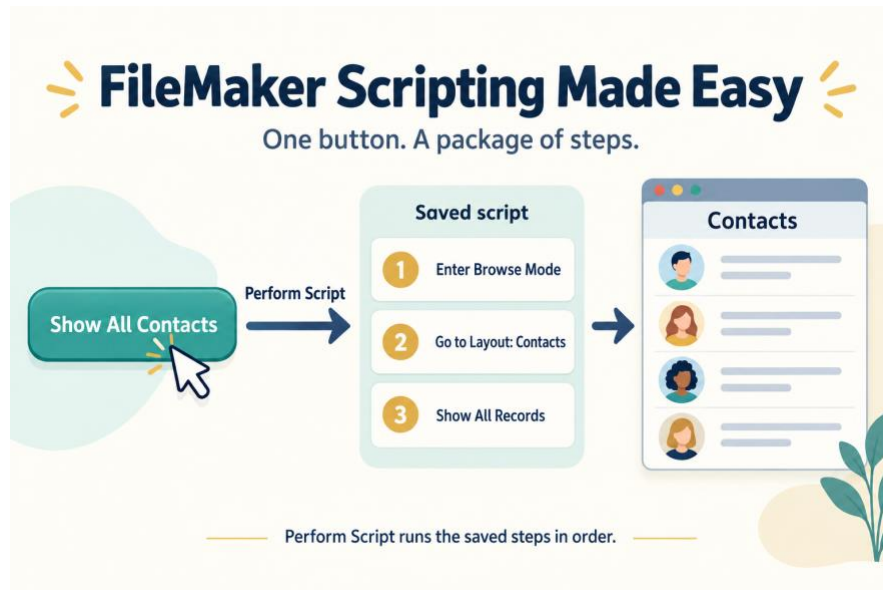




FileMaker Scripting Made Easy

The basics you need to get started

I am essentially self-taught when it comes to FileMaker scripting. I have learned by following my nose—trying something, seeing what happens, and making adjustments. Attaching an action to a button, whether a single step or a saved script, is a useful place to begin.

You do not need to understand everything about scripting to make something useful. Start with a small task you already know how to do manually.

What is a script?

A script is a set of instructions that tells FileMaker what to do. A simple script might open a layout and display all its records. A more involved script might find certain records, sort them, and prepare a report.

Think of a script as a checklist. For a straightforward script, FileMaker works through the steps from top to bottom. Your job is to choose the steps, set their options, and put them in the right order.

FileMaker provides a library of script steps, so you can begin by selecting commands rather than writing programming code from scratch. [Claris: Automating tasks with scripts](#)

Somewhat like an Access query

If you have used Microsoft Access queries, part of FileMaker scripting may feel familiar. An Access query can retrieve matching records, perform calculations, combine information from

tables, or make changes to data. You can save it and run it again whenever needed. [Microsoft: Introduction to queries](#)

A FileMaker script can also save instructions for finding and working with records. Its role is broader, though: a script can coordinate a whole sequence of actions, such as opening a layout, finding contacts in a particular city, sorting them, and preparing a report.

An Access query mainly describes the data result or data operation you want. A FileMaker script describes the steps you want the application to carry out. The comparison is useful, although scripts are closer in purpose to Access macros than to queries alone.

A package of steps—like an old batch file

If you remember the old MS-DOS batch files, you already have a useful way to think about scripts. A batch file collected commands in one place so you could run them together instead of entering each command separately.

A saved FileMaker script is a similar **package of steps**. You give that package a name, arrange its instructions, and run it when needed.

Strictly speaking, **the saved script is the package; Perform Script is the command that runs it**. When you assign Perform Script to a button, you are telling FileMaker, “When someone clicks here, run this named package of instructions.” [Clariss: Perform Script](#)

For example, a button labeled **Show All Contacts** might run three steps: enter Browse mode, open the Contacts layout, and show all its records. One click starts the sequence.

A button: one step or a saved script?

When setting up a button, you have two useful choices:

- **Single Step:** Assign one command directly to the button—for example, **Go to Layout**.
- **Perform Script:** Have the button run a script you have saved in the Script Workspace.

A single step is handy for a simple action. A saved script gives you room to add more steps and lets several buttons run the same instructions. A saved script can contain just one step, too.

[Clariss: Defining or changing a button](#)

Start by describing the task

Before opening the Script Workspace, write down what you want to happen in everyday language.

For example:

“When I click this button, I want FileMaker to open my Contacts layout and show all my contacts.”

That gives us a small, useful first project.

For this exercise, use a practice file with a few records and an existing Contacts layout. If your layout has a different name, substitute that name below.

Create your first script

- 1 Choose **Scripts > Script Workspace**.
- 2 Click **New Script** and name it **Show All Contacts**.
- 3 Add **Enter Browse Mode**. Set its **Pause** option to **Off**.
- 4 Add **Go to Layout**. In its options, select your Contacts layout.
- 5 Add **Show All Records**.
- 6 Close the script tab and choose **Save** when prompted.

To add a step, begin typing its name in the script editing area and select the matching command, or double-click it in the script steps pane. Click a step's options to choose its settings. [Clariss: Creating and editing scripts](#)

Your finished script will look roughly like this:

```
Enter Browse Mode [ Pause: Off ]  
Go to Layout [ "Contacts" ]  
Show All Records
```

This is a guide to the steps and their order. Build it using FileMaker's script steps and options; you do not need to type the brackets and punctuation yourself.

What do those three steps do?

Enter Browse Mode puts FileMaker in the mode used to view and edit records. Setting Pause to Off lets the script continue immediately. [Clariss: Enter Browse Mode](#)

Go to Layout opens the layout you selected. Choosing the layout explicitly helps the script start its work in the right place. [Clariss: Go to Layout](#)

Show All Records replaces the current found set—the records selected by a search—with all records in the table associated with that layout. It does not show every record from every table in your file. [Clariss: Show All Records](#)

The order matters. We go to the Contacts layout before asking FileMaker to show all records there.

Connect the script to a button

- 1 Return to your layout and enter **Layout mode**.
- 2 Use the **Button tool** to draw a button.
- 3 Give it a clear label, such as **Show All Contacts**.
- 4 In **Button Setup**, choose **Perform Script** for the action.
- 5 Select your saved **Show All Contacts** script and click **OK**.
- 6 Return to **Browse mode**, saving the layout if prompted, and click the button.

You have now connected a visible button to a reusable set of instructions. You can also assign this same script to buttons on other layouts. [Claris: Button setup](#)

Give it a useful test

First, perform a find that displays only some of your contacts. Then click your new button.

Does FileMaker open the correct layout? Are all the contacts displayed again? Try running it from another layout as well.

Testing from different starting points helps you discover assumptions you may have built into the script.

If something does not work, check:

- **The button:** Did you assign the intended script?
- **The destination:** Does Go to Layout point to the correct layout?
- **The order:** Are the steps arranged as shown?
- **The options:** Is a step pausing or waiting for input?
- **The saved changes:** Did you save your script after editing it?

Change one thing at a time and try again. That makes it much easier to understand what fixed the problem.

A few habits that help

Give scripts descriptive names. **Show All Contacts** will be easier to recognize later than **Script 1**.

Keep your first scripts short. A small script that does one useful job is a good foundation.

Add a **Comment** step when a note would help explain your intention. FileMaker ignores comments when running the script, but they can help you understand it when you return months later. [Clariss: Editing scripts](#)

Practice in a copy of your file when experimenting with steps that change or delete data.

Build from what you know

You can leave conditions, loops, and variables for later. Begin with a task you repeat, describe it in plain language, and turn that description into a few script steps.

The first time a button performs a useful task for you, scripting starts to feel less mysterious. From there, you can learn one new step at a time.