Lesson Four:
Functions and the Myo
Debug Console

Lesson Four:
Functions and the Myo Debug Console

Overview

In lesson three, we were introduced to `scriptId`, `scriptTitle`, and `scriptDetailsUrl`.

This lesson will look at how to format functions and use the **Myo Debug Console**. Myo Scripts are completely based on functions, so being comfortable with writing functions is an important asset.

Objective

By the end of this lesson you will have written your first functional script, and you will be able to use the Myo Debug Console for future testing.

Goals

- Learn how to structure functions
- Familiarize yourself with the Debug Console
- Write a Myo Script

Lesson Length (Time)

60 Minutes

PART 1: FUNCTIONS

Functions in Lua follow this format:

```
function <function name>(<argument 1>, <argument 2>, <... etc>)
```

on a new line, replacing the values in angle brackets with your own names. Put the logic you need indented on the next few lines, and end the function with end on its own line.

Note: In some programming languages, semicolons are required at the end of each line; this is not necessary in Lua. However, your code will not break if you put them in by force of habit. You also don't need to specify what type of variables your arguments are, or what you'll be returning.

PART 2: YOUR FIRST FUNCTION

Our first function is called `onForegroundWindowChange`. This function lets you determine if your script should be active or not. Each script is targeted at a certain application (Spotify, Notepad, Microsoft Word, etc.), so this function activates every time a new window is in the foreground. The foreground refers to whichever window is currently in focus.

`onForegroundWindowChange` has two variables: `app` and `title`. (This is a callback defined by Myo Script, which cannot be changed). `app` is a unique string that helps identify your application, and `title` is the actual title of the window.

Below your `scriptDetailsUrl` assignment, you will want to have this:

```
function onForegroundWindowChange(app, title)
end
```

You could write a script that works on a specific web page, on the browser in general, or on any other application. If you detect an application that your script supports, return `true`. This is built into the `onForegroundWindowChange` function, and the first script to return `true` gets control of the application.

For now, we'll just be returning `true`. To begin, we'll output the `app` and `title` arguments to the **Myo Debug Console**. This is usually the first thing you want to do when supporting a new application to get an idea of how you can identify the application in question.

For example, if you were to open a new tab on Google Chrome, the `app` and `title` would be "com. google.Chrome" and "New Tab", respectively.

PART 3: MYO DEBUG

To identify the name and title of an application, we can print them to the Debug Console. The **Myo Debug Console** will be useful in the future for verification that your script is working as intended.

Myo Scripts have a global `myo` object defined, which stores all of the myo-specific functions. This means all the myo-specific functions will be available by default. One of these functions is called `debug`. `myo.debug` will accept a string and then output it to a special console window. Remember that this window will only appear if you have **Developer Mode** enabled in Myo Connect's **Preferences**.

Inside your `onForegroundWindowChange` function, you will want to have this:

```
myo.debug("onForegroundWindowChange: " .. app .. ", " .. title)
```

`..` concatenates (links together) two strings together in Lua, and `<object>.<function name>(<arguments>)` calls a function. Note that in the above function, "onForegroundWindowChange: " is something being displayed on the **Myo Debug Window**. The comma, ", " is also only there for the clarity so the user is not confused. These can both be replaced with whatever you want to appear in the debug window.

PART 4: USING YOUR SCRIPT

Your full script should look something like this:

```
scriptId = 'com.thalmic.examples.myfirstscript'
scriptTitle = "My First Script"
scriptDetailsUrl = ""

function onForegroundWindowChange(app, title)
   myo.debug("onForegroundWindowChange: " .. app .. ", " .. title)
   return true
end
```

Save any changes, then add it using the **Application Manager**. For simplicity sake, you can disable any other connectors. Make sure you press the refresh button every time you update your script so that you can see your changes. If you used the same format as above, you should see something like this pop up every time you click on a new application or window:
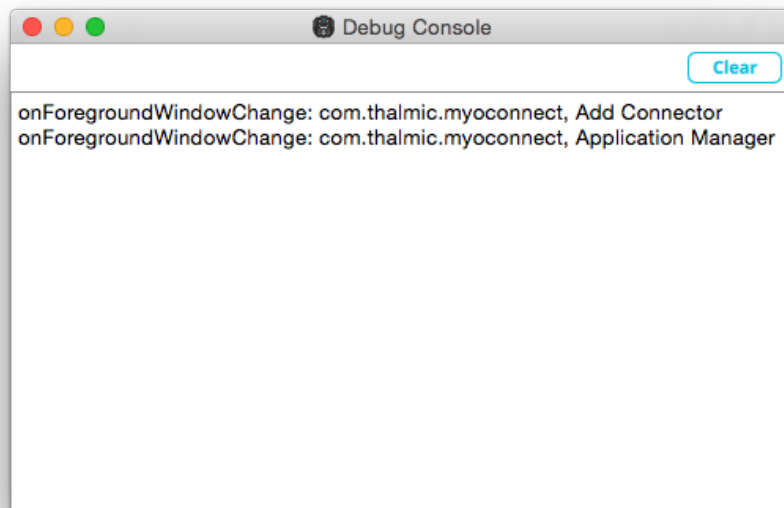

You just wrote your first Myo Script! This script will output the application name and title of each window you switch your foreground to. The application name and title will help us in the future when we want our script to only work for specific applications.

Challenge Activities

The questions below refer to the full script shown in Part 4 of this lesson.

1. Answer questions based on the following image:
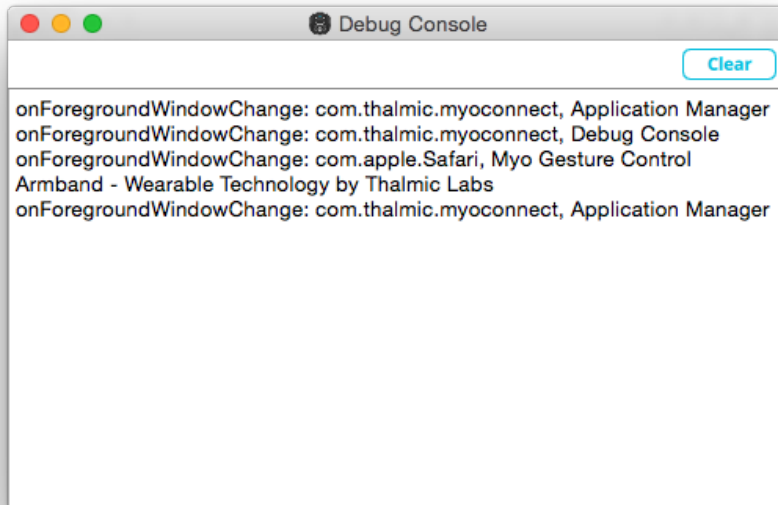
a) What are the names for the app and title of the first window?

b) What are the names for the app and title of the second window?

c) What are the names for the app and title of the third window?

2. Using the `onForegroundWindowChange` function, find the `app` and `title` for the following applications.
a) Your text editor

b) A word document

c) The Debug Console

3. (Exercise) The following exercise is an optional activity to help you better understand the `onForegroundWindowChange` function. **<u>You will not need to make the following changes in a real script</u>.**

onForegroundWindowChange can be modified to perform other tasks whenever the foreground window is changed. Let's modify our script to output a number before the app and title when the foreground window is changed, and make that number increase by 1 each time the foreground window changes.

Your modified script should look something like below. The bolded text shows what has been changed.

```
scriptId = 'com.thalmic.examples.lesson4Q2'
scriptTitle = "Lesson 4 Question 2"
scriptDetailsUrl = ""

number = 0

function onForegroundWindowChange(app, title)
number = number + 1
myo.debug(number)
myo.debug("onForegroundWindowChange: " .. app .. ", " .. title)
end
```

Solutions:

1. a)

```
function onForegroundWindowChange(app, title)
        myo.debug("onForegroundWindowChange: " .. title .. ", " .. app)
        return true
end
```

1. b)

```
function onForegroundWindowChange(app, title)
        myo.debug("Current Window: " .. app .. ", " .. title)
        return true
end
```

1. c)

```
function onForegroundWindowChange(app, title)
        myo.debug("App: " .. app .. ", Title: " .. title)
        return true
end
```

2. c)

```
app: com.thalmic.myoconnect
title: Debug Console
```