Quick Lesson -

HTML Requests for Cloud Variables

“Introduction to Functions, Arguments, and Return Data Types”

# Table of Contents

[Table of Contents](#_v47flr3lgp60)

[Introduction](#_4j8sc2chtrcv)

[Security and Access Codes](#_c5d5q71s9da5)

[HTML GET Requests for Cloud Variables](#_b75a61imsun)

[HTML POST for Cloud Functions](#_yvyyqh5vsgvx)

[Dynamic HTML pages with JavaScript for Photon](#_lf1h0t1qrk2m)

# Introduction

HTML is underlying the language of The Web. Since the Particle Photon is an IoT device it shouldn’t be surprising that it is very easy to access and control your Photon. This quick lesson will walk you through how to write just a few lines of HTML to return values stored in your Photon and execute Cloud Functions.

# Security and Access Codes

Assuming we don’t want the entire world to know the status of our microcontroller variables...it is important that IoT devices use access codes. The access “codes” for our Photons consist of two alphanumeric values a “Device ID” (never changes) and an “Access Token” (can be reset to block all old program access.)

1. In the Particle web IDE, click on Devices (), then click on the arrow () next to your Photon. It should look something like this:

|  |
| --- |
|  |
| Photon Device ID |

1. Record your Device ID (Hint: Cut-n-Paste is your friend here!)
2. Click on the Settings button (). It should look something like this:

|  |
| --- |
|  |
| Photon Access Token |

1. Record your Access Token

# HTML GET Requests for Cloud Variables

You can now use the following syntax to access your cloud variable from anywhere on the planet with internet connectivity!

1. Modify the following URL and enter it into a new tab on your web browser:

|  |
| --- |
| **https://api.particle.io/v1/devices/0123456789abcdef/TrigPos?access\_token=123412341234** |

Where “**0123456789abcdef**” should be replaced by your Device ID and “**123412341234**” should be replaced by your Access Token. You might noticed above that term “**TrigPos**” is contained above, you can modify this to access other cloud variables identified in your Photon code.

Hopefully when you get everything right it will look something like this:

|  |
| --- |
|  |
| Cloud Variable Query Result |

As you can see above the value of the Cloud Variable “TrigPos” is “true”. This is a good thing, since we declared “bool TriggerPosition = 1;” in our program.

# HTML POST for Cloud Functions

Once a cloud function has been registered, it can be called using a POST HTML request. Unfortunately, POST requests cannot be generated with a URL alone (ie., you can’t just type something into the browser address bar.) Instead, we’ll have to create a simple webpage of our own to issue a POST request.

The following is a template to start from:

|  |
| --- |
| **<html>**  **<body>**  **<form action="https://api.particle.io/v1/devices/31003f00154f343339383037/setMode?access\_token=cd5d28f314ddbae6724d9efbf703cedc493f3802" method="POST">**  **<input type="radio" name="args" value="Heat">Set thermostat mode to HEAT.<br>**  **<input type="radio" name="args" value="Off">Set thermostat mode to OFF.<br>**  **<input type="radio" name="args" value="Cool">Set thermostat mode to COOL.<br>**  **<input type="submit" value="Do it!">**  **</form>**  **</body>**  **</html>** |

For now, copy and paste this text into a new text file called “test.html” on your desktop. Sometimes copying from a pdf file garbles text a bit, so make sure the file looks exactly like the text in the box above. Replace your **Device ID** and **Access Token** in the code. Open the file in your web browser; it should look something like this:

|  |
| --- |
|  |
| Webpage for controlling thermostat mode |

When you click on “Do it!” the cloud will initiate a request to your photon to run the setMode function.When the function completes, the cloud will relay the return value back to your web browser.It should look something like this:

|  |
| --- |
|  |
| Text returned after clicking “Do it!” |

# Further Reading - Dynamic HTML Pages

If you want to make prettier and more functional web pages with your photon you can continue reading in the next [Quick Lesson - Dynamic Web Pages for Photon](https://drive.google.com/open?id=1QP5puXLcIQExmOuSTPMXwmQS4q_tPUvqdtB9yRRmp7Y).

In addition, there are many helpful websites that will get you started creating good looking and functional websites. A Google search will treat you well, but here is one option → <https://www.w3schools.com/html/default.asp>