

# What is AppleScript?

AppleScript is a powerful programming language, which gives you the ability to control any application, provided it provides the AppleScript library.

If you're comfortable working with scripts and using a Mac, AppleScript may be the automation solution for you. AppleScript is a powerful programming language, which gives you the ability to control any application, provided it provides the AppleScript library.

Use AppleScript for common tasks like automatically resizing photos in Photoshop, renaming folders and locking files with a password! Today's article will show you how to start using AppleScript.

## Learn about AppleScript

1. What is AppleScript?
2. Overview of AppleScript pre-installed
3. Learn about the tell command
4. Write the first AppleScript

## What is AppleScript?

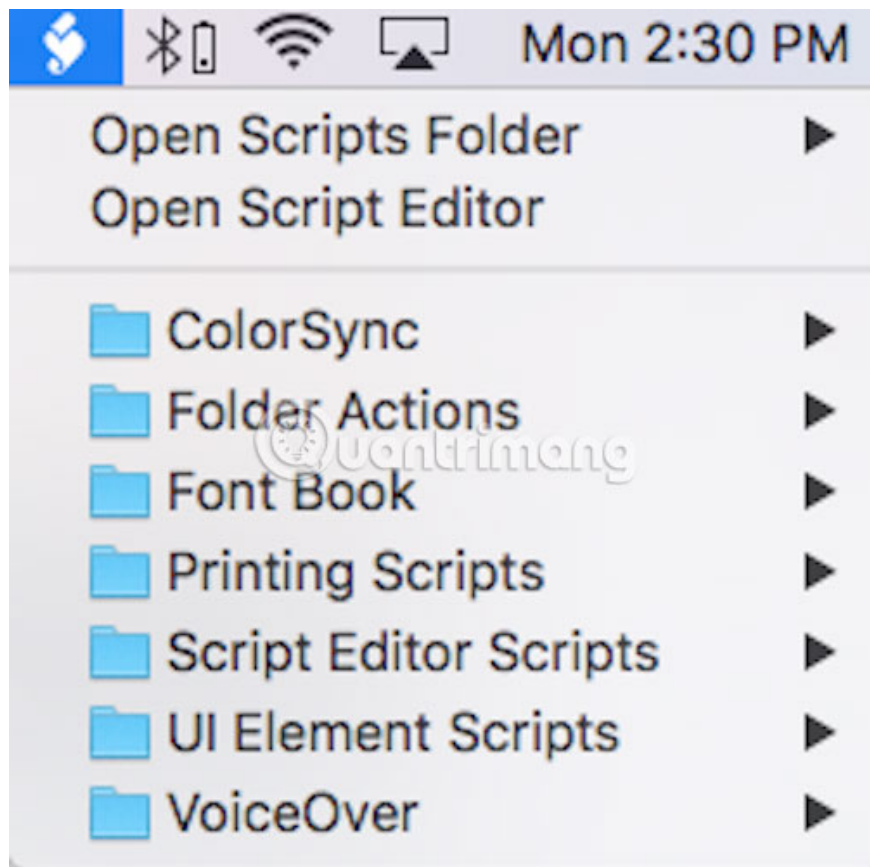
Like bash, AppleScript is a scripting language. And similar to Automator, it interacts primarily with applications and Finder to automate tasks. AppleScript has been released as part of Mac OS System 7, since 1993. But since then, AppleScript has been stuck in the **Utilities** folder .

AppleScript has increased its power with the launch of Mac OS X. The Cocoa framework helps developers create applications compatible with AppleScript. That flexibility increased, combined with the ability to work directly with the command line, making AppleScript one of the best tools to give macOS an advantage over iOS when it comes to automation.

## Overview of AppleScript pre-installed

Before analyzing exactly what AppleScript can do, consider the pre-installed scripts with Script Editor and how you can use them.

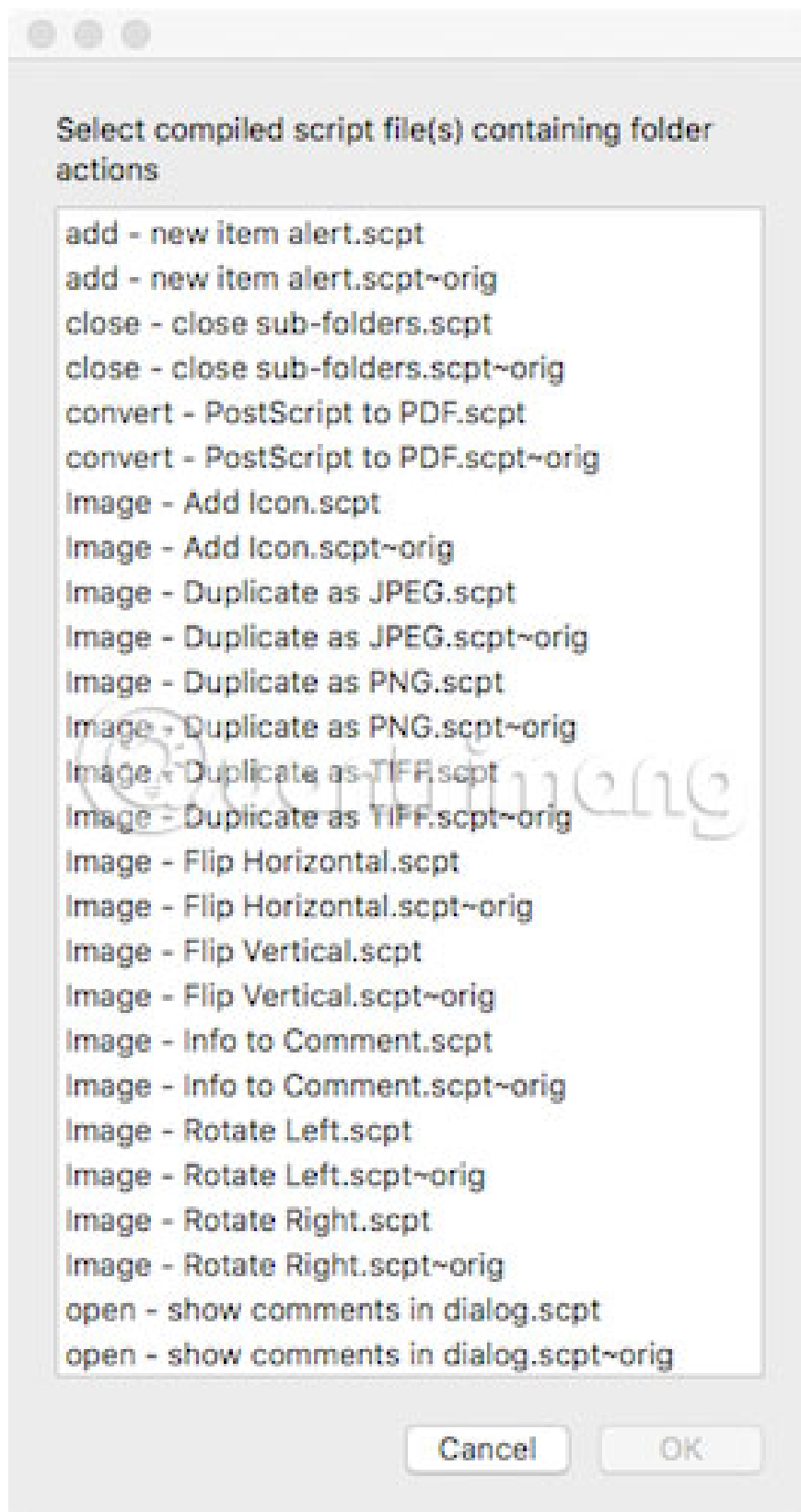
The preinstalled scripts are in **Macintosh HD> Library> Scripts** . You can also access them by opening the **Script Editor** (searching with Spotlight), going to **Preferences> General> Show Script menu bar menu** , then clicking on the script icon that appears in the menu bar.



You can only run one of these scripts from the menu bar.

Let's take a look at **Folder Actions**. Folder Actions is an AppleScript attached to the folder. When enabled, this script will run on any file added to that directory.

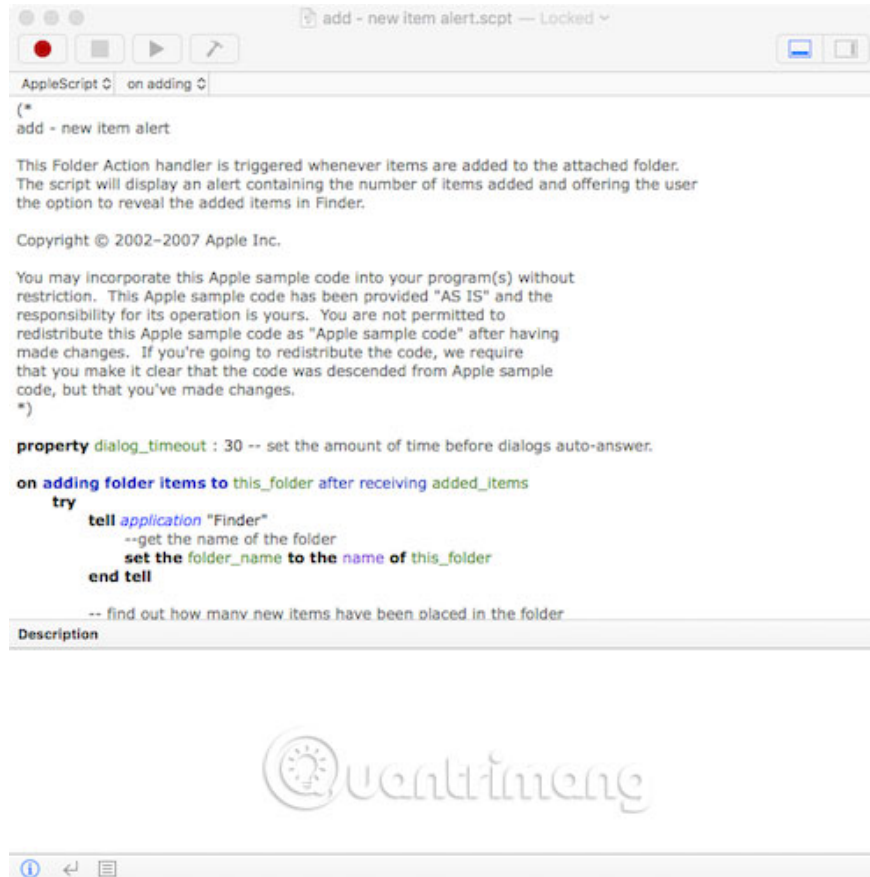
If you go to **Folder Actions**> **Attach Scripts to a Folder** , a pop-up window will ask for the type of script you want to add to the folder. You can flip images horizontally or vertically, copy them in JPEG or PNG format, rotate them or warn them when a new item is added.



Once you've selected the script and the folder you want to attach, right-click on the folder itself. Go to **Services > Folder Action Setup** and select **Enable Folder Actions** . Then drag a file at the top of the folder to see your AppleScript running.

Explore the **Scripts** menu bar to find out what AppleScript can do. Go to the **Scripts** folder , right-click any script and open it with Script Editor.

## Learn about the tell command



AppleScript uses a user-readable syntax. This means that, compared to many other programming languages, AppleScript is written in an easy-to-understand format. Because AppleScript uses full words and sentences to send commands, it is easy to understand and learn.

Let's look at the **add - new item alert.sctpt syntax** in Folder Actions. This will give an idea of the most basic statement in AppleScript: The **tell** command .

```
on adding folder items to this_folder after receiving added_items try tell applic
```

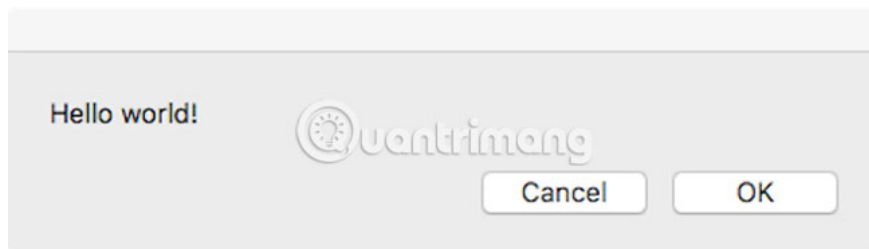
A tell statement has three parts:

1. From **tell**
2. Object for reference (in this case the **Finder** application)
3. Action to take (here is '**set the folder\_name to the name of this\_folder**').

The purpose of AppleScript is to automate tasks by asking the application to perform the given tasks. Therefore, the **tell** command is very necessary. You can go further in the AppleScript world just with this **tell** command alone.

Also note that the content **--get the name of the folder is** actually just a comment, telling the user what the script is doing at the time.

## Write the first AppleScript



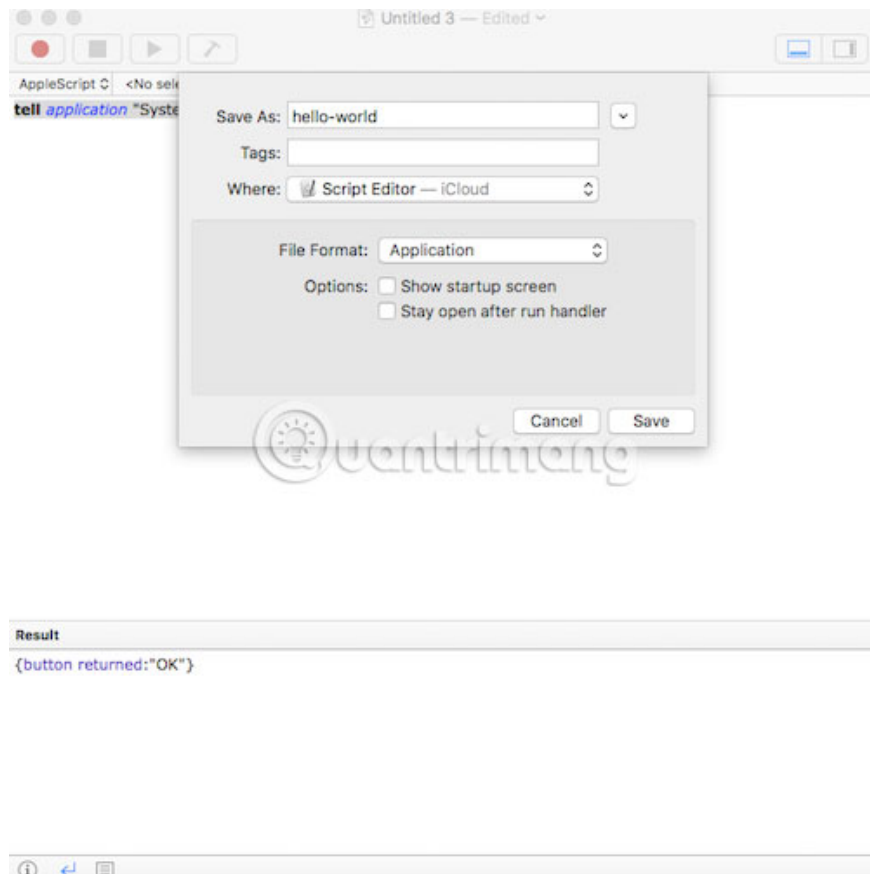
If you have a bit of programming experience and are familiar with concepts like variables, do-while loops and conditions, you can do a lot of things with AppleScript, beyond the scope of this introduction. Currently, the article will show you how to create, write, run and save basic scripts:

1. Create script: Open **Script Editor** and go to **File> New** .
2. Write your script: The Script Editor window is divided into two halves. The upper half is to enter your script and the bottom half will show you the output when you run it. Import:

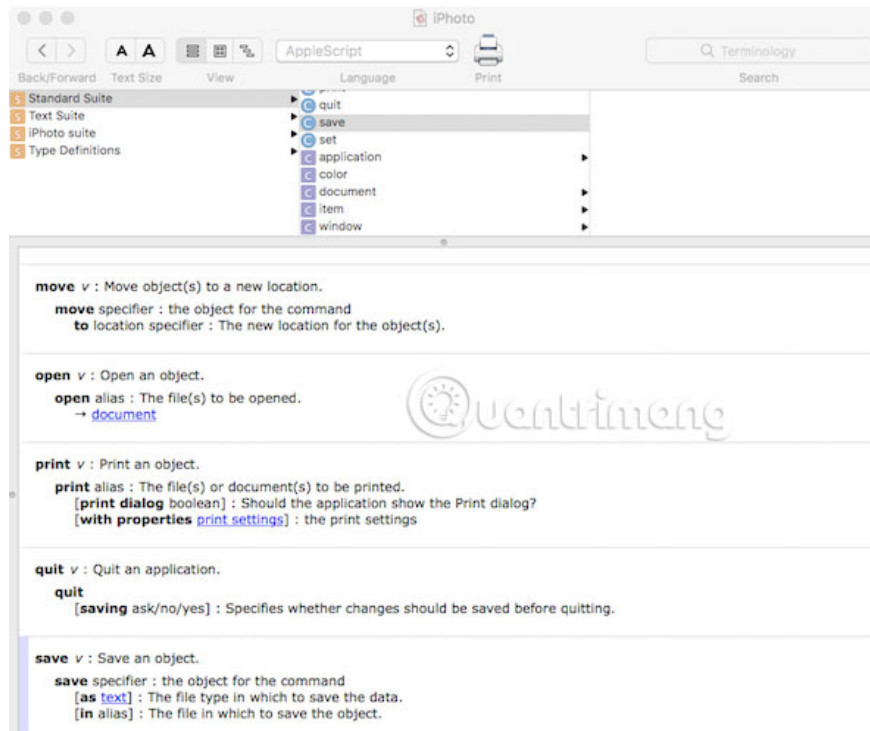
```
tell application "System Events" to display dialog "Hello world!"
```

Then press the button in the hammer-shaped menu bar right above the script to compile it. This will run through the script to check for syntax errors. If you do not receive the error dialog and the script changes the format and font, it has been successfully compiled.

3. Run the script: Next to the button shaped hammer is the **Play** button . Click that, and see what happens.
4. Save the script: Now you have a basic script. You can save it as a clickable application. Go to **File> Save** and in **File Format**, select **Application**. Now, instead of opening the **Script Editor** and clicking **Play**, simply double-click the script to run it. If you want to run the script in bash, you can use AppleScript to turn the bash script into clickable applications.



With this simple syntax, you can ask almost any Mac application to do a lot of things. To review the available commands for a certain application, go to **File > Open Dictionary** and select the application. From there, you can see all available AppleScript commands.



If programming hurts you, there are more simple ways to automate tasks. Automator uses a friendly GUI and a simple interface to turn annoying tasks into single-taskable tasks with just one click.

Although Automator can't customize or do complicated tasks like AppleScript, it's simple and much harder to break. Take a look at some of the processes that Automator saves you time like resizing images on a Mac or creating a simple script, if you are interested in this tool.

You finished reading the article "**What is AppleScript?**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.