doc"]

Update Link (Windows)

**Purpose**
Updates the OLE link (manual or automatic) in the current container field.

**Format**
Update Link [<table::field>]

**Options**
Select **Go to target field** or click **Specify** to specify the field to be updated.
**Compatibility**

This script step is:

- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**

If the field does not contain an OLE link, then this script step does not return an error code.

**Note** In addition to using the Update Link script step to update both automatic and manual links in your database, automatic links are updated when an OLE object is activated with a Go to Field script step with Select/perform on, or when the print option Update all Links before printing is used.

**Examples**

Enter Browse Mode []
Update Link [Table1::Spreadsheet]

---

**Replace Field Contents**

**Purpose**

Uses the value in a specified field in the current record or uses a calculation to replace the value in that field in every record in the current found set.

**Format**

Replace Field Contents [No Dialog; <table::field>;
Current contents/Serial numbers/Calculation results]

**Options**

- **Perform without dialog** prevents display of the Replace Field Contents dialog box when the script step executes.
- Select **Go to target field** or click **Specify** to specify the target field for the replace operation.
- Click **Specify** to display the Replace Field Contents dialog box, where you can determine the settings required for the Replace Field Contents command.
  - **Replace with "current contents"** uses the current value in the specified field as the replacement value to place in that field in every other record in the current found set.
  - **Replace with serial numbers** reserializes the field in every record in the current found set.
  - **Entry options** uses the next available value in Entry Options as the first record number, incrementing with whatever value is in Entry Options.
  - **Custom Values** lets you enter a value to be used as a starting point for the serialization, as well as a value by which to increment each serialized field in the current found set.
  - **Update serial number in Entry Options?** resets the serial number value in Entry Options so that the next serial number that is automatically entered will follow the records you have reserialized with this script step. If this option is not used, the serial
value in Entry Options is not changed, and may not be in sequence with the newly reserialized records.

- If the field to be replaced was set up for auto-entry of a serial number and **Prohibit modification of value** is not selected, FileMaker Pro will still put sequential numbers in the selected field, but will do so starting with the next number to be automatically entered.
- **Replace with calculated result** displays the Specify Calculation dialog box, where you can enter a calculation to be used as the replacement value.

**Compatibility**

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Notes**

- This script step can also be used to reserialize a field in every record in the current found set.
- When you use the Replace Field Contents script step, the data must be committed first before the operation is performed, or you may get unexpected results. For example, if a field value is modified and a replace is attempted before the data is committed, then the replace will be based on the old data rather than the modified data. (For more information about committing data, see Committing data in records.)

**Examples**

Sort Records [Restore]
Go to Record/Request/Page [First]
Replace Field Contents [No dialog; script examples::Serial Number; Serial numbers]

**Relookup Field Contents**

**Purpose**

Copies new values from the lookup source field into the records that make up the current found set.

**Format**

Relookup Field Contents [No dialog; <table::field>]

**Options**

- **Perform without dialog** prevents a dialog box from displaying when the script step executes that lets the user confirm field information.
- Select **Go to target field** or click **Specify** to specify the field that is the target of the relookup operation. FileMaker Pro moves the cursor to the field you specify. This must be the match field for the relationship upon which the lookup is based, not the Lookup source or target field. If no field is selected, Relookup Field Contents returns an error code that can be captured with the **Get(LastError) function.**
**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Notes**
- The target field specified for the Relookup Field Contents step must be an editable field.
- Relookups are only performed on the found set of records in the active table.
- Calculation fields cannot be used for the Relookup Field Contents command. If you use the Relookup Field Contents command, and you're using a calculation field as the matching value, you must specify one of the input fields for the calculation as the field parameter for the Relookup Field Contents.
- When you use the Relookup Field Contents script step, the data must be committed first before the operation is performed, or you may get unexpected results. For example, if a field value is modified and a relookup is attempted before the data is committed, then the relookup search will be based on the old data rather than the modified data. (For more information about committing data, see *Committing data in records*.)

**Examples**
Show All Records
Go to Layout ["Data Entry"]
Go to Field [Table1::Invoice Number]
Set Field [Table1::Invoice Number; Max(Table1::Invoice Number)]
Relookup Field Contents [No dialog, Table1::Customer ID]

**Export Field Contents**

**Purpose**
Exports the contents of a single field in the active record to a new file.

**Format**
Export Field Contents [<table::field>; "<filename>"

**Options**
- Select **Specify target field** or click **Specify** to specify the field to be exported. If no field is specified, FileMaker Pro will export the contents of the current field in the active table.
- Select **Specify output file** or click **Specify** to specify the name and location of the file to which the field contents will be exported. For more information, see *Creating file paths*. If no file is specified, the user is asked to choose a filename and location when the script step is performed. You can choose to **Automatically open file** or **Create email with file as attachment** after saving.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script
Examples

Go to Layout [Images::Nature Scenes]
Export Field Contents [Images::Original Pictures;"Uncropped.jpg"]
Records script steps

Records script steps affect specific records and find requests. Use these script steps to:

- add, delete, and copy records or find requests
- change field contents in all records in the found set
- commit and revert records
- import and export records

New Record/Request

**Purpose**
In Browse mode, creates a new, blank record. In Find mode, creates a new find request.

**Format**
New Record/Request

**Options**
None.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Examples**
Enter Browse Mode []
Go to Layout ["Layout #1"]
New Record/Request
Go to Field [Table1::First Name]

Duplicate Record/Request

**Purpose**
In Browse mode, makes a duplicate of the current record. In Find mode, duplicates the current find request.

**Format**
Duplicate Record/Request

**Options**
None.
Compatibililty
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
If the record has a field set up for automatic entry of values, this script step does not duplicate the value in the field of the current record. In that case, FileMaker Pro generates and enters a new value for the duplicated record.

To use the Duplicate Record/Request script step with portal rows, first use the Go to Portal Row script step, and then use the Duplicate Record/Request step. Using the Duplicate Record/Request step when a portal row is selected will duplicate the selected row in the related table. This option is only available when the option to allow creation of related records has been enabled for the relationship.

Examples
Perform Find [Restore]
Go to Record/Request/Page [Last]
Duplicate Record/Request

Delete Record/Request

Purpose
In Browse mode, deletes the current record. In Find mode, deletes the current find request.

Format
Delete Record/Request [No dialog]

Options
Perform without dialog prevents the deletion confirmation dialog box from displaying when the script step executes.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.
Description

<table>
<thead>
<tr>
<th>To delete</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>A master record</td>
<td>Make sure that the record you want to delete is active (use navigation script steps), and then use Delete Record/Request. If the current layout contains a portal, use Exit Record/Request to make sure that no related record is selected, then use Delete Record/Request.</td>
</tr>
<tr>
<td>A related record</td>
<td>Use the Delete Portal Row script step. If Delete Record/Request executes when a related field in a portal is selected, the user sees a dialog box that asks whether the master record or related record should be deleted. (Note that Allow deletion of portal records must also be selected in the Portal Setup dialog box to enable deletion of related records.)</td>
</tr>
<tr>
<td>A find request</td>
<td>Make sure that the find request you want to delete is active (use navigation script steps), and then use Delete Record/Request.</td>
</tr>
</tbody>
</table>

Important  You cannot undo a Delete Record/Request script step.

Examples

Perform Find [Restore]
Omit
Delete Record/Request [ ]

Delete Portal Row

Purpose
Deletes the selected portal row (which deletes data in a related record).

Format
Delete Portal Row [No dialog]

Options
Perform without dialog prevents a message box from displaying when the script step executes that asks the user to confirm the deletion of the portal row.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
Use the Go to Portal Row script step to control which row is selected. (If no portal row is selected, nothing happens when this script step executes.)

Important  You cannot undo a Delete Portal Row script step.
Examples
The following script selects the first portal row in the current record and then deletes that row with no confirmation dialog box.

Go to Portal Row [First]
Delete Portal Row [No dialog]

Delete All Records

Purpose
Deletes all the records in the current found set.

Format
Delete All Records [No dialog]

Options
Perform without dialog prevents a message box from displaying when the script step executes that asks the user to confirm the deletion of records.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
First, use the Perform Find script step to find the records you want to delete. Then, use this script step to delete them.

Important  You cannot undo a Delete All Records script step.

Examples
Perform Find [Restore]
Omit
Delete All Records
Perform Script ["Data Entry"]

Open Record/Request

Purpose
Makes an existing record or find request available for editing.

Format
Open Record/Request
Options
None.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
When performed on a record, this script step verifies that the user has sufficient access privileges to edit the record. If the user has the proper privilege set, FileMaker Pro attempts to prevent others from simultaneously editing or deleting the record. Once locked, the record is considered "opened" and may be edited.

If an error occurs (for example, if the current user has insufficient access privileges, the record is currently locked by another user, or the record is open in another window), FileMaker Pro generates an error message, which may be captured using the Set Error Capture script step and the Get(LastError) function. (Opening a find request will not return an error, because requests can’t be locked by other users.)

Important Because any attempt to modify a field or a record will also attempt to open the record for editing, in most cases you will not need to use the Open Record/Request script step to open a record explicitly. But when using the Go to Record/Request/Page script step to enter a field or record, use the Open Record/Request script step first to obtain a write lock.

Examples
Perform Find [Restore]
Go to Record/Request/Page [Last]
If [IsEmpty(script examples::Field1)]
  Open Record/Request
End If

Revert Record/Request

Purpose
Returns the current record or request to the way it was before you began adding or changing data in the record.

Format
Revert Record/Request [No dialog]

Options
Perform without dialog prevents a dialog box from displaying when the script step executes that asks the user to confirm the revert action.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.
Examples
Show Custom Dialog ["Commit record or revert changes?";
"Click 'Commit' to save your changes, or 'Revert' to
discard changes to this record."]

#1 = Commit, 2 = Revert

If [Get(LastMessageChoice) = 1]
Commit Records/Requests
Else
  Revert Record/Request [No dialog]
End If

Commit Records/Requests

Purpose
Commits a record.

Format
Commit Records/Requests [No dialog]

Options
Skip data entry validation overrides any data entry validation options set for fields and commits the
record anyway. This option only skips validation for fields set with the Only during data entry
validation option in the Options for Field dialog box; fields set to Always validate will still validate,
even if the Skip data entry validation option is selected. For more information on field validation,
see Defining field validation.

Perform without dialog prevents a dialog box from displaying when the script step executes that
asks the user to confirm the commit action.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
This script step exits the current record or find request, updating field data and making no field
active.
Examples

Show Custom Dialog ["Commit record or revert changes?"; "Click 'Commit' to save your changes, or 'Revert' to discard changes to this record."]

#1 = Commit, 2 = Revert

If [Get(LastMessageChoice) = 1]
    Commit Records/Requests
Else
    Revert Record/Request [No dialog]
End If

Copy Record/Request

Purpose
Copies the values of a record or find request to the Clipboard.

Format
Copy Record/Request

Options
None.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
Text fields are copied in tab-delimited format, without styles, in the order they appear in the record layout. Text in container fields is also copied. (This script step is equivalent to choosing Copy from the Edit menu when there is no active field.)

Note When using the Copy Record/Request step with repeating fields, FileMaker Pro inserts the group separator character between each repetition. You can use a text editor to replace these characters with another delimiter such as a tab or space.

Examples
This script finds and sorts a set of records, switches to the Data Entry layout, goes to the last record, copies the entire record, switches layouts and tables, creates a new record, and pastes the contents of the Clipboard to the History field.

Perform Find [Restore]
Sort Records [Restore; No dialog]
Go to Layout ["Data entry"]
Go to Record/Request/Page [Last]
Copy Record/Request
Go to Layout ["History table"]
New Record/Request
Copy All Records/Requests

**Purpose**
In Browse mode, copies the values of all records to the Clipboard. In Find mode, copies all find requests.

**Format**
Copy All Records/Requests

**Options**
None.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
Text fields are copied in tab-delimited format, without styles, in the order they appear in the record layout. Text in container fields is also copied. (This script step is equivalent to choosing Copy from the Edit menu while holding down the Shift key (Windows) or Option key (Mac OS) when there is no active field.)

**Note** When using the Copy All Records/Requests step with repeating fields, FileMaker Pro inserts the group separator character between each repetition. You can use a text editor to replace these characters with another delimiter such as a tab or space.

**Examples**
This script finds and sorts a set of records, switches to the Data Entry layout, copies the contents of the found set to the Clipboard, switches layouts and tables, creates a new record, and pastes the contents of the Clipboard to the History field.

Perform Find [Restore]
Sort Records [Restore; No dialog]
Go to Layout ["Data entry"]
Copy All Records/Requests
Go to Layout ["History table"]
New Record/Request
Paste [Select; History table::History]
Commit Records/Requests[]
Import Records

**Purpose**
Imports records from another file or data source you specify.

**Format**
Import Records [No dialog; "<source or filename>"; Add/Update existing/ Update matching; <platform and character set>]

**Options**
- **Perform without dialog** prevents display of import-related dialog boxes when the script step executes. However, if a data source file has not been specified, the Open File dialog box will be displayed when a script is run from FileMaker Pro.
- **Specify data source** lets you choose the file or source of the data to be imported. Depending on the file or source you choose, a dialog box may appear for specifying the following additional options:

<table>
<thead>
<tr>
<th>When you choose this file or source</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>In the Specify File dialog box, choose the file you want to import, or type file paths directly into the list. Specify one path per line. FileMaker Pro will use the first path it locates as the file to import. See <a href="#">Creating file paths</a>.</td>
</tr>
<tr>
<td>Folder</td>
<td>In the Folder of Files Import Options dialog box, choose the folder of image or text files that you want to import, or type the folder paths directly into the list. Specify one path per line. FileMaker Pro will use the first path it locates as the folder to import. For more information, see <a href="#">Importing a folder of files all at once</a> and <a href="#">Creating file paths</a>. This option is not supported from scripts running on FileMaker Server.</td>
</tr>
<tr>
<td>Digital Camera (Mac OS)</td>
<td>In the FileMaker Pro Photo Import Options dialog box, choose your camera or other image device, and choose how you want to download and import the photos. For more information, see <a href="#">Importing photos from a digital camera or other device (Mac OS)</a>. This option is not supported from scripts running on FileMaker Server.</td>
</tr>
<tr>
<td>Bento Data (Mac OS)</td>
<td>In the Select Bento Source dialog box, select the library and collection from that library that you want to import, then select field mapping options. For more information, see <a href="#">Importing Bento data (Mac OS)</a>. This option is not supported from scripts running on FileMaker Server.</td>
</tr>
<tr>
<td>XML Data</td>
<td>In the Specify XML and XSL Options dialog box, choose the source of the XML data that you want to import, and choose an XSLT style sheet if you want to apply one prior to import. The XML and XSLT source may be a file or the result of an HTTP request, or a calculation that generates a file path or an HTTP request. For more information, see <a href="#">Importing XML data</a>.</td>
</tr>
</tbody>
</table>
Records script steps

### When you choose this file or source

| ODBC Data | Specify the data source name and location, the user ID and password (if any), and the SQL query to be executed. You can enter a query directly or specify a calculation to generate the query. For more information, see [Automating ODBC import using the Import Records script step](#).  
  
  **Note** If you use the Import Records script step to import ODBC data that contains Unicode strings, your ODBC driver must support Unicode. Otherwise, the results may be inconsistent.

**• Specify import order** tells FileMaker Pro to use a predefined import order. The last import order used in the file appears as the default and can be edited. This option also lets you choose whether to keep repeating field data together or to split repeating fields into separate records, as well as whether to add new records, replace data in the current found set, or import data as a new table.  

**Note** When import source fields and target fields are mapped using matching names, field name matching is performed dynamically each time the script step is performed.

### Compatibility

This script step is:

- not supported in web publishing  
- also supported in a FileMaker Server scheduled script

### Description

You can set the import order and use the **Specify import order** option, or perform this script step with the dialog box so the user can set new criteria, such as importing data as a new table.  

If the source file is open, the found set is imported; if not, all records in the source table are imported.

When you use this script step to import an XML data file:

- your file path must be the absolute path to your data file  
- you can use variables in the file path  
- use different syntax for Windows and Mac OS. For Windows, follow the syntax `file:///\[full path to file\]`. For Mac OS, follow the syntax `file:///volumes/\[full path to file\]`. If you use a variable in the file path, it must generate the proper syntax.

When you include this script step in a FileMaker Server scheduled script and you do not select **Perform without dialog** in the Edit Script dialog box, the text (**NOT compatible**) appears after the script step. The script, however, will run. The behavior will be the same as if **Perform without dialog** was selected. See [Get(DocumentsPathListing) function](#) for more information.

If you schedule a FileMaker Server script to do any of the following, you will receive an error:

- importing records to new table  
- importing records from a folder  
- importing images to a container field
Records script steps

Notes

• When using the Import Records script step or Export Records script step in a FileMaker Server scheduled script, keep the following in mind:
  
  • Any specified file must be in the FileMaker Server Documents folder, the temporary folder, or a child folder of either the FileMaker Server Documents folder or the temporary folder. For example, the following are all valid paths for file.csv:

    <Documents>/file.csv
    <Temporary Path>/file.csv
    <Documents>/Folder1/file.csv
    <Temporary Path>/Folder1/Folder2/Folder3/Folder4/file.csv

  • Any path specified that isn’t a complete path to the file (for example, anything other than /Library/FileMaker Server/Data/Documents/<0 or more directories>/<filename> ) is evaluated as being relative to the temporary path.
  
  • Any paths that include “..” are considered invalid.

  • For FileMaker Pro, if an absolute path is not specified in a script that will be executed from FileMaker Pro, the path is assumed to be relative to the location of the database file from which the script was run. For example, if a script containing the Export Records script step is run with the pathname file:/export.tab, and the file running the script is /MyFiles/Library/Books.fp7, the exported file will be created as /MyFiles/Library/export.tab.

Examples

Go to Layout ["Layout #4"]
Import Records [No dialog; "Contacts"; Add; Mac Roman]

Export Records

Purpose

Exports records to a specified file.

Format

Export records [No dialog; "<output filename>"; Automatically open; Create email; <platform and character set>]

Options

• Perform without dialog prevents dialog boxes from displaying when the script step is run from FileMaker Pro. These dialog boxes would let the user set new export criteria.

• Select Specify output file or click Specify to display a dialog box where you can specify the file and file type to export. Choose the folder you want to export to, or type the folder path directly into the list. Specify one path per line. FileMaker Pro will use the first path it locates. See Creating file paths. The file type you use depends on the import requirements of the program using the exported data. You can also specify options to Automatically open the file and Create email with the file as attachment after saving.

Note You can export records as a Microsoft Excel or Excel 2007 worksheet by choosing Excel 95-2004 Workbooks (.xls) or Excel Workbooks (.xlsx) from the File Type list. For more
Records script steps

Information on saving FileMaker Pro files as Excel files, see Saving and sending records as an Excel file.

If you specify XML as the export file type, the Specify XML and XSL Options dialog box appears, where you can specify an XML grammar and choose an XSLT style sheet if you want to transform the XML. The XSLT source may be a file, the result of an HTTP request, or a calculation that generates a file path or an HTTP request.

**Note** Do not export fields whose name is entirely numeric (such as "2") or whose name is a single-byte kana character (Japanese version) using the FMPDSORESULT grammar.

- Select Specify export order or click Specify to use the export order that was in effect when you added the script step. The last export order used in the file appears as the default and can be edited or deleted. If necessary, choose an output file character set from the list.
- Select Apply current layout's data formatting to exported data, or the formatting of the last exported data will be used.

### Compatibility

This script step is:

- not supported in web publishing
- also supported in a FileMaker Server scheduled script (except exporting to the .fp7 file type)

### Description

You can set the export order before adding this script step, or perform the step with a dialog box so a user can set new criteria. Export Records exports all the records currently being browsed. If you want to limit the set of records you're exporting, use a find request before using Export Records.

When you include this script step in a FileMaker Server scheduled script and you do not select Perform without dialog in the Edit Script dialog box, the text (NOT compatible) appears after the script step. The script, however, will run. The behavior will be the same as if Perform without dialog was selected. See Get(DocumentsPathListing) function for more information.

### Notes

- When using the Export Records script step or Import Records script step in a FileMaker Server scheduled script, keep the following in mind:
  - Any specified file must be in the FileMaker Server Documents folder, the temporary folder, or a child folder of either the FileMaker Server Documents folder or the temporary folder. For example, the following are all valid paths for file.csv:
    ```
    <Documents>/file.csv
    <Temporary Path>/file.csv
    <Documents>/Folder1/file.csv
    <Temporary Path>/Folder1/Folder2/Folder3/Folder4/file.csv
    ```
  - Any path specified that isn't a complete path to the file (for example, anything other than /Library/FileMaker Server/Data/Documents/<0 or more directories>/<filename> ) is evaluated as being relative to the temporary path.
  - Any paths that include ".." are considered invalid.
  - For FileMaker Pro, if an absolute path is not specified in a script that will be executed from FileMaker Pro, the path is assumed to be relative to the location of the database file from which the script was run. For example, if a script containing the Export Records script step
is run with the pathname file:/export.tab, and the file running the script is /MyFiles/Library/Books.fp7, the exported file will be created as /MyFiles/Library/export.tab.

**Examples**

Go to Layout ["Layout #4"]
Export Records [No dialog, "Contacts"]

**Save Records As Excel**

**Purpose**
Saves records to a specified Excel or Excel 2007/2008 worksheet.

**Format**
Save Records As Excel [No dialog; “<output filename>”; Automatically open; Create email; Current record]

**Note** In order to save records as a Microsoft Excel worksheet, your privilege set must include Allow exporting, or you must select Run script with full access privileges in the Edit Script dialog box.

**Options**
- **Perform without dialog** prevents the Excel Options dialog box from displaying as the script step executes if you have already specified a file. If you have not specified a file, the Save Records as Excel dialog box displays as the script step executes, but the Excel Options dialog box is not displayed.
- **Select Specify output file** or click Specify to display a dialog box where you can specify the file path. Choose the folder you want to export to, or type the file path directly into the list. Specify one path per line. FileMaker Pro will use the first path it locates. See Creating file paths. You can choose to Automatically open file or choose to Create email with file as attachment to create a blank email with the Excel file as an attachment.
- **Select Specify options** or click Specify to display the “Save Records as Excel” Options dialog box. From the Save list, choose Records being browsed or Current record. Specify whether the values in the first row should be used as field names or as data. For **Worksheet**, **Title**, **Subject**, and **Author**, you can enter text directly, or click Specify to enter a field name or values from a calculation.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
This script step operates in all modes except Find mode.
Examples

Go to Layout ["Layout #4"]
Save Records As Excel [No dialog; "Contacts.xls"]

Save Records As PDF

Purpose
Saves records to a specified PDF file.

Format
Save Records As PDF [Append; No dialog; "<output filename>"; Automatically open; Create email; User specified options]

Note In order to save records as a PDF file, your privilege set must include Allow printing, or you must select Run script with full access privileges in the Edit Script dialog box.

Options

• Select Append to existing PDF to append the records being browsed, the current record, or a blank record after the last page of the specified PDF file. When you append records, the PDF Options dialog box settings in the Document and Initial View tabs are ignored, but the settings in the Security tab are maintained.

• Perform without dialog prevents dialog boxes from displaying when the script step executes if a file has already been specified.

• Select Specify output file or click Specify to display a dialog box where you can specify the file path. Choose the folder you want to save to, or type the file path directly into the list. Specify one path per line. FileMaker Pro will use the first path it locates. See Creating file paths. You can choose to Automatically open file or you can choose to Create email with file as attachment after saving.

• Select Specify options or click Specify to display the "Save Records as PDF" dialog box. From the Save list, choose Records being browsed, Current record, or Blank record. If you select Blank record, the Appearance option is enabled.

Select Options to display the PDF Options dialog box.

• In the Document tab, you can specify descriptive information for the PDF file. For each of the options in the document tab, you can enter text directly, or click Specify to enter a field name or values from a calculation.

• In the Security tab, you can assign passwords to the PDF file, as well as print and edit privileges. If print and edit privileges are allowed, you can specify if copying and screen reading software are permitted.

• In the Initial View tab, you can define the initial view for the layout and magnification for the PDF file.
Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script
• not supported in runtime solutions

Description
This script step operates in all modes except Find mode.

Note (Mac OS) For runtime solutions, you can choose to print records as a PDF file, and the script will be saved with the printed file.

Examples
Go to Layout "Layout #4"
Save Records As PDF [No dialog, "Contacts.pdf"; Records being browsed]

The following adds the records being browsed after the last page of the existing Contacts.pdf file.

Go to Layout "Layout #4"
Save Records As PDF [Append; No dialog, "Contacts.pdf"; Records being browsed]
Found Sets script steps

With Found Sets script steps, you can:

• specify a find request to find records
• extend or constrain the found set
• modify the most recent find request
• show all records
• omit records from the found set
• sort and unsort records

Perform Find

Purpose

Enters Find mode and finds records that match one or more find requests that you set up, that are stored with the script step.

Format

Perform Find [Restore]

Options

Select Specify find requests or click Specify to create and manage find requests. The requests you create are stored with the script step.

• New opens the Edit Find Request dialog box, where you define criteria for a find request.
• Edit opens a selected find request from the list.
• Duplicate duplicates one or more selected find requests from the list.
• Delete deletes one or more selected find requests from the list.

The Edit Find Request dialog box allows you to create or edit find request criteria.

• For Action, select Find Records or Omit Records to specify whether this find request will find or omit records. Finding records adds them to your found set. Omitting records excludes them. An individual request can find or omit records; use multiple requests if you need to find and omit records during the same Perform Find script step.

• Find records when (or Omit records when) shows a list of the fields in your current table. To construct a find request, begin by selecting a field from this list.
  • To select a field from a related table, click the name of the current table at the top of the list and select the related table you want. Select a related field from this new list.
  • Change the value in Repetition to specify a particular cell of a repeating field.
• Type your search criteria for the selected field in the Criteria area.
  • Click Insert Operator to further refine your search criteria. See Finding records.
  • Click Add to add your criteria to the find request.
Found Sets script steps

- To change existing criteria, select the line containing the field and criteria from the top of the dialog box, and make your changes to field and/or criteria. Click Change to store your changes.
- To delete existing criteria, select the line containing the field and criteria from the top of the dialog box and click Remove.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
If you do not create a find request, this script step will perform the last find request that was executed.

You can use a Perform Find script following an Enter Find Mode script step, to pause the script and allow the user to enter find criteria before performing a find.

Examples
If FileMaker Pro doesn't find any records that match the find criteria when you perform the script, you can stop the script, continue the script with zero records in the current found set, or change the find criteria. By using the Set Error Capture script step and the Get(LastError) function, you can set up a script to handle such situations. For example:
The following script tries to perform a find. If no records are found, a custom dialog will appear giving the user the option to modify the find request or go back to Browse mode.

Set Error Capture [On]
Perform Find [Restore]
If [Get (LastError) > 0]
    Show Custom Dialog ["No records were found. Click OK to modify your request or click Cancel to return to Browse mode.""]
    If [Get (LastMessageChoice) = 1]
        Modify Last Find
    Else
        Enter Browse Mode []
    End If
End If

Constrain Found Set

Purpose
Narrows the existing found set using the criteria you specify in the stored find request.

Format
Constrain Found Set [Restore]

Options
Select Specify find requests or click Specify to create and store a find request with the script step.
Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
This script step is equivalent to a logical AND search.

Note  If no records match the find criteria, this script step returns a found set of zero records.

Examples
Go to Layout ["Data entry"]
Constrain Found Set [Restore]
Sort Records [Restore; No dialog]

Extend Found Set

Purpose
Broadens the existing found set using the criteria in the stored find request.

Format
Extend Found Set [Restore]

Options
Select Specify find requests or click Specify to create and store a find request with the script step.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
This script step is equivalent to a logical OR search.

Examples
Go to Layout ["Data entry"]
Extend Found Set [Restore]
Sort Records [Restore; No dialog]

Modify Last Find

Purpose
Changes the most recent find request.

Format
Modify Last Find
**Options**
None.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
Add a [Perform Find script step] after this script step to find records.

**Examples**
Show Custom Dialog ["Repeat the last Find?"]
If [Get (LastMessageChoice) = 1]
  Modify Last Find
  Pause/Resume Script [Indefinitely]
  Perform Find[]
Else
  Enter Find Mode [Pause]
  Perform Find[]
End If

**Show All Records**

**Purpose**
Finds all the records in the table underlying the foreground window and leaves you in the current record.

**Format**
Show All Records

**Options**
None.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
Use this script step to redisplay all the records after working with a found set.
Use Show All Records in Browse mode or Preview mode. If you perform this script step in Find mode or Layout mode, FileMaker Pro switches to Browse mode after the records have been found.

**Options**
None.
**Examples**

Go to Layout ["Address Layout"]
Show All Records

**Show Omitted Only**

**Purpose**
Finds the records not in the current found set.

**Format**
Show Omitted Only

**Options**
None.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
Use this script step in Browse mode or Preview mode. If you perform this script step in Find mode or Layout mode, FileMaker Pro switches to Browse mode after the records have been found.

**Examples**
Show All Records
Omit Record
Show Omitted Only
Enter Preview Mode [Pause]

**Omit Record**

**Purpose**
Omits the current record from the found set and places you on the next record in the table.

**Format**
Omit Record

**Options**
None.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.
Description
This script step temporarily hides the current record.

Omit Record only operates in Browse mode and Find mode. When this script step is performed in Find mode, it causes the find request to become an Omit request, as if the Omit checkbox had been selected.

Examples
Show All Records
Omit Record
Show Omitted
Enter Preview Mode [Pause]

Omit Multiple Records

Purpose
Omits several records from the found set, starting with the current record, and places you on the next record in the table.

Format
Omit Multiple Records [No dialog; <number of records>]

Options
- Perform without dialog prevents a dialog box from displaying when the script step executes; the dialog lets the user enter the number of records to be omitted.
  When Perform without dialog is selected, if you do not specify the number of records, only the current record is omitted.
- Select Specify records or click Specify to enter the exact number of records you want to omit. You can also click Specify in the Options dialog box and enter a calculation. The calculation result must be a number.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
This script step temporarily hides the omitted records. If this script step is performed from Layout mode, FileMaker Pro switches to Browse mode after the script step has been performed.

Examples
Perform Find [Restore]
Omit Multiple Records [No Dialog, 3]
Sort Records

**Purpose**
Sorts the records in the current found set according to specified criteria.

**Format**
Sort Records [Restore; No dialog]

**Options**
- **Perform without dialog** prevents display of a dialog box when the script step executes that lets the user enter a different set of sort instructions.
- **Select Specify sort order** or click Specify to create a sort order and store it with the script step. When Specify sort order is not selected, FileMaker Pro uses the most recently executed sort instructions.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
Find the records you want to sort (using Perform Find script step or Show All Records script step) before using this script step. If you sort a repeating field, FileMaker Pro sorts on only the first entry in that field. You don't need to use the Enter Browse Mode script step or Go to Record/Request/Page script step after using Sort Records.

Each Sort Records script step can store a unique sort order; the sorting instructions are stored with the step, not the script. You can use this step several times within the same script and store a different sort order for each occurrence.

**Note** Items in the sort order that aren't valid when the script step is performed are ignored. When you specify a sort order in a database containing multiple tables, FileMaker Pro stores the table name for each sort field in the sort order. For example, a sort order using the Last Name field of the Contacts table is stored in the sort order as Contacts::Last Name. If the Sort Records script step is performed when a table other than Contacts is the active table, Contacts::Last Name cannot be evaluated, and is ignored in the sort order.

**Examples**
Go to Layout ["My sort layout"]
Show All Records
Sort Records [Restore; No dialog]
Print [ ]
Unsort Records

**Purpose**
Restores the records in the current found set to their creation order (the order in which they were entered in the file).

**Format**
Unsort Records

**Options**
None.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
Use this script step when it's important to maintain the original creation order, as with invoices entered by invoice number or membership records entered by date.

**Note** You don't need to perform an Unsort Records script step before executing a [Sort Records script step](#).

**Examples**
Sort Records [Restore; No dialog]
Go to Layout ["Sales Summary"]
Enter Preview Mode [Pause]
Go to Layout [original layout]
Enter Browse Mode []
Unsort Records
Go to Record/Request/Page [First]
Windows script steps

Windows script steps affect screen elements and windows. You can:

- open or close a window
- select a window
- arrange windows
- move or resize a window
- update or freeze a window
- scroll a window
- show or hide a window, the status toolbar, or the text ruler
- set the title of a window
- set the zoom level
- view data as a list, table, or form

New Window

**Purpose**

Creates a new window based on the foreground window.

**Format**

```
New Window[Name: <name of window>; Height: n; Width: n; Top: n; Left: n]
```

**Options**

Click Specify to set options for the new window.

- **Window Name** is the name you specify for your new window. You can enter literal text or click Specify to create a window name from a calculation.
- **Height** is the height of the new window in pixels. You can enter a number or click Specify to generate a number from a calculation.
- **Width** is the width of the new window in pixels. You can enter a number or click Specify to generate a number from a calculation.
- **Distance from top** is the new window’s distance in pixels from the top of the screen (Mac OS) or top of the FileMaker Pro window (Windows). You can enter a number or click Specify to generate a number from a calculation.
- **Distance from left** is the new window’s distance in pixels from the left of the screen (Mac OS) or left of the FileMaker Pro window (Windows). You can enter a number or click Specify to generate a number from a calculation.

**Compatibility**

This script step is also supported in web publishing and in a FileMaker Server scheduled script.
Description
The new window has the same layout, same table, same found set, and same current record as the original window.

Notes
- Window names created with the New Window script step do not have to be unique.
- Window name selection is not case-sensitive.
- It is not necessary to enter values for each option. When no value is entered, FileMaker Pro uses the default value associated with the Window menu > New Window command.
- Minimum height and minimum width depend on the operating system of the user performing the script. If the height or width you specify are less than the minimum, FileMaker Pro uses the minimum values.
- If the window height or width values you specify exceed the user’s screen resolution, FileMaker Pro uses the maximum values possible.
- You can create a new window off-screen by supplying negative top and/or left values, which can be useful for multiple monitor environments.
- If you select Show Compatibility, the string (VIRTUAL WINDOW ON WEB) is added to the script step display to indicate to the user that a new virtual window is opened in the same browser window.

Examples
Go to Layout ["Data Entry"]
New Window [Name: “Enter data here”; Height: 400; Width: 600; Top: 16; Left: 16]

Select Window

Purpose
Specifies a window by name and makes it the foreground window.

Format
Select Window [Current window or Name:<name of window>; Current file]

Options
- Click Current Window to bring the active window of the file that contains the script to the foreground.
- Click Specify to select which window you want FileMaker Pro to bring to the foreground. You can enter literal text or click Specify to generate a name from a calculation.
- Select Current file only to restrict matches to the current file (not selecting this option matches all available FileMaker Pro files).
Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
Because FileMaker Pro script steps are always performed in the foreground table, it is sometimes necessary to bring a specific window to the front. Use this script step when you are working with scripts in multi-table files to make certain that a script step is performed in the intended table.

Notes
• Window name selection is not case-sensitive. This script will select the first matching window and bring it to the front.
• The Select Window script step does not open a window of a related file when the related file is opened in a hidden state, such as when a file is opened because it is the source file of a related field. To use the Select Window script step with this type of related file, open the related file using the Open File script step.

For example, a layout in the file Data Entry contains a related field from the file Companies. When this layout is displayed, FileMaker Pro opens the Companies file in a hidden state. To open a new window displaying the hidden related file, use a script such as:

Open File[Open Hidden; "Companies"]
Select Window["Companies"]

To return the file to its original state, perform a Close Window [] script step to close the open window.

Examples
Perform Find [Restore]
Sort Records [Restore]
Select Window [Name: “Sales records”]

Close Window

Purpose
Closes the currently active window or any other window by name.

Format
Close Window [Current window or Name: <name of window>; Current file]

Options
• Click Specify to select which window you want FileMaker Pro to close. You can enter literal text or click Specify to generate a name from a calculation.
• Select Current file only to restrict matches to the current file (not selecting this option matches all available FileMaker Pro files).

Note Closings the last window of a file closes the file and halts execution of the current script.
**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Examples**
Perform Find [Restore]
Sort Records [Restore]
Pause/Resume Script [Indefinitely]
Close Window [Name: "Sales records"]

**Adjust Window**

**Purpose**
Hides or changes the size of a window.

**Format**
Adjust Window [Resize to fit/Maximize/Minimize/Restore/Hide]

**Options**
Use Specify to choose an adjustment option.
- **Resize to fit** resizes the window to the minimum size possible while keeping all items in the layout visible.
- **Maximize** resizes the window to full-screen size.
- **Minimize** reduces the window to an icon on the status bar (Windows) or Dock (Mac OS)
- **Restore** returns the window to its previous size.
- **Hide** hides the current database window from view. This option is equivalent to the Hide Window menu command

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Examples**
Go to Layout ["Data Entry"]
Adjust Window [Maximize]
Move/Resize Window

**Purpose**
Adjusts the size or location of the chosen window.

**Format**
Move/Resize Window [Current window or Name: <name of window>; Current file; Height: <n>; Width: <n>; Top: <n>; Left: <n>]

**Options**
Click Specify to set the move/resize options.
- **Current Window** selects the current foreground window when the script step is performed.
- **Window Name** selects an open window by name. You can enter literal text or click Specify to create a window name from a calculation.
- Select **Current file only** to restrict matches to the current file (not selecting this option matches all available FileMaker Pro files).
- **Height** is the height of the adjusted window in pixels. You can enter a number or click Specify to generate a number from a calculation.
- **Width** is the width of the adjusted window in pixels. You can enter a number or click Specify to generate a number from a calculation.
- **Distance from top** is the adjusted window’s distance in pixels from the top of the screen (Mac OS) or top of the FileMaker Pro window (Windows). You can enter a number or click Specify to generate a number from a calculation.
- **Distance from left** is the adjusted window’s distance in pixels from the left of the screen (Mac OS) or left of the FileMaker Pro window (Windows). You can enter a number or click Specify to generate a number from a calculation.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
The adjusted window maintains the same layout, same table, same found set, and same current record as the original window.

**Notes**
- Window name selection is not case-sensitive.
- It is not necessary to enter values for each option. When no value is entered, FileMaker Pro uses the original location and size of the window as default values.
- Minimum window height and width depend on the script user’s operating system. If the height and width values you specify are less than the minimum, FileMaker Pro uses the minimum values.
• Maximum window height and width depend on the script user’s screen resolution. If the height or width values you specify are greater than the maximum, FileMaker Pro uses the maximum value possible.
• You can move a window off-screen by supplying negative top and/or left values, which can be useful for multiple monitor environments.
• Windows: FileMaker Pro orients the moved window to the top left corner of the visible part of the application window. Note that this may not be the (0,0) point of the window, depending on how the current file window is positioned (for example, if half of the file window extends past the left border of the application window, you would need to scroll to the left to see the (0,0) point of the application window).

Examples
Go to Layout ["List of Members"]
Move/Resize Window [Current Window; Height: 400; Width: 600; Top: 16; Left: 16]

Arrange All Windows

Purpose
Adjusts the size and location of all open windows.

Format
Arrange All Windows [Tile Horizontally/Tile Vertically/Cascade Window/Bring All to Front]

Options
• Tile Horizontally positions open windows in a left/right orientation. Windows are resized to prevent overlapping.
• Tile Vertically positions open windows in a top/bottom orientation. Windows are resized to prevent overlapping.
• Cascade Window positions windows in an overlapping pattern, beginning in the top left corner of the screen. Windows are resized to fill the screen, less any offset.
• Bring All to Front (Mac OS) moves all open FileMaker Pro windows to the front. Windows are not resized.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Description
The size and location of the open windows are the only things that change; current table, active window, and active record are not affected by this script step.
Freeze Window

**Purpose**

Stops updating the active window.

**Format**

Freeze Window

**Options**

None.

**Compatibility**

This script step is:

- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**

Use Freeze Window to hide actions from the user while FileMaker Pro performs the remainder of the script. Use Refresh Window after Freeze Window to resume updating the window.

**Note** A Refresh Window step is not needed if it would be the last step in a script. FileMaker Pro automatically turns off Freeze Window and refreshes the display when a script ends. Use Refresh Window to force a window update within a script.

**Examples**

Freeze Window
Go to Record/Request/Page [First]
Loop
  Set Field [Table1::Salary; Table1::Salary * 1.1]
  Go to Record/Request/Page [Next; Exit after last]
End Loop
Refresh Window

**Purpose**
Updates the entire contents of the FileMaker Pro document window, including any related records.

**Format**
Refresh Window

**Options**
Select **Flush cached join results** to delete the results of queries for related records and cause related records to be refreshed. Do not select this option if you know your script does not affect related data, and if you want to minimize the performance impact of re-accessing related data (particularly when sharing a database over a network).

Select **Flush cached SQL data** to delete the results of queries for related ODBC data source records and cause related ODBC records to be refreshed. FileMaker Pro dumps the internal cache and refreshes record data. Do not select this option if you know your script does not access ODBC data.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
You do not need to select **Flush cached join results** in order to refresh the entire window.

This script step updates records (rows) in the current ODBC table.

**Examples**
Go to Record/Request/Page [First]
Loop
    Set Field [Table1::Salary; Table1::Salary * 1.1]
    Go to Record/Request/Page [Next; Exit after last]
End Loop
Refresh Window
Beep

Scroll Window

**Purpose**
Scrrolls a window up or down, scrolls to the top or bottom of a layout, or brings the current field into view.

**Format**
Scroll Window [Home/End/Page Up/Page Down/To Selection]
Options

Use **Specify** to choose a scrolling option.

- **Home, End, Page Up, or Page Down** scrolls the window to the beginning, to the end, up a page, or down a page.
- **To Selection** brings the current field into view (similar to tabbing into a field).

Compatibility

This script step is:

- not supported in web publishing
- not supported in a FileMaker Server scheduled script

Examples

Go to Record/Request/Page [First]
Loop
  Set Field [Table1::Salary; Table1::Salary * 1.1]
  Go to Record/Request/Page [Next; Exit after last]
End Loop
Scroll Window [Home]

Show/Hide Status Area

Purpose

Shows or hides the status toolbar.

Format

Show/Hide Status Area [Lock; Show/Hide/Toggle]

Options

- **Lock** prohibits the user from using the status toolbar control to manually show or hide the status toolbar.
- **Show** tells FileMaker Pro to show the status toolbar.
- **Hide** tells FileMaker Pro to hide the status toolbar.
- **Toggle** switches between showing and hiding the status toolbar (equivalent to clicking Show/Hide Status Toolbar).

Compatibility

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Examples

Go to Layout ["Contacts"]
Toggle Status Area [Hide]
Show/Hide Text Ruler

**Purpose**
Shows or hides the text ruler.

**Format**
Show/Hide Text Ruler [Show/Hide/Toggle]

**Options**
- **Show** tells FileMaker Pro to show the text ruler.
- **Hide** tells FileMaker Pro to hide the text ruler.
- **Toggle** switches between showing and hiding the text ruler.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
Choosing the **Toggle** option switches the current state of the ruler. The **Text Ruler** is used to format paragraphs and tabs within fields and on layouts.

**Examples**
Go to Layout ["Notes"]
Toggle Text Ruler [Show]

Set Window Title

**Purpose**
Changes the title of a window.

**Format**
Set Window Title [Current window or Name: <name of window>; Current file; New Title: <new window name>]

**Options**
Click **Specify** to set options for the title of a window.
- **Window to Rename** tells FileMaker Pro which window to rename. Select **Current Window** to rename the current window. To specify another window, enter the window name in literal text or click **Specify** to create a name using a calculation.
- Select **Current file only** to restrict matches to the current file (not selecting this option matches all available FileMaker Pro files).
• **Rename window to** is the new title for the window. You can enter literal text or click **Specify** to create a name using a calculation.

**Compatibility**

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**

You can change the name of any open window.

**Note**  Window name selection is not case-sensitive.

**Examples**

Perform Find [Restore]
Set Window Title [Current Window; New Title: "Find Results"]

---

Set **Zoom Level**

**Purpose**

Enlarges or reduces the image on the screen.

**Format**

Set Zoom Level [Lock; 25%...400%/Zoom In/Zoom Out]

**Options**

• **Lock** prohibits users from making changes to the zoom level.

• **Specify** lets you select a zoom level.
  • Reduction values: **100%**, **75%**, **50%**, or **25%**.
  • Enlargement values: **150%**, **200%**, **300%**, or **400%**.
  • **Zoom In** reduces the screen image by one zoom level.
  • **Zoom Out** enlarges the screen image by one zoom level.

**Compatibility**

This script step is:

• not supported in web publishing

• not supported in a FileMaker Server scheduled script

**Examples**

Go to Layout ["Data entry"]
Set Zoom Level [Lock; 100%]
View As

**Purpose**
Displays data in the specified format.

**Format**
View As [View as Form/View as List/View as Table/Cycle]

**Options**
- **View as Form** tells FileMaker Pro to display records page by page in the format determined by the database designer.
- **View as List** tells FileMaker Pro to display records as records in a list, so they can be browsed without clicking the left and right arrows in the book.
- **View as Table** tells FileMaker Pro to display the records on screen in a grid, so that many fields and records may be viewed at one time in a format similar to a spreadsheet.
- **Cycle** switches from the current view to the next view.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
You can choose **View as Form, View as List, View as Table**, or you can choose **Cycle** to switch from the current format to the next format.

**Examples**
Perform Find [Restore]
View As [View as Form]
Pause/Resume Script [Indefinitely]
View As [View as List]
Pause/Resume Script [Indefinitely]
View As [View as Table]
Pause/Resume Script [Indefinitely]
View As [Cycle]
Files script steps

Files script steps operate on entire files. With Files script steps, you can:

• create a file
• open or close a file
• save a copy of a file
• convert a file to FileMaker Pro
• set multi-user status
• recover a file
• set print setup options
• print

New File

Purpose
Allows the user to create a new database file.

Format
New File

Options
None.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script
• not supported in runtime solutions

Description
If the general preference for Show FileMaker Quick Start Screen is selected, this script step opens the FileMaker Quick Start Screen. Otherwise, this script step opens the “Create a new file named:” dialog box. After the user exits the Manage Database dialog box, the script becomes active again, and if script steps remain, the script continues. The new database remains open, but not active.

Examples
New File
Open File

**Purpose**

Opens the specified FileMaker file or allows the user to select a file to open.

**Format**

Open File [Open hidden; "<filename>”]

**Options**

- **Open hidden** causes FileMaker Pro to open and hide the specified database.
- **Specify** lets you select a FileMaker Pro database or ODBC data source to open. Choose Add FileMaker Data Source or Add ODBC Data Source to locate and select a file. After you select a file, it is added to the Specify Table list. Choose Manage Data Sources to modify or delete a data source you've already added to the list.

**Compatibility**

This script step is:

- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**

For ODBC data sources, this script step re-establishes a link to the ODBC data source. Script steps after Open execute in the FileMaker file that contains the script, not the file opened by the script.

If you don't specify a file, FileMaker Pro displays the Open File dialog box when the Open File script step executes so that the user can specify a file. The Open File dialog box will also be displayed if the file used in the script has been moved, deleted, or renamed.

**Notes**

- If new database creation has been disabled in your installation of FileMaker Pro, it will not be possible to convert file types such as Microsoft Excel or tab delimited files into FileMaker Pro databases. However, you can still import the data from other file types into FileMaker Pro (if you open the database with a password that allows you to import). If you need more information, contact your system administrator.
- In runtime solutions, the Open File script step returns an error if an external file is not bound to the solution.

**Examples**

If [DayName(Get(CurrentDate))="Monday"]
    Open ["Weekly Planner"]
Else
    Open ["Daily Planner"]
End If
Close File

**Purpose**
Closes the specified FileMaker file.

**Format**
Close File [Current File/"<filename>"]

**Options**
*Specify* lets you select a FileMaker Pro to close or an ODBC data source to disconnect from. Choose *Add FileMaker Data Source* or *Add ODBC Data Source* to locate and select a file. After you select a file, it is added to the *Specify Table* list. Choose *Manage Data Sources* to modify or delete a data source you've already added to the list.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
If you do not specify a file, FileMaker Pro stops the currently running script and closes the FileMaker database that the script is running from.

This script step disconnects the specified ODBC data source. This enables you to “re-login” to the specified ODBC data source with a different user name and password, without closing the FileMaker file.

**Note** If a user moves the specified file, FileMaker Pro might not be able to close the file.

**Examples**
Close ["Names"]
Close ["Payments"]
Close ["Tasks"]
Close [Current File]

Convert File

**Purpose**
Converts a supported file type into a FileMaker Pro file.

**Format**
Convert File ["<filename>"]
Options

- **Specify data source** lets you choose the file or source of the data to be converted. If you don’t specify a file, FileMaker Pro displays the Open File dialog box when the script step executes. For more information about converting files, see the conversion information in the FileMaker Pro User’s Guide.

- Depending on the file or source you choose, a dialog box may appear for specifying the following additional options.

<table>
<thead>
<tr>
<th>When you choose this file or source</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>In the Specify File dialog box, choose the file you want to import. See <a href="#">Creating file paths</a>.</td>
</tr>
<tr>
<td>XML Data</td>
<td>In the Specify XML and XSL Options dialog box, choose the source of the XML data that you want to import, and choose an XSLT style sheet if you want to apply one prior to import. The XML and XSLT source may be a file or the result of an HTTP request, or a field that contains a file path or an HTTP request. For more information, see Importing XML data.</td>
</tr>
<tr>
<td>ODBC Data</td>
<td>Specify the data source name and location, the user ID and password (if any), and the SQL query to be executed. For more information, see <a href="#">Constructing an SQL query for importing via ODBC</a>.</td>
</tr>
</tbody>
</table>

- **Perform without dialog** prevents certain dialog boxes from displaying when the script step is run.
  - When converting FileMaker files, the dialog boxes used to rename the old file and name the converted file are not displayed.
  - When converting other file types, the dialog boxes that are used to specify the way a file and its data are converted are not displayed.

  In these two cases, the default settings for those dialog boxes are used instead.

Compatibility

This script step is:

- not supported in web publishing
- not supported in a FileMaker Server scheduled script

Description

See [Supported import/export file formats](#) for more information.

**Note** You cannot use this command to convert the contents of a folder. Each file in a folder must be converted separately.

Examples

Convert File [“mydata.fp5”]
Set Multi-User

**Purpose**
Allows or disallows network access to a database.

**Format**
Set Multi-User [On/On (Hidden)/Off]

**Options**
- Select **On** to allow network access via FileMaker Network Sharing. This is the same as selecting **All Users** in the FileMaker Network Settings dialog box.
- Select **On (Hidden)** to allow network access but prevent the name of the shared database from appearing in the Open Remote File dialog box. This is the same as selecting the **All Users** and **Don't Display in Open Remote File dialog** options in the FileMaker Network Settings dialog box.
- Select **Off** to disallow network access. This is the same as selecting **No Users** in the FileMaker Network Settings dialog box.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script
- not supported in runtime solutions

**Description**
A shared, or multi-user, database is a file available to users on a network. See [Sharing and hosting files](#) for more information.

**Examples**
This script could be a startup script to guarantee that a database is hosted upon launch.

Show Custom Dialog ["Make this file available on the network?"]
If [Get (LastMessageChoice) = 1]
    Set Multi-User [On]
End If

**Tip** The `Get(MultiUserState)` function could be used to verify the resulting multi-user setting. Get(MultiUserState) is set to 0 for a single-user file and 1 for a multi-user file.
Set Use System Formats

**Purpose**
Allows the user to choose between the formats stored with the file or the user's current system formats.

**Format**
Set Use System Formats [On/Off]

**Options**
- **On** instructs FileMaker Pro to use the current system formats.
- **Off** instructs FileMaker Pro to use the formats saved with the file.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
Each FileMaker Pro database file stores the date, time, and number formatting of the computer on which the database was created. These regional settings may be different from the system formats currently set on your computer.

When you use the current system formats, you don't change the formats saved with the file — you simply enter and view data in a format that's familiar to you. For example, if a database was created in Australia, where dates are usually written day-month-year, and you open it in the United States, where dates are usually written month-day-year, FileMaker Pro alerts you that the system formats are different.

A startup script that uses the Set Use System Formats script step can be used to control how FileMaker Pro handles date, time, and number formats for a file that has system formats that are different from the current system formats.

**Examples**
This script could be a startup script that checks the system language before using system formats.

```plaintext
If [Get (SystemLanguage) = "Japanese"]
    Set Use System Formats [On]
End If
```
Save a Copy as

**Purpose**
Saves a copy of the current database file.

**Format**
Save a Copy as ["<filename>"; copy/compacted/clone]

**Options**
- **Specify output file** displays the Specify Output File dialog box, where you can specify the name and location of the resulting copy. For more information, see Creating file paths. If you do not specify a file, FileMaker Pro saves a copy of the current file and displays the Save As dialog box so the user can specify copying options. You can choose to **Automatically open file** or **Create email with file as attachment** after saving.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
Use this script step to back up your database.

**Note** This script step is not supported in FileMaker Pro database files hosted by FileMaker Server.

**Examples**
Save a Copy as ["backup"]

---

Recover File

**Purpose**
Recovers a damaged FileMaker Pro file.

**Format**
Recover File [No dialog; "<filename>"]

**Options**
- **Perform without dialog** prevents a dialog box from displaying after the script step performs that shows how many bytes of data were recovered, the number of records and field values skipped, and the number of field definitions recovered.
Perform without dialog applies only to default recovery operations. The Advanced Recover Options dialog box is not supported.

- Select Specify source file or click Specify to display a dialog box where you can select the file to be recovered. For more information, see Creating file paths. If you don’t select a source file, the Open Damaged File dialog box displays at runtime.

**Compatibility**

This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**

FileMaker Pro attempts to repair and recover as much of the information in a damaged file as possible, and then creates a new, recovered file. The original file is not deleted or replaced.

**Note** Before you begin, be sure you have enough disk space for the recovered file. If there isn’t enough space, the recovered file won’t be usable.

**Examples**

#Note: The Recover command makes an aggressive attempt to reopen a damaged file. It is intended for data recovery, not file repair. Do not use the Recover command as part of routine file maintenance.

Recover File []

**Print Setup**

**Purpose**

Sets print options, such as paper size and orientation, which can be stored with this script step.

**Format**

Print Setup [Restore; No dialog]

**Options**

- **Perform without dialog** prevents the Print Setup dialog box from displaying when the script step executes. If you select Perform without dialog, the output from this script step will be sent to the last specified printer and not the one specified in the Edit Script dialog box.
- Select Specify page setup or click Specify to open the Print Setup dialog box and choose page setup options that are stored with the script step.

**Compatibility**

This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script
Description

You can also have this script step performed with a dialog box so the user can specify print options. You can use multiple Print Setup script steps within a script.

Examples

Enter Browse Mode []
Go to Layout ["Layout #1"]
Print Setup [Restore; No dialog]
Show All Records
Sort Records [Restore; No dialog]
Print []

Print

Purpose

Prints information in a file.

Format

Print [Restore; No dialog]

Options

• Select Perform without dialog to prevent the Print dialog box from displaying when the script step executes.

• Select Specify print options or click Specify to specify an output destination. When the script executes, the output from this script step is sent to the printer or fax you specified.

Mac OS: As an alternative to specifying a printer or fax, you can:

• Click PDF in the Print dialog box, then choose a PDF option from the list or choose Edit menu to define a custom PDF output format or destination

• Click Preview in the Print dialog box to specify that the output of the script opens in the user’s default print preview application

If you do not specify an output destination before saving the script or if the specified printer cannot be found when the script executes, the output from the script step is sent to the user’s default printer.

Compatibility

This script step is:

• not supported in web publishing

• not supported in a FileMaker Server scheduled script
Description

Multiple print steps can be added to a script. You can store print options with this script step, or allow the user to enter printing criteria when the step executes.

If the print job needs special page setup options to print correctly, add a Print Setup script step before the Print step.

Notes

• When you use more than one Print script step in a script, the saved printer name applies to only the containing Print script step.

• A printer name specified in a Print Setup script step is not inherited by any Print script step in a script.

• Mac OS: If a FileMaker Pro 9 (or earlier) file contains a script with Print Script [restore; no dialog], the output goes to the default OS printer when you run the script in FileMaker Pro 10. You can recreate print scripts in FileMaker Pro 10 to change the printer.

Examples

Enter Browse Mode []
Go to Layout ["Layout #1"]
Print Setup [Restore; No dialog]
Show All Records
Sort Records [Restore; No dialog]
Print []
Accounts script steps

Accounts script steps allow you to create and manage user accounts. With Accounts script steps, you can:

- add accounts
- delete accounts
- reset account passwords
- change passwords
- enable accounts
- log in to a file using a different account and password

Notes

- Accounts created using script steps cannot be granted full access privileges (only accounts created manually).
- Users who are not assigned the Full Access privilege set can only execute account scripts if Run script with full access privileges is selected in the Edit Script dialog box when the script is created or edited. For more information on access privileges and script steps, see Editing scripts privileges.

Add Account

**Purpose**

Adds a new account name, password, and privilege set.

**Format**

Add Account [Account Name: <account name>; Password: <password>; Privilege Set: "<privilege set>"; Expire password]

**Options**

Click Specify to display the “Add Account” Options dialog box, where you can set the following options.

- **Account Name** is the name for the new account. You can enter literal text or click Specify to create a new account name from a calculation.
- **Password** is the password for the new account. You can enter literal text or click Specify to create a new password from a calculation.
- **Privilege Set** lets you assign a predefined privilege set for this user or create a new privilege set.
  The Full Access privilege set cannot be assigned via this script step. Accounts with Full Access privilege sets must be created manually.
- **User must change password on next login** When selected, this option forces users to change their password the next time they log in to the database.
Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
The account name and password can be literals stored with this script step, or generated at runtime based on calculations that you set up.

Notes
- Account names must be unique.
- You must be assigned the Full Access privilege set to perform this script step. Select Run script with full access privileges to enable users with less than full access privileges to perform this script step.

Examples
Add Account [Account Name: "MyAccount"; Password: "MyPassword"; Privilege Set: "[Data Entry Only]"; Expire password]

Delete Account

Purpose
Deletes the specified account.

Format
Delete Account [Account Name: <account name>]

Options
Click Specify to set the Account Name to be deleted. You can enter literal text or click Specify to create an account name from a calculation.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
- You must specify an existing account.
- You must be assigned the Full Access privilege set to perform this script step. Select Run script with full access privileges to enable users with less than full access privileges to perform this script step.
- You cannot use this script step to delete an account that has full access privileges.
- This script step deletes an account without displaying a dialog box.
Examples
Delete Account [Account Name: "Guest Access account"]

Reset Account Password

Purpose
Resets the password of the existing FileMaker Pro account you specify.

Format
Reset Account Password [Account Name: <account name>; New Password: <password>; Expire password]

Options
- Click Specify to display the “Reset Account Password” Options dialog box, where you can set the following options.
- Account Name is the name of the existing FileMaker Pro account with the password to be reset. You can enter literal text or click Specify to create a new account name from a calculation.
- New Password is the new password for this account. You can enter literal text or click Specify to create a new password from a calculation.
- User must change password on next login When selected, this option forces users to change their password the next time they log in to the database.

Compatibility
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description
Use multiple occurrences of this script step to reset multiple accounts in a single script.

Notes
- You must be assigned the Full Access privilege set to perform this script step. Select Run script with full access privileges to enable users with less than full access privileges to perform this script step.
- This script step resets the account password without displaying a dialog box.

Examples
Reset Account Password [Account Name: "Guest Account"; New Password: "MyPassword"; Expire password]
Change Password

**Purpose**
Changes the password for the current account.

**Format**
Change Password [Old Password: <old password>; New Password: <new password>; No dialog]

**Options**
Click *Specify* to display the “Change Password” Options dialog box, where you can set the following options.

- **Old Password** is the old password for the current account. You can enter literal text or click *Specify* to generate the password from a calculation.
- **New Password** is the new password for the current account. You can enter literal text or click *Specify* to create a new password from a calculation.
- **Perform without dialog** prevents the Change Password dialog box from displaying when the script step executes. When this option is selected, FileMaker Pro uses the literal password information stored with the script step, or generates a new password based on a calculation.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**
This script step displays the Change Password dialog box, unless *Perform without dialog* is selected. Change Password allows you to change your password but not your access privileges.

**Notes**
- Users must have password change privileges to use this script step to change their passwords. To enable users who do not have password change privileges to run this script step, select *Run script with full access privileges*.
- Unless the *Set Error Capture script step* has enabled error capture, users get five attempts to change their password.
- If Set Error Capture is on, users get a single attempt to enter their old and new password.
### Examples

The following, used as a startup script with a password-protected database, presents the user with the Change Password dialog box every other time the database is opened (to encourage the user to change his or her password frequently). The gOpenCount field is a global number field that records how many times the database has been opened.

```plaintext
Allow User Abort[Off]
Set Field [Table1::gOpenCount; Table1::gOpenCount + 1]
If [Int(Table1::gOpenCount/2)*2 < > Table1::gOpenCount]
   Change Password []
End If
```

---

#### Enable Account

**Purpose**

Enables or disables a specific account.

**Format**

Enable Account [Account Name: <account name>; Activate/Deactivate]

**Options**

Click **Specify** to display the “Enable Account” Options dialog box, where you can set the following options.

- **Account Name** is the name of the account to be activated or deactivated. You can enter literal text or click **Specify** to generate the account name from a calculation.
- **Activate account** enables the specified account.
- **Deactivate account** disables the specified account.

**Compatibility**

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**

- You must specify an existing account.
- You must be assigned the Full Access privilege set to perform this script step. Select **Run script with full access privileges** to enable users with less than full access privileges to perform this script step.
- You cannot use this script step to deactivate an account with full access privileges.

**Examples**

Enable Account [Account Name:"MyAccount"; Activate]
Re-Login

**Purpose**

Allows the user to log in to the database using a different account and password without closing and reopening the file.

**Format**

Re-Login [Account Name: <account name>; Password: <password>; No dialog]

**Options**

*Perform without dialog* prevents the Open <filename> dialog box from displaying when the script step executes. This dialog box requires the user to manually enter an account and password (or edit information already displayed in the dialog box) to open a database file. When *Perform without dialog* is selected, FileMaker Pro uses the account and password information that is stored with the script step, or generates information from calculations when the step executes.

Click **Specify** to display the "Re-Login" Options dialog box, where you can set the following options.

- **Account Name** is the name of the account to be authenticated. You can enter literal text or click **Specify** to create a new account name from a calculation.
- **Password** is the password for this account. You can enter literal text or click **Specify** to create a new password from a calculation.

**Compatibility**

This script step is supported in web publishing and in a FileMaker Server scheduled script, but only if *Perform without dialog* is selected.

**Description**

Privileges assigned to the new account take effect immediately, including access to tables, records, layouts, scripts, and value lists.

**Notes**

- You do not need full access privileges to perform this script step. Users with any level of access can use this script step to re-login to the database.

- Users get five attempts to enter their account and password, unless the *Set Error Capture script step* is enabled.

- If the Set Error Capture script step is enabled, users get a single attempt to enter their account and password.

**Examples**

Re-Login [Account Name:"MyAccount"; Password:"MyPassword"; No dialog]
Spelling script steps

With spelling script steps you can:

• check the spelling in a selection of text
• check the spelling of a record
• check the spelling of an entire found set
• correct a word
• set spelling options for a file
• select a dictionary
• edit the user dictionary

Check Selection

**Purpose**
Uses the spelling checker to check the selected text.

**Format**
Check Selection [Select; table::field]

**Options**

• **Select entire contents** checks all the text in the active field. If you do not use **Select entire contents**, you must select some text before this script step executes.

• Select **Go to target field** or click **Specify** to specify a field to be checked.

**Compatibility**
This script step is:

• not supported in web publishing
• not supported in a FileMaker Server scheduled script

**Examples**
Check Selection [Select; Table1::Balance Due Letter]

Check Record

**Purpose**
Uses the spelling checker to check the contents of every field in the current record.

**Format**
Check Record
Options
None.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Examples
Go to Record/Request/Page [First]
Loop
  Check Record
  Go to Record/Request/Page [Next; Exit after last]
End Loop

Check Found Set

Purpose
Uses the spelling checker to check the contents of every field in the records being browsed.

Format
Check Found Set

Options
None.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Examples
Perform Find [Restore]
Check Found Set
Correct Word

**Purpose**
Opens the Spelling dialog box so you can correct a word that FileMaker Pro has identified as misspelled.

**Format**
Correct Word

**Options**
None.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
This script step is the same as choosing Edit menu > Spelling > Correct Word.
Use this script step to open the Spelling dialog box for users if you have restricted their access to FileMaker Pro menus.

**Notes**
- Check spelling as you type must be selected (File menu > File Options > Spelling tab.)
- This script step can only correct a word after FileMaker Pro identifies it as being misspelled.

**Examples**
The following script displays a message box that asks if the user wants to open the Spelling dialog box, and opens it if the response is Yes.

Show Custom Dialog ["Open Spelling dialog box to correct spelling?"]
If [Get (LastMessageChoice) = 1]
    #1=Yes, 2=No
    Correct Word
End If
Spelling Options

**Purpose**
Opens the File Options dialog box to the **Spelling** tab.

**Format**
Spelling Options

**Options**
None.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
This script step is the same as choosing File menu > File Options > Spelling tab.
Use this script step to open the File Options dialog box for users if you have restricted their access to FileMaker Pro menus.

**Examples**
The following script displays a message box that asks if the user wants to turn on the **Check spelling as you type** option, and opens the File Options dialog box if the response is **Yes**.

Show Custom Dialog ["Would you like to turn on the "Check spelling as you type" option?"]
If [Get (LastMessageChoice) = 1] #1=Yes, 2=No
    Spelling Options
End If

Select Dictionaries

**Purpose**
Opens the Select Dictionaries dialog box.

**Format**
Select Dictionaries

**Options**
None.
Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Description
This script step is the same as choosing Edit menu > Spelling > Select Dictionaries.
Use this script step to open the Select Dictionaries dialog box for users if you have restricted their access to FileMaker Pro menus.

Examples
Select Dictionaries

Edit User Dictionary

Purpose
Opens the User Dictionary dialog box.

Format
Edit User Dictionary

Options
None.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Description
This script step is the same as choosing Edit menu > Spelling > Edit User Dictionary.
Use this script step to open the Edit User Dictionary dialog box for users if you have restricted their access to FileMaker Pro menus.
Examples

The following script displays a message box that asks if the user wants to open the Edit User Dictionary dialog box, and opens it if the response is Yes.

Show Custom Dialog ["Would you like to edit the user dictionary used for spell checking?"]
If [Get (LastMessageChoice) = 1]
    #1=Yes, 2=No
    Edit User Dictionary
End If
Open Menu Item script steps

Open Menu Item script steps open specific FileMaker Pro dialog boxes as if you had chosen the command from a menu. You could use one of these script steps to open a dialog box for users if you have restricted their access to FileMaker Pro menus with access privileges. With these script steps you can:

- open the Preferences dialog box
- open the Edit Saved Finds dialog box
- open the File Options dialog box
- open the Manage Database dialog box
- open the Manage External Data Sources dialog box
- open the Manage Value Lists dialog box
- open the Find/Replace dialog box
- open Help
- open the Open Remote File dialog box
- open the Manage Scripts feature
- open the FileMaker Network Settings dialog box

Open Preferences

**Purpose**

Opens the Preferences dialog box to the **General** preferences area.

**Format**

Open Preferences

**Options**

None.

**Compatibility**

This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**

This script step is the same as choosing **Edit menu > Preferences** (Windows) or **FileMaker Pro menu > Preferences** (Mac OS).

Use this script step to open the Preferences dialog box for users if you have restricted their access to FileMaker Pro menus.
Examples
The following script displays a message box that asks if the user wants to open the Preferences dialog box, and opens it if the response is Yes.

Show Custom Dialog ["Open Preferences dialog box?"]
If [Get (LastMessageChoice) = 1]
    #1=Yes, 2=No
    Open Preferences
End If

Open Edit Saved Finds

Purpose
Opens the Edit Saved Finds dialog box, where the user can add or change saved finds.

Format
Open Edit Saved Finds

Options
None.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Description
Use this script step to open the Edit Saved Finds dialog box for users if you have restricted their access to FileMaker Pro menus.

Examples
The following script displays a message box that asks if the user wants to open the Edit Saved Finds dialog box, and opens it if the response is Yes.

Show Custom Dialog ["Do you want to change a saved find?"]
If [Get (LastMessageChoice) = 1]
    #1=Yes, 2=No
    Open Edit Saved Finds
End If
Open File Options

**Purpose**
Opens the File Options dialog box to the **General** preferences area.

**Format**
Open File Options

**Options**
None.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script
- not supported in runtime solutions except for spell checking (the Spelling tab opens)

**Description**
This script step is the same as choosing File menu > File Options.
Use this script step to open the File Options dialog box for users if you have restricted their access to FileMaker Pro menus.

**Examples**
The following script displays a message box that asks if the user wants to open the File Options dialog box, and opens it if the response is Yes.

Show Custom Dialog ["Open File Options dialog box?"]
If [Get (LastMessageChoice) = 1]
  #1=Yes, 2=No
  Open File Options
End If

Open Manage Database

**Purpose**
Opens the Manage Database dialog box, where the user can create or edit tables, fields, and relationships.

**Format**
Open Manage Database

**Options**
None.
Compatibility
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script
- not supported in runtime solutions

Description
This script step is the same as choosing File menu > Manage > Database.

Notes
- The user’s account must be assigned the Full Access privilege set, or this script step will not be performed. (Select Run script with full access privileges to enable the script for all users.)
- See Sharing databases on a network for information about making schema changes to shared databases.

Examples
The following script displays a message box that asks if the user wants to create a field, and opens the Manage Database dialog box if the response is Yes.

Show Custom Dialog ["Do you want to create or edit a field?"
If [Get (LastMessageChoice) = 1]
   #1=Yes, 2=No
   Open Manage Database
End If

Open Manage Data Sources

Purpose
Opens the Manage External Data Sources dialog box, where the user can create, edit, or delete external FileMaker or ODBC data sources.

Format
Open Manage Data Sources

Options
None.

Compatibility
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script
- not supported in runtime solutions
Description
This script step is the same as choosing File menu > Manage > External Data Sources.

Notes
• The user’s account must be assigned the Full Access privilege set, or this script step will not be performed. (Select Run script with full access privileges to enable the script for all users.)
• See Sharing databases on a network for information about making schema changes to shared databases.

Examples
The following script displays a message box that asks if the user wants to create or edit a data source, and opens the Manage External Data Sources dialog box if the response is Yes.

Show Custom Dialog ["Do you want to create or edit a data source?"]
If [Get (LastMessageChoice) = 1]
   #1=Yes, 2=No
   Open Manage Data Sources
End If

Open Manage Value Lists

Purpose
Opens the Manage Value Lists dialog box, where the user can manage new or edit existing value lists.

Format
Open Manage Value Lists

Options
None.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script
• not supported in runtime solutions

Description
This script step is the same as choosing File menu > Manage > Value Lists.
Use this script step to open the Manage Value Lists dialog box for users if you have restricted their access to FileMaker Pro menus.
Notes

- The user's account must be assigned the Full Access privilege set, or this script step will not be performed. (Select Run script with full access privileges to enable the script for all users.)
- See Sharing databases on a network for information about making schema changes to shared databases.

Examples

The following script displays a message box that asks if the user wants to create or edit a value list, and opens the Manage Value Lists dialog box if the response is Yes.

Show Custom Dialog ["Do you want to create or edit a value list?"]
If [Get (LastMessageChoice) = 1]
    #1=Yes, 2=No
    Open Manage Value Lists
End If

Open Find/Replace

Purpose

Opens the Find/Replace dialog box.

Format

Open Find/Replace

Options

None.

Compatibility

This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

Description

This script step is the same as choosing Edit menu > Find/Replace > Find/Replace.

Use this script step to open the dialog box for users if you have restricted their access to FileMaker Pro menus.
Examples

The following script displays a message box that asks if the user wants to open the Find/Replace dialog box, and opens it if the response is Yes.

Show Custom Dialog ["Open the Find/Replace dialog box?"]
If [Get (LastMessageChoice) = 1]
    #1=Yes, 2=No
    Open Find/Replace
End If

Open Help

Purpose
Opens FileMaker Pro Help to the Help contents screen.

Format
Open Help

Options
None.

Compatibility
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script
- not supported in runtime solutions

Description
This script step is the same as choosing Help menu > FileMaker Pro Help.

Examples

The following script displays a message box that asks if the user wants to see the onscreen Help, and opens it if the response is Yes.

Show Custom Dialog ["Do you want to open onscreen Help?"]
If [Get (LastMessageChoice) = 1]
    #1=Yes, 2=No
    Open Help
End If
Open Remote

**Purpose**
Opens the Open Remote dialog box.

**Format**
Open Remote

**Options**
None.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script
- not supported in runtime solutions

**Description**
This script step is the same as choosing File menu > Open Remote.
The Open Remote dialog box allows you to select and open a shared FileMaker Pro database over a network connection.

**Examples**
The following script displays a message box that asks if the user wants to open a shared database, and opens the Open Remote dialog box if the response is Yes.

Show Custom Dialog ["Do you want to open a shared database?"]
If [Get (LastMessageChoice) = 1]
  #1=Yes, 2=No
  Open Remote
End If

Open Manage Scripts

**Purpose**
Displays the Manage Scripts dialog box.

**Format**
Open Manage Scripts

**Options**
None.
Compatibility

This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script
- not supported in runtime solutions

Description

This script step is the same as choosing Scripts menu > Manage Scripts, or File menu > Manage > Scripts.

Note Once an Open Manage Scripts step is performed, FileMaker Pro halts execution of the current script. This prevents unexpected conditions from occurring if the currently running script is edited.

Examples

The following script displays a message box that asks if the user wants to create or edit a script, and opens the Manage Scripts dialog box if the response is Yes.

Show Custom Dialog ["Do you want to create or edit a script?"]
If [Get (LastMessageChoice) = 1] #1=Yes, 2=No
   Open Manage Scripts
End If

Open Sharing

Purpose

Opens the FileMaker Network Settings dialog box, where users can set up network database sharing.

Format

Open Sharing

Options

None.

Compatibility

This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script
- not supported in runtime solutions
Description

This script step is the same as choosing File menu > Sharing > FileMaker Network.

Use this script step to open the FileMaker Network Settings dialog box for users if you have restricted their access to FileMaker Pro menus.

Note  The user’s account must be assigned a privilege set that permits changes to sharing settings, or this script step will not be performed. (Select Run script with full access privileges to enable the script for all users.)

Examples

The following script displays a message box that asks if the user wants to share the current database, and opens the FileMaker Network Settings dialog box if the response is Yes.

Show Custom Dialog ["Do you want to share your current database?"]
If [Get (LastMessageChoice) = 1] #1=Yes, 2=No
  Open Sharing
End If
Miscellaneous script steps

Miscellaneous script steps allow you to script miscellaneous actions, like:

- displaying a dialog box that presents the user with different options
- playing the system alert sound
- launching a web browser and displaying a specified URL
- sending internet email
- sending a DDE command (Windows) or an event to another application
- exiting the application

**Tip** Use the Comment script step to annotate your scripts so your colleagues can understand them.

Show Custom Dialog

**Purpose**
Displays a custom message dialog box, with custom text and labels.

**Format**
Show Custom Dialog [<title>; <message text>; Table1::input field 1;...]

**Options**
Click **Specify** to display the “Show Custom Dialog” Options dialog box, where you can set the dialog box title, message text, and buttons, and specify up to three fields to use for input or display.

**General options**
- **Title** lets you specify the title of the custom dialog box. You can enter literal text or click **Specify** to create the dialog box title from a calculation.
- **Message** lets you specify the message of the dialog box. You can enter literal text or click **Specify** to create the message text from a calculation.
- **Button Labels** let you specify how many buttons (up to three) to display in the custom dialog box and labels for these buttons. If you leave a button label blank, the button does not appear in the custom dialog box. If you leave all button titles blank, an OK button displays in the lower-right corner of the custom dialog box.

**Input Field options**
- Select **Show input field <n>** to activate an input field.
- Select **Specify** to choose the field for input. Each input area maps to one field.
- Select **Use password character (*)** to mask text as it is entered, or as it is displayed from the database. This option obscures data being input into the custom dialog box or being displayed, but does not alter the actual data as it is stored in the database.
- Use **Label** to specify a field label (the text that will identify this input to the user.) You can enter literal text or create the label from a calculation.
Compatibility

This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Description

The custom message dialog box can take user input for up to three fields in the database, and display data from up to three fields in the database. The script pauses as long as the custom dialog box stays on the screen. Fields used for input can be of type text, number, date, time, timestamp, or container. Your custom dialog box can also have up to three buttons, with custom button titles.

Use the Get(LastMessageChoice) function to determine which button the user presses.
• 1 represents the rightmost button
• 2 represents the middle (or second) button
• 3 represents the leftmost button

Button 1, the default or rightmost button, is the only button that will write information from the input fields to a file.

Notes

• If values entered into input fields don’t match the field type, a validation error message displays. The user must resolve validation errors before the dialog box can be closed. See Defining field validation for more information.
• The fields you specify don’t need to appear on the current layout. Show Custom Dialog input fields are independent of layouts, similar to the Set Field script step.
• Data can’t be inserted into calculation or summary fields.
• As with Set Field script step, Show Custom Dialog will bypass the Allow entry into field field formatting option.
• Data entry via the Show Custom Dialog script step is limited by any access privileges criteria that may be in place. (Select Run script with full access privileges to enable the script for all users.)
• (Windows) You can create a keyboard shortcut for a custom dialog box button by placing an ampersand before the shortcut key letter in the button label. For example, to create a keyboard shortcut ‘D’ (Alt+D) for a button labeled ‘Done’, type the label ‘&Done’.
**Example 1**

The following performs a search using a custom dialog box. The dialog box asks users to enter a customer name and city. The dialog box shows a custom title, custom text, and two input fields.

```plaintext
#This script begins by entering Find mode.
Enter Find Mode []
The custom dialog box solicits the name and city of the customer to be found.
Show Custom Dialog ["Find a customer"; "Enter the name and city of the customer below:"; Customers::Name; Customers::City]
#In this example, button 1 is "OK" and button 2 is "Cancel."
If [Get (LastMessageChoice) = 1]
    #If the user selects button 1, the Find is performed.
    Perform Find [Restore]
End If

**Example 2**

The following creates a new record and enters data via a custom dialog box. The dialog box asks users to enter a product ID, product description, and product cost. The dialog box shows a custom title, custom text, and three input fields. If the user cancels the dialog box, the record is deleted.

```plaintext
#In this example, you must create a new record before you can enter data in it.
New Record/Request
#The custom dialog box solicits information from the user.
Show Custom Dialog ["Product information"; "Enter identifying information for this product:"; Products::ProductID; Products::Product description; Products::Product cost]
#After the user exits the custom dialog box, the script evaluates which button the user has selected.
#In this example, button 1 is "OK" and button 2 is "Cancel."
If [Get (LastMessageChoice) = 2]
    #If the user cancels the script, the record created at the beginning of the script is deleted.
    Delete Record/Request [No dialog]
End If
```

**Allow Toolbars**

**Purpose**

Shows or hides the formatting bar.

**Format**

Allow Toolbars [<On/Off>]
Options

- **On** indicates that the formatting bar, the View menu > Formatting Bar menu item, and the Formatting Bar button in the layout bar are enabled.
- **Off** hides the formatting bar, disables the View menu > Formatting Bar menu item, and disables the Formatting Bar button in the layout bar.

Compatibility

This script step is:

- not supported in web publishing
- not supported in a FileMaker Server scheduled script

Description

Use this script step to disable the formatting bar and menu items related to the formatting bar. The formatting bar settings are only affected when the file that calls this script step is active.

In runtime solutions created with the FileMaker Pro Advanced software, the formatting bar is hidden in Kiosk mode. Allow Toolbars has no effect in Kiosk mode.

Examples

Allow Toolbars [Off]

Beep

Purpose

Plays a system beep sound.

Format

Beep

Options

None.

Compatibility

This script step is:

- not supported in web publishing
- not supported in a FileMaker Server scheduled script
Examples

Set Error Capture [On]
Perform Find [Restore]
If [Get (LastError) <> 0]
    Beep
    Show Custom Dialog ["Couldn't find the record..."]
End If

Speak (Mac OS)

Purpose

Produces speech from text.

Format

Speak [<text to be spoken>]

Options

Click Specify to display the “Speak” Options dialog box, where you can set the following options.

- Type the text to be spoken directly in the text entry area, or click Specify to create your spoken text from a calculation.
- Use Voice lets you select from the various voices available on your computer.
- Wait for speech completion before continuing tells FileMaker Pro to wait until the speech is completed before continuing with the next script step.

Compatibility

This script step is:

- not supported in web publishing
- not supported in a FileMaker Server scheduled script

Description

You can enter a literal text string or use a calculation to create the text to be spoken. You can also specify which voice synthesizer to use and whether or not to wait for the speech to be completed, before continuing with the next script step.

If the database is opened on a non-speaking computer, the script can still be edited, but only the default voice synthesizer is available. Speak script steps are not executed when the script is run on a non-speaking computer.
Examples
The following script checks a database to see if all records have phone numbers entered, audibly notifies the user when it finds one missing, and waits so the user can enter one if desired.

Go to Record/Request/Page [First]
Loop
  If [IsEmpty(Contacts::PhoneNumberField)]
    Speak ["Phone number is missing"]
    Pause/Resume Script [Indefinitely]
  End If
  Go to Record/Request/Page [Next; Exit after last]
End Loop

Dial Phone

Purpose
Dials a phone from within a script.

Format
Dial Phone [No dialog; <phone number>]

Options
• **Perform without dialog** prevents the Dial Phone dialog box from displaying when the script step executes.
• **Click Specify** to display the “Dial Phone” Options dialog box where you can set the following options.
  • **Phone Number** lets you enter a phone number to dial.
  • **Specify** lets you create a calculation to generate the phone number.
  • **Use Dialing Preferences** (if NDISWAN TAPI is not installed) tells FileMaker Pro to use the current phone dialing preferences, based on your location. These preferences remove, insert, and append digits to phone numbers, as when a dialing prefix is required in a business setting. This option can apply to phone numbers provided from a field value or to a number that the user enters. (If TAPI is installed, modem and dialing preferences are set in the Phone Dialer accessory application.)

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script
**Description**

You can enter a phone number or specify a calculation to create a phone number. You can also choose whether to use the current phone preferences based on your current location. You can specify number, text, calculation, or global fields, and letters within the phone number are translated to numbers (except for q and z).

**Important** The Dial Phone script step is not supported in Mac OS X.

**Examples**

The following script repeatedly dials the phone number in the PhoneNumber field, up to ten times.

Set Field [Contacts::gCount; 0]
Loop
  Dial Phone [No dialog; Contacts::PhoneNumber]
  Set Field [Contacts::gCount; Contacts::gCount + 1]
  Exit Loop If [Contacts::gCount > 10]
End Loop

This script dials local directory assistance.
Dial Phone [No dialog; 411]

**Install Menu Set**

**Purpose**

Changes the menu set based on conditions established in the script.

**Format**

Install Menu Set [specified menu set name]

**Options**

*Use As File Default* overrides the file’s default menu set specified in the Manage Custom Menus dialog box with the menu set specified in this script step. Once you close the file, the default menu set reverts to the one specified in the Manage Custom Menus dialog box. Select the menu set that you want the script step to install from the list.

**Compatibility**

This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Examples**

The following example changes the installed menu set on the menu bar to Custom Menu Set #1.
Install Menu Set ["Custom Menu Set #1"]
Set Web Viewer

Purpose
Controls the specified web viewer.

Format
Set Web Viewer [Object Name: "<object name>"; Action: <action>]

Options
Click Specify to display the Set Web Viewer Options dialog box, where you can set the following options:

- **Object Name** is the name of the web viewer to act upon. To assign an object name, select the web viewer, choose View menu > Object Info in Layout mode, and enter a name.
- For **Action**, choose one of the following:
  - **Reset** resets the named web viewer to its originally specified web address. This action also clears this web viewer’s Back and Forward history.
  - **Reload** reloads the web page that the web viewer is displaying.
  - **Go Forward** goes forward one page, the same way a web browser does.
  - **Go Back** goes back one page, the same way a web browser does.
  - **Go to URL** lets you specify a new web address to load in the web viewer. The new web address is a calculation that you specify in the Set Web Viewer - Go to URL Options dialog box. In this script step, you cannot change whether the web viewer allows interaction, shows a progress bar or status messages, or displays content in Find mode. You can set these options only when you add or change a web viewer in Layout mode.

Compatibility
This script step is supported in web publishing and in a FileMaker Server scheduled script, but the **Go Forward** and **Go Back** actions are not.

Description
Click Specify to open the Set Web Viewer Options dialog box and specify the web viewer’s object name and the action to be performed.

Notes
- This script step works only for named web viewer objects.
  - To specify the object name, either type the name in the Object Name box or click Specify and create a calculation.
  - If you choose the action **Go to URL**:
    - You specify a web address in much the same way you do when you create a web viewer on a layout. For more information, see Adding a web viewer and Defining a custom web address.
• You can send HTML data to a web viewer by including the data in a URL, using the following format:

```
data:[<mediatype>][;base64],<data>
```

Where the following syntax applies:

```
dataurl = "data:" [ mediatype ] [ ";base64" ] "" data
mediatype = [ type "/" subtype ] *( ";" parameter )
data = *urlchar
parameter = attribute "=" value
```

More information about the “data URL scheme” can be found on the web.

• In FileMaker Pro 8.5, data URLs will work on Mac OS_X but not in Windows. In addition, FileMaker Pro supports only UTF-16 character encoding. Character data encoded using other methods will not display the target of the URL.

**Examples**

The following example resets the web viewer named “Web Viewer 1” to its originally specified web address:

```
Set Web Viewer [Object Name: "Web Viewer 1"; Action: Reset]
```

The following example displays the FileMaker, Inc. homepage in the active web viewer, or displays an error message if the active object is not a web viewer:

```
If [GetLayoutObjectAttribute( Get (ActiveLayoutObjectName); 
   "objectType") = "web viewer"]
   Set Web Viewer [Object Name: Get (ActiveLayoutObjectName); 
   URL: "http://www.filemaker.com"]
Else
   Show Custom Dialog ["This object is not a web viewer."]
End If
```

The following example uses the data URL scheme to display a small icon in a web viewer named WV2:

```
Set Web Viewer [Object Name: "WV2"; Action: URL "data:image/ 
gif;base64,R0lGODlhFwAMAKEAAL+/v/// 
AAAAUwAAACH5BAAAMALAAAAAAXAAwAAAAA17hBGHapHcXJKPumizpigI+QliSH0XIjokWJ6oB 
4+qt0Zmaqpjesxz7st1YD8ZBhJajAuDgfSYTx60wIAOw="]
```

**Open URL**

**Purpose**

Allows the user to open a URL.

**Format**

```
Open URL [No dialog; <URL>]
```
Options

- **Perform without dialog** prevents the “Open URL” Options dialog box from displaying when the script step executes.
- Click **Specify** to display the “Open URL” Options dialog box, where you can type the URL directly in the text entry area or click **Specify** to create your URL from a calculation.

Compatibility

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

Description

Windows:

Open URL supports http, ftp, file, and mailto URL types. For example, if the URL starts with http, FileMaker Pro opens the user’s web browser and processes the URL.

**Note** FileMaker Pro uses the preferences stored in the Windows system file URL.DLL to determine the application to handle the URL. The URL is then passed to ShellExecute for execution.

Mac OS:

If the user has configured the internet system preferences, this script step allows the user to open a URL. Open URL supports http, ftp, file, and mailto URL types, using the applications specified in either the control panel or system preferences. For example, if the URL starts with http, FileMaker Pro opens the user's web browser and processes the URL.

Examples

The following example opens the user's web browser and displays the FileMaker, Inc. homepage (the URL specified when the Open URL script step was defined).

```
Open URL [No dialog; "http://www.filemaker.com/"]
```

The following example opens Windows Notepad and opens the file My_File.txt on the root level of the user's hard disk.

```
Open URL [No dialog; "file://c:/My_File.txt"]
```

The following example launches the preferred email application, opens a new email message, and uses the value in the Email Address field to address the message. The email address must start with mailto;.

```
Open URL [No dialog; "mailto:email address"]
```

You can also use the Open URL script step to open a shared FileMaker Pro file running on another system, as follows:

```
Open URL [No dialog; "fmp7://system:user@testlab-mac/testdb"]
```

where ‘system’ is the FileMaker Pro account name, ‘user’ is the password, ‘testlab-mac’ is a DNS entry for the machine where the file resides (you can also use an IPv4 or IPv6 address for this parameter), and ‘testdb’ is the FileMaker Pro filename.

**Important** Account name and password information entered in an Open URL script step is visible to users with script editing access privileges, and is therefore not secure.
Send Mail

Purpose
Sends an intranet or internet email message (with or without a file attachment) to one or more recipients. Email can be sent through an email application or via SMTP (Simple Mail Transfer Protocol, a set of criteria for sending and receiving email).

Format
Send Mail [No dialog; To: <to>; CC: <CC>; BCC: <BCC>; Subject: <subject>; Message: <message>; "<attachment>"

Options
• Perform without dialog instructs FileMaker Pro to put the composed email message in the email application's outbox, ready to be sent. If this option is not selected, the composed message is left open in the email application so it can be reviewed. In Microsoft Outlook Express or Microsoft Entourage on the Macintosh operating system, the new message is left in the Drafts folder.
• Click Specify to display the Send Mail dialog box, where you can set options for your mail. Choose the method by which to send mail, E-Mail Client or SMTP Server. (If you choose SMTP Server, the SMTP Options dialog box appears. For information about setting SMTP options, see Entering or editing SMTP options.) Then choose to create One email using data from the current file, or create Multiple emails (one for each record in found set).

Note If you choose SMTP Server and Multiple emails (one for each record in the found set) and FileMaker encounters an error while emailing the one of the records, the remaining records will not be sent.

For each of the following options, you can enter text directly, or click to enter values from an address book (Windows), field, or calculation.
• (Windows) Select Specify Email Addresses to enter one or more email addresses. Separate each address with a semicolon or a carriage return character.
• Select Specify Field Name to specify a field that contains one or more email addresses.
• Select Specify Calculation to specify a calculation that generates one or more email addresses.
• If you use the Specify Field Name option to specify a value for the To, CC, or BCC entries, you can also select For each message, collect addresses across found set to specify that all the values from this field in the current found set be used (to address a message to multiple recipients).
• To stores the address(es) of the recipient(s).
• CC stores the address(es) of the carbon copy recipient(s).
• BCC stores the address(es) of the blind carbon copy recipient(s)
• Subject indicates the title for the email message.
• Message indicates the text of the email message. You can type the message as text, use a field value, create a message from a calculation, or insert text from a file.
• Select Attach File to select a file to send as an attachment to the mail message. For more information, see Creating file paths.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Notes
• To send email, you must have an internet connection. In addition, to send mail through an email application, you must have the following configurations.
  Windows:
  • Microsoft Exchange, Microsoft Outlook, Microsoft Outlook Express and Eudora installed and configured properly.
  Mac OS:
  • Mac OS X Mail or Microsoft Entourage installed to send email with FileMaker Pro.
  • Internet system preferences configured for use with one of the supported mail applications.
  • If you use Eudora 5, start the Eudora application before performing this script step.
• If you select Multiple emails and For each message, collect addresses across found set, FileMaker Pro generates an email message for every record in the found set. Each message is addressed to everyone specified in the TO, CC and BCC boxes for every record in the found set.
• You can create an Adobe PDF or a Microsoft Excel file from your data to send as an attachment. For more information, see Save Records As PDF and Save Records As Excel.

Examples
Perform Find [Restore]
Send Mail [To: script examples::Field1; Subject: "This is a test email"; Message: "Hello, world." ]
Send Mail [Send via SMTP Server; To: "Inventors SIG"; Subject: "New Member Greetings"; Message: "Welcome to our group."]

Send DDE Execute (Windows)

Purpose
Sends a DDE (Dynamic Data Exchange) command to another application to execute a series of commands available in that application.

Format
Send DDE Execute [<topic text or filename>; <service name>]

• Select Attach File to select a file to send as an attachment to the mail message. For more information, see Creating file paths.
Options

Click Specify to display the “Send DDE Execute” Options dialog box, where you can set the following options:

- **Service name** is the name of the application that executes the commands. Refer to the documentation of the application you specify for the valid service name. You can enter the service name directly as text or click Specify to create the service name from a calculation.

- **Topic** is a filename or text string that describes the topic that the application executes the commands on. Refer to the documentation of the application specified in the Service Name for valid topics. You can enter the topic name directly as text or click Specify to create the topic name from a calculation. For more information, see Creating file paths.

- **Commands** are a calculated value or text string that specifies what the application does. Refer to the documentation of the application specified in the Service Name for valid commands and formats. You can enter the commands directly as text or click Specify to create the commands from a calculation.

Compatibility

This script step is:

- not supported in web publishing
- not supported in a FileMaker Server scheduled script

Description

- FileMaker Pro sends DDE execute commands, but does not receive them.

- Because DDE is a Windows feature, if you run a script containing a Send DDE execute script step on a Mac OS based computer, the script step is ignored. (An error code is generated, which can be captured with the Get(LastError) function.)

- When a FileMaker Pro script first establishes a DDE connection, the connection stays open to execute subsequent script steps for the same service name and topic. If the script includes another DDE Execute script step using a different service name or topic, FileMaker Pro closes the current connection and opens another with the new service name and topic. All open connections close when the script is completed.

Examples

The following script opens a URL in Internet Explorer. (Internet Explorer must be open when this step executes.)

Send DDE Execute [Service Name: "iexplore"; Topic: "WWW_OpenURL"; Commands: "www.filemaker.com"]

Perform AppleScript (Mac OS)

**Purpose**

Sends AppleScript commands to another application.

**Format**

Perform AppleScript [“<applescript text>”]
Options
Click Specify to display the “Perform AppleScript” Options dialog box, where you can set the following options.

• Calculated AppleScript lets you create a calculation to use as the AppleScript text.
• Native AppleScript lets you manually enter the text of an AppleScript (up to 30000 characters). Indenting of repeat loops and conditional statements is not supported.

Compatibility
This script step is:
• not supported in web publishing
• not supported in a FileMaker Server scheduled script

Description
You can create a calculation to generate the AppleScript commands to be sent, or you can type the commands directly into the Native AppleScript text area.

For calculated scripts and scripts stored in FileMaker Pro fields, the AppleScript commands are compiled each time the script is performed. If you type the script into the Native AppleScript text area, then FileMaker Pro compiles the script and detects any scripting or system errors. (The script is recompiled whenever it’s edited.)

Note Because AppleScript is a Mac OS feature, if you run a script containing a Perform AppleScript script step on a Windows computer, the script step is ignored. (An error code is generated, which can be captured with the Get(LastError) function.)

Tips on usage
• FileMaker Pro can use the Perform AppleScript script step to send Apple events to itself or to FileMaker Pro software running on other machines.
• If you are using FileMaker Pro for Mac OS and you performed a complete installation, see the FileMaker Pro Apple Events Reference database at www.filemaker.com for additional information and examples.
• Use Perform AppleScript to communicate with applets and other scriptable applications. For example, if you have created an AppleScript applet with sub-routines, you can call the handlers using the Perform AppleScript command, like this:
  Perform AppleScript ["tell application "My Applet" to doMyRoutine()"

Examples
This example copies text from a field in a FileMaker Pro database and pastes it into a new AppleWorks word processing document.
Copy [Select; Correspondence::Letter]
Perform AppleScript ["Tell Application "AppleWorks 6" activate make new document paste end tell"]

This example sets the primary monitor to its minimum bit depth.
Perform AppleScript ["tell application "Finder" to set bounds of window "My Files" to {100,100,100,100}"
Execute SQL

**Purpose**
Executes any SQL statement.

**Format**
Execute SQL [No Dialog; ODBC: <datasource name>; <native SQL or calculated SQL>]

**Options**
- **Perform without dialog** prevents the Specify SQL dialog box, the Select ODBC Data Source dialog box, and the Password dialog box from displaying when the script step executes.
- Click **Specify** to display the Specify SQL dialog box, where you can set the following options.
  - **Specify** displays the Select ODBC Data Source dialog box. Select a data source from the list, click **OK**, and enter a valid user name and password.
    - **Note** Select **Save user name and password** if you selected **Perform without dialog**, or users may be unable to access your data source.
  - **Calculated SQL text** lets you create a calculation to use as the SQL query.
  - **SQL text** lets you manually enter a SQL query statement.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script
- not supported in runtime solutions

**Description**
You can use this script step to manage data in a foreign data source through an ODBC SQL interface. This provides more control over the FileMaker Pro interaction with ODBC data sources. You can add multiple Execute SQL steps to a script.

**Important** See [Get functions](#) for information about error handling.

- **Get(LastODBCError) function** returns the latest error state returned by ODBC.
- **Get(LastError) function** returns the last FileMaker Pro error.

**Notes**
- If you use the Execute SQL script step to send ODBC data that contains Unicode strings, your ODBC driver must support Unicode. Otherwise, the results may be inconsistent.
- ODBC import, the Execute SQL script step, and external SQL data sources are not supported in runtime solutions created with FileMaker Pro Advanced.
Examples
The following example of a manual SQL statement entry executes once each time you run the 
Execute SQL script step.

```sql
INSERT INTO Employees (EmployeeID, FirstName, LastName, Title, WorkPhone, Salary) VALUES (100, N'Joe', N'Smith', N'Software Engineer', '987-7000', 100000)
UPDATE Employees SET Title = N'Manager' WHERE EmployeeID = 103
DELETE FROM Employees WHERE EmployeeID = 103
```

The following example of a field-based SQL statement shows a calculation field that concatenates text with a field that can be used to insert or update a number of records in the found set.

```sql
"INSERT INTO Employees (EmpID, LastName) Values (" & EmpID & "," & "N'" & Last Name & ")"
```

With EmpID of 103 and Lastname of "Smith" the calculation of the field above is translated to:

```sql
INSERT INTO Employees (EmpID, LastName) Values (103,N'Smith')
```

**Note** Because Microsoft SQL Server supports both Unicode and non-Unicode field types, you must prefix all Unicode strings with an uppercase "N" (which stands for "National" in the SQL-92 standard). Otherwise, when a Unicode string containing non-English characters is passed to Microsoft SQL Server, you may lose any data that doesn't exist in the Microsoft SQL Server code page.

Some Database Management Systems (DBMS), such as the MySQL 3.51 driver, don't support the "N" prefix. For more information, see the manual for your DBMS.

Send Event (Mac OS)

**Purpose**
Sends an Apple event to another application, called the target application.

**Format**

```
Send Event ["<Target Application>"; "<Event Class>"; "<Event ID>", 
"<Document or Calculation or Script Text>"]
```

**Options**
Click Specify to display the “Send Event” Options dialog box, where you can set the following options.

- **Send the <value> event with** lets you choose between the following:
  - **open application** tells FileMaker Pro to open an application. Click Specify Application to select the application.
  - **open document** tells FileMaker Pro to open a document in the target application. You can also specify a calculated value or script.
  - **do script** tells FileMaker Pro to perform a script in the language of the target application. Click Specify Application to select an application, and use Document to select the document to use with the target application. Or, select Script text and enter script text or type in the name of the script (make sure it is one that will be recognized by the target program).
Miscellaneous script steps

- **other** displays the Specify event dialog box, where you can manually enter the Apple event **Event class** and **Event ID**.

- Select **Document** or click **Specify** to select the document you want used with the target application. For more information, see Creating file paths.

- Select **Calculation** or click **Specify** to create a calculation that generates a value you want to send with the event.

- **Bring Target Application to foreground** activates the target application and displays it on the screen. Displaying the target application can slow down the performance of your script. If **Bring Target Application to foreground** is not selected, the event is performed in the background.

- **Wait for event completion before continuing** tells FileMaker Pro to wait until the event is finished before continuing. If you don't want to wait until the event is completed, deselect this option.

- **Copy event result to the clipboard** copies the resulting events data to the Clipboard, from which it can later be retrieved. This option is disabled if **Bring Target Application to foreground** is selected.

- Click **Specify Application** to display a dialog box where you can select the target application.

**Compatibility**

This script step is:

- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**

Select the event you want to send in the “Send Event” Options dialog box.

- When FileMaker Pro sends an Apple event, it sends text (not compiled) data. You must know what information the target application expects to receive with an event.

- Each Send Event script step sends one event. You can include more than one Send Event in a script.

If you are using FileMaker Pro for Mac OS and performed a complete installation, see the FileMaker Pro Apple Events Reference database at www.filemaker.com for additional information and examples.

**Examples**

The following example opens an application:

Send Event ["TextEdit", "aevt", "oapp"]
Send Event (Windows)

**Purpose**
Starts another application, opens a document in another application, or prints a document in another application.

**Format**
Send Event ["<aevt>"; "<event name>"; "<filename>"]

**Options**
Click **Specify** to display the “Send Event” Options dialog box, where you can set the following options.

- For **Send the <event name> message**, select:
  - **open document/application** to tell FileMaker Pro to open a document file or application. Documents are opened using the application that Windows has associated with the document’s file type.
  - **print document** to tell FileMaker Pro to print a document in another application.
- Select **File** or click **Specify** to specify a document/application to open, or a document to print. For more information, see [Creating file paths](#).
- Select **Calculation** or click **Specify** to create a message from a calculation.
- Select **Text** to manually enter text for the message to be sent.
- Select **Bring target application to foreground** to activate the target application and display it on the screen. Displaying the target application can slow down the performance of your script. If **Bring target application to foreground** is not selected, the event is performed in the background.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
For additional scripting capabilities, you can create a program in Basic or C and run that program with this script step.

**Note** When specifying a document or application by calculation or text, the file type or application name will appear as <unknown> in the script definition.
Examples

To launch the Notepad application, select the open document/application message, click **File**, and specify notepad.exe. The following script step appears in the Script Definition dialog box:

Send Event ["aevt"; "odoc"; "NOTEPAD.EXE"]

To open a document created by another application, select the open document/application message, click **File**, and specify the file (for example image.bmp). The following script step appears in the Script Definition dialog box:

Send Event ["aevt"; "odoc"; "image.bmp"]

Comment

**Purpose**

Adds comments to the list of steps in the script.

**Format**

`#<comment text>`

**Options**

Click **Specify** to display a dialog box to enter the comment text.

**Compatibility**

This script step is also supported in web publishing and in a FileMaker Server scheduled script.

**Description**

Use comments to document your script. Comments appear in bold and are prefaced with a `#`. Comments are present only when you are viewing the script and do not appear when the script is performed.

**Note** Comments print in italics.

**Examples**

Set Error Capture [On]
Perform Find [Restore]
If [Get (LastError) = 401]
    #If the current error is 401, nothing was found
    Show Custom Dialog ["No records were found."]
    If [Get (LastMessageChoice) = 1]
        Modify Last Find
    Else
        Enter Browse Mode []
    End If
End If
Flush Cache to Disk

**Purpose**
Performs an immediate flush of the FileMaker Pro internal disk cache to the computer's hard disk.

**Format**
Flush Cache to Disk

**Options**
None.

**Compatibility**
This script step is:
- not supported in web publishing
- not supported in a FileMaker Server scheduled script

**Description**
This operation is normally done periodically during idle time and after extensive structural changes, such as converting files and defining fields.

**Examples**
Perform Find [ ]
Sort Records [Restore]
Go to Record/Request/Page [First]
Replace Field Contents [script examples::Serial Number; Serial numbers]
Flush Cache to Disk

Exit Application

**Purpose**
Closes all open files and exits the FileMaker Pro application.

**Format**
Exit Application

**Options**
None.

**Compatibility**
This script step is also supported in web publishing and in a FileMaker Server scheduled script.
Description
If this script step is associated with a button and if the file is accessed through the web, performing the script will log out the current web session.

Examples
The following example, used as a startup script, prevents the user from opening the database on a weekend.

Allow User Abort[Off]
If [DayName(Get (CurrentDate)) = "Saturday" or DayName(Get( CurrentDate)) = "Sunday"]
   Exit Application
End If