# User Manual

How to: Download a Script

If you want your device to perform particular tasks, you need to have a task script saved to your device. This script should be written in Javascript code, using the available state triggers, and saved online in your own webspace, eg. Google Drive. Open Scripter and hit the orange Download button. This will display a window where you can enter the URL of your script and give it a file name. If no file name is given, Scripter will take the name found at the end of the URL. Hit Download and a notification at the bottom of the screen will appear to tell you that the script is downloading, when it is complete or if an error has occured while downloading. Once the download is successful, the script will be available to use, and can be seen on the Scripters "Scripts" page.
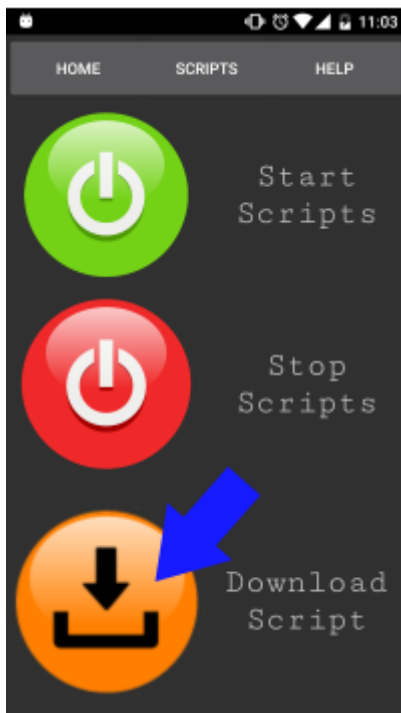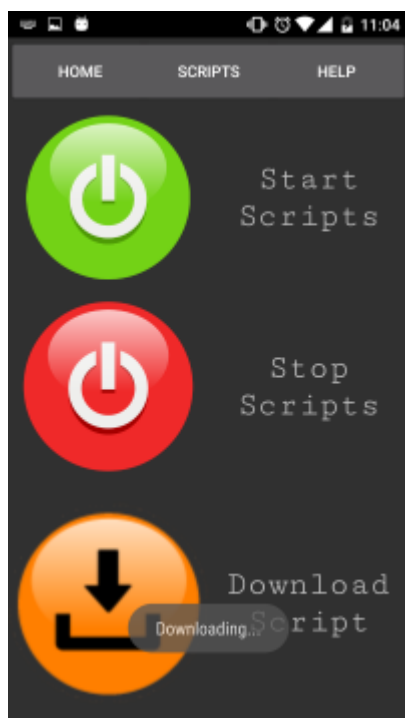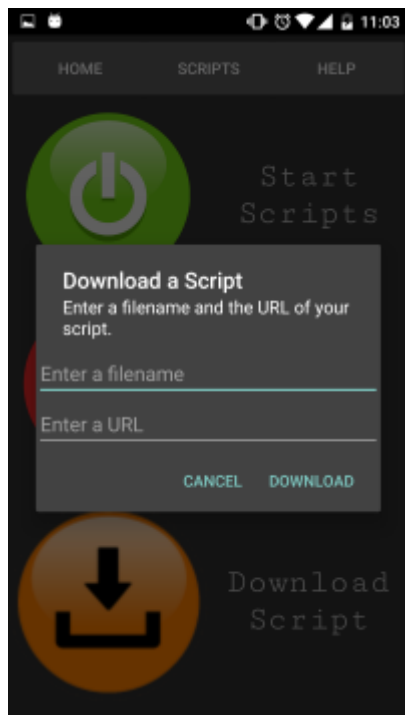
How to: Enable/Disable your scripts

Once you have multiple task scripts written and saved to your device, you have the ability to decide which scripts will be enabled when you hit Start. This is for users who may not want all scripts to run all of the time. Go to the "Scripts" page. Here you will see a list of the scripts you have downloaded to your device. Tapping the name of a script will enable and disable a script. A green tick signals that the script is currently enabled, and will be active once you hit Start.
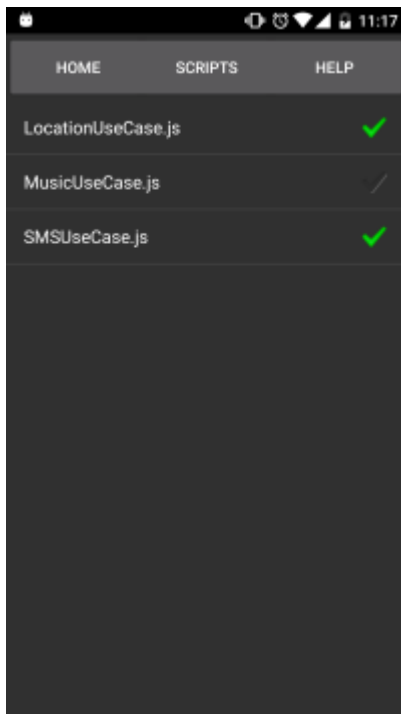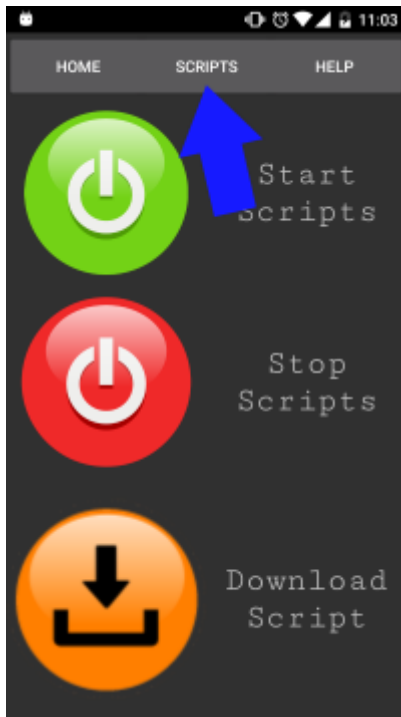
How to: Start and Stop Scripts

Enabled scripts can be started and stopped at any time from the Home page.

To start scripts, simply press the green "Start Scripts" button. This will display a constant notification in the notification bar of your device to notify you that your scripts are running in the background. This will remain in your notification bar after Scripter has been closed and can be used to re-open Scripter at any time.

To stop scripts, simply press the red "Stop Scripts" button. The constant notification will be removed from your notification bar and the scripts will stop running in the background on your device.
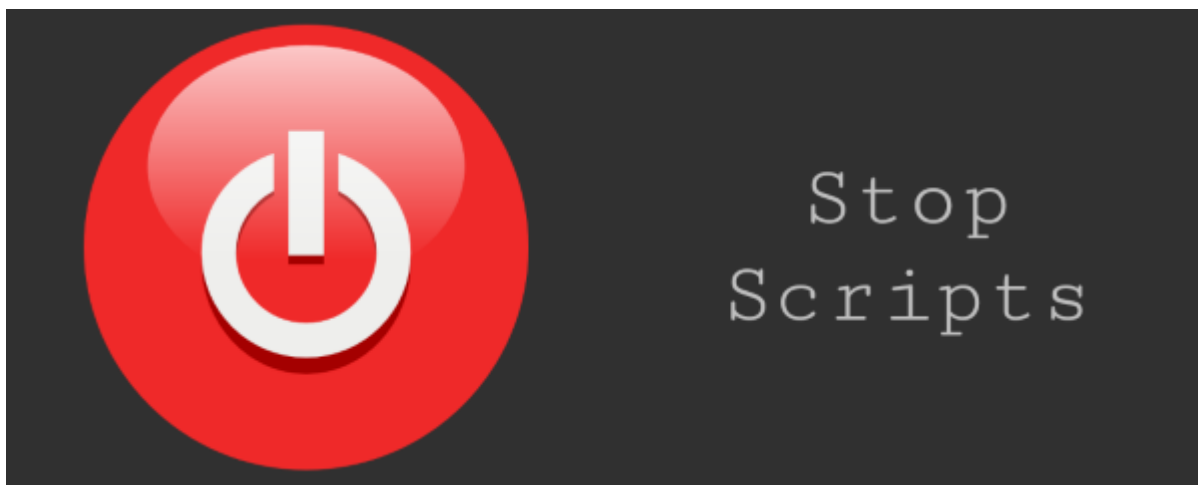

How to: Use Script Options

Script options for each script can be accessed from the Scripts page.

To access these options, simply long-click (hold down your finger) on a particular script until you are presented with a list to select from.
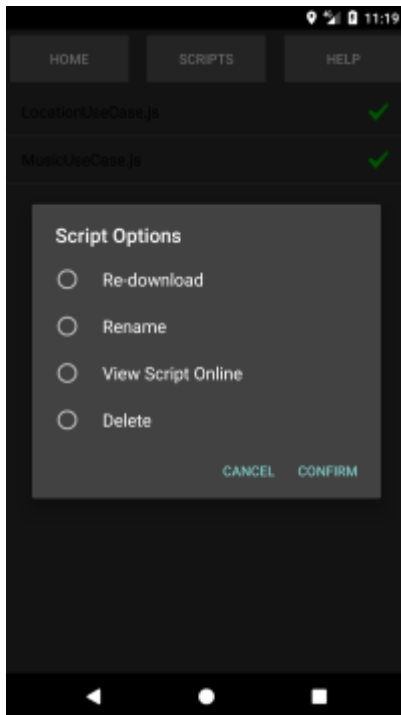
From here, the following options can be selected:

- **Re-download:** The selected script will be downloaded from it's source URL again and saved with the same name as before on your device. Messages will be displayed on screen to notify you of what's happening.
- **Rename:** An additional dialog will be presented to you in which you can enter a new name for the script. Once a new name has been entered, press the "Rename" button. The script will then be given the name that you entered. Note: script names must end in ".js" which will be automatically appended to any names entered if not already present.
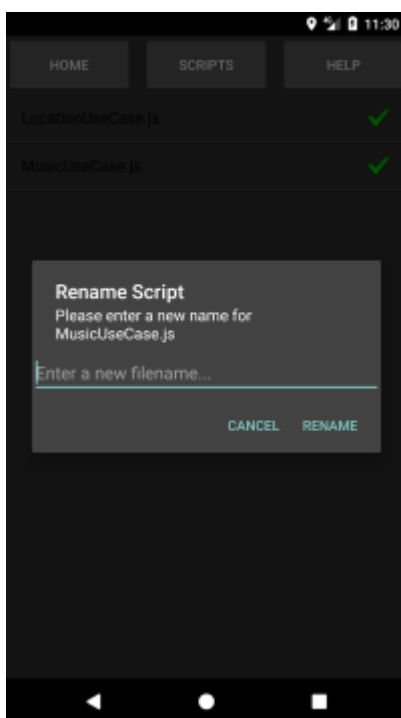
- **View Script Online:** The URL of the selected script will be opened in your device's web browser for

viewing of it's content. Note: this content may not match the content of the script storedlocally on your device. If this is the case, simply use the re-download option to update the script.

- **Delete:** The selected script will be removed from your device with a message being displayed to confirm that it has been removed successfully.

Understanding: States, Functions and Dependencies

Scripter uses scripts written in Javascript to run your desired tasks. These scripts have state dependencies. These variables can also be thought of as "triggers", as they are the states that trigger a particular script to run. For example, a Location dependency set to the location of your home can be used to run certain scripts only when you are at home.

States

For the release of this application, we have supplied the following common state dependencies:

minute:

- *Description*: The current minute of the device's clock.
- *Range*: 0 - 59.

hour:

- *Description*: The current hour of the device's clock in "24 hour" format.
- *Range*: 0 - 23.

month:

- *Description*: The current month of the device's calendar.
- *Range*: 1 - 12.

year:

- *Description*: The current year of the device's calendar.
- *Range*: N/A.

dayOfMonth:

- *Description*: The current day of the device's calendar in numerical form.
- *Range*: 1 - 31.

dayOfWeek:

- *Description*: The current day of the device's calendar in word form.
- *Range*: "Monday" - "Sunday".

headphonesPluggedIn:

- *Description*: Indicates whether a headset is connected to the device or not. It will be true if a headset is connected.
- *Range*: true or false.

latitude:

- *Description*: The latitude of the device's current location.
- *Range*: N/A.

longitude:

- *Description*: The longitude of the device's current location.
- *Range*: N/A.

wifiConnected:

- *Description*: Indicates whether the device is connected to a Wifi network or not. It will be true if the device is connected to a Wifi network.
- *Range*: true or false.

SSID:

- *Description*: The SSID of the device's current Wifi network, if connected to one. Otherwise it's value will be "unknown ssid".
- *Range*: N/A.

Functions

The following functions allow you to interact with your device from scripts:

toastMessage(*message*) :

- *Description*: Displays the given message on your device's screen for a couple of seconds. This type of notification is known as a "Toast Notification".
- *Arguments*:
  - message: The message to be displayed.
- *Return value*: N/A.

openCamera() :

- *Description*: Opens the device's default camera application.
- *Arguments*: N/A.
- *Return value*: N/A.

openMusicPlayer() :

- *Description*: Open's the device's default music application.
- *Arguments*: N/A.

- *Return value*: N/A.

isWeekday(*dayOfWeek*) :

- *Description*: Determines whether dayOfWeek is between Monday and Friday (inclusive) or not.
- *Arguments*:
  - dayOfWeek: A day of the week in word form with it's first letter capitalized. E.g. "Monday".
- *Return value*: true or false.

isWeekend(*dayOfWeek*) :

- *Description*: Determines whether or not dayOfWeek is one of Saturday or Sunday.
- *Arguments*:
  - dayOfWeek: A day of the week in word form with it's first letter capitalized. E.g. "Monday".
- *Return value*: true or false.

openWebpage(*url*) :

- *Description*: Opens the device's default web browser with the given url.
- *Arguments*:
  - url: The url to be opened, "http://" will be automatically added to the front of the url if not already present.
- *Return value*: N/A.

displayNotification(*title*) :

- *Description*: Creates a notification in the device's notification bar with the given title.
- *Arguments*:
  - title: The title to be used in the notification.
- *Return value*: N/A.

displayNotification(*title*, *description*) :

- *Description*: Creates a notification in the device's notification bar with the given title and description.
- *Arguments*:
  - title: The title to be used in the notification.
  - description: The description to be used in the notification.
- *Return value*: N/A.

withinMeterRadiusOf(*meters*, *latitude*, *longitude*) :

- *Description*: Determines whether the device's current location is within the given amount of metres of the location represented by the given latitude and longitude.
- *Arguments*:
  - meters: The distance, in meters, the device can be in any direction from the given location in

order for this function to return true.
  - ○ latitude: the latitude of some location.
  - ○ longitude: the longitude of some location.
- *Return value*: true or false.

turnOnSilent() :

- *Description*: Turns the device on "Silent" (also called "Do Not Disturb") mode.
- *Arguments*: N/A.
- *Return value*: N/A.

turnOnVibrate() :

- *Description*: Turns the device on "Vibrate" mode.
- *Arguments*: N/A.
- *Return value*: N/A.

turnOnSound() :

- *Description*: Turns the device on "Normal" mode.
- *Arguments*: N/A.
- *Return value*: N/A.

isOnSilent() :

- *Description*: Determines whether the device is on "Silent" (also called "Do Not Disturb") mode or not.
- *Arguments*: N/A.
- *Return value*: true or false.

isOnVibrate() :

- *Description*: Determines whether the device is on "Vibrate" mode or not.
- *Arguments*: N/A.
- *Return value*: true or false.

isOnSound() :

- *Description*: Determines whether the device is on "Normal" mode or not.
- *Arguments*: N/A.
- *Return value*: true or false.

sendSMS(*phoneNumber, message*) :

- *Description*: Sends an SMS to the given number containing the given message.
- *Arguments*:
  - ○ phoneNumber: The phone number to send the SMS to.

- message: The text to contain in the SMS.
- *Return value*: N/A.

Dependencies

Dependencies define the states a script will listen for changes in before running. These can be specified in each script.

For example, the following script listens to the devices headphone jack. If it is a weekday, the music player will automatically open when the user inserts their headphones. This is makes it quicker and easier for you to listen to your music on the go!

```
1   var dependencies = ["headphonesPluggedIn"];
2
3   if (headphonesPluggedIn && isWeekday(dayOfWeek)) {
4       openMusicPlayer();
5   }
```

If no dependencies are specified, your script will run whenever any supported states change.