

Homework #5 Microsoft Azure with Python

This semester we are allowing all students to explore cloud computing as offered by the Microsoft Azure. Using the instructions below one can establish a service using Microsoft Azure App Service developed for Assignment #6 and have it executed there.

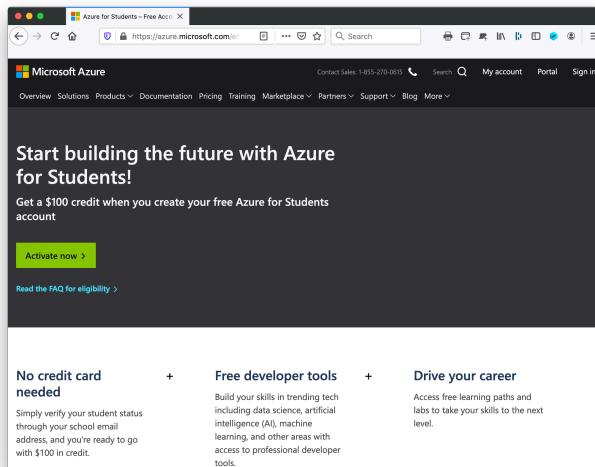
1. Sign up for Microsoft Azure for Students

To sign up for the Free Trial, with a \$100 credit, you will not need a credit card. First of all, use a browser in **Incognito** or **InPrivate** mode (such as Chrome – File New Incognito Window or Firefox – File – New Private Window).

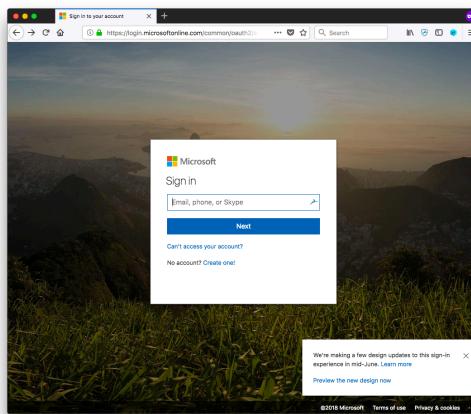
To sign up go to:

<https://azure.microsoft.com/en-us/free/students/>

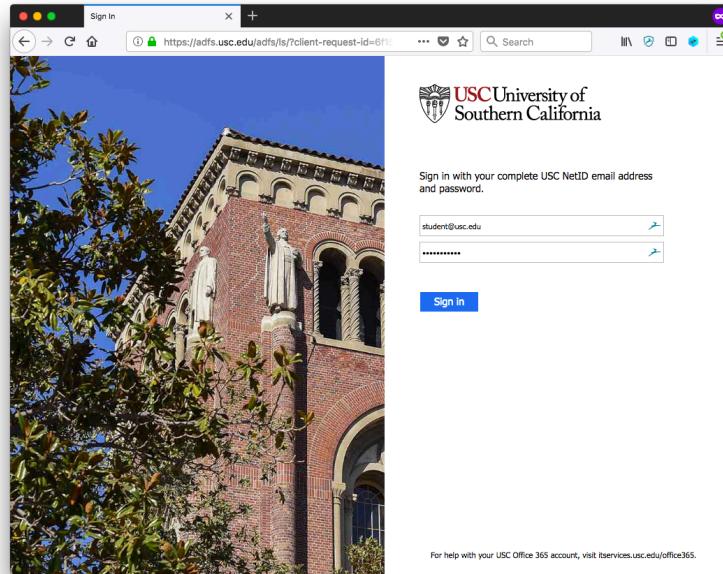
In the Microsoft Azure for Students page, click on **Activate now >**:



The Microsoft **Sign in** page will be displayed.



Enter your **USC e-mail address**. The USC Shibboleth page will be displayed and will be used to verify your academic status and receive the \$100 credit.

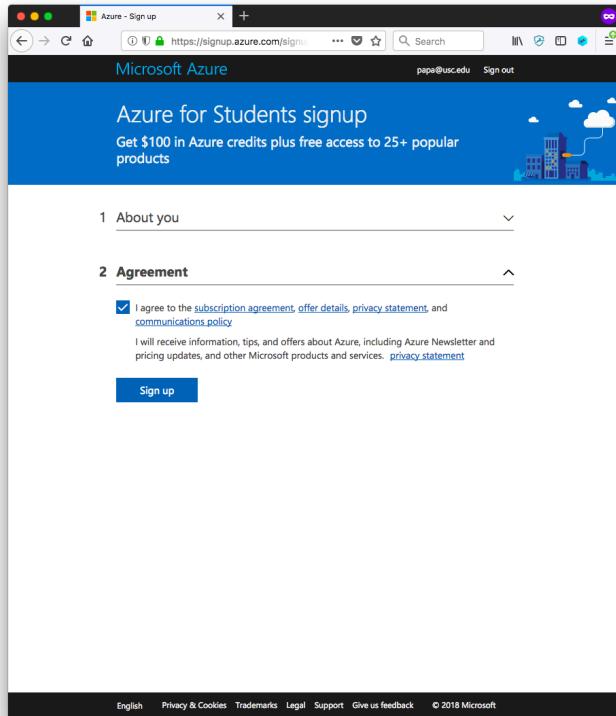


Enter your **USC account password**. Click **Sign in**.

The **Azure for Students signup** page is displayed.

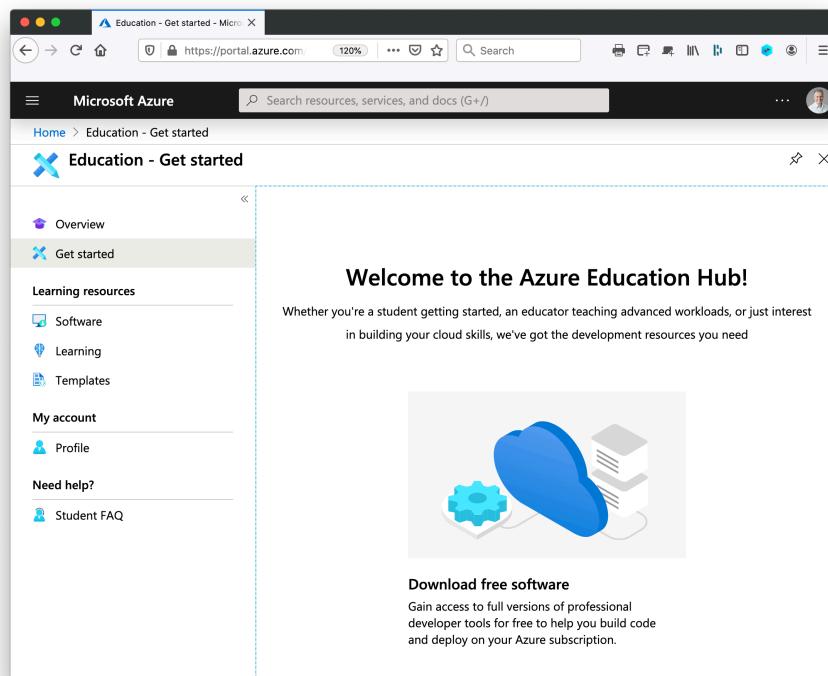
A screenshot of a web browser window showing the Microsoft Azure for Students signup page. The URL is https://signup.azure.com/signin. The page has a blue header with the Microsoft Azure logo and the text "Azure for Students signup" and "Get \$100 in Azure credits plus free access to 25+ popular products". Below the header, there are two sections: "1 About you" and "2 Agreement". The "1 About you" section is expanded, showing form fields for "Country/Region" (United States), "First name" (Marco), "Last name" (Papa), "Email address for important notifications" (papa@usc.edu), and "Phone" ((310) 555-1212). There is also a note: "By proceeding you acknowledge the [privacy statement](#) and [subscription agreement](#)". A blue "Next" button is at the bottom of this section. The "2 Agreement" section is collapsed. At the very bottom, there is a footer with links: English, Privacy & Cookies, Trademarks, Legal, Support, Give us feedback, and © 2018 Microsoft.

Enter your personal data in the form. Click **Next**.

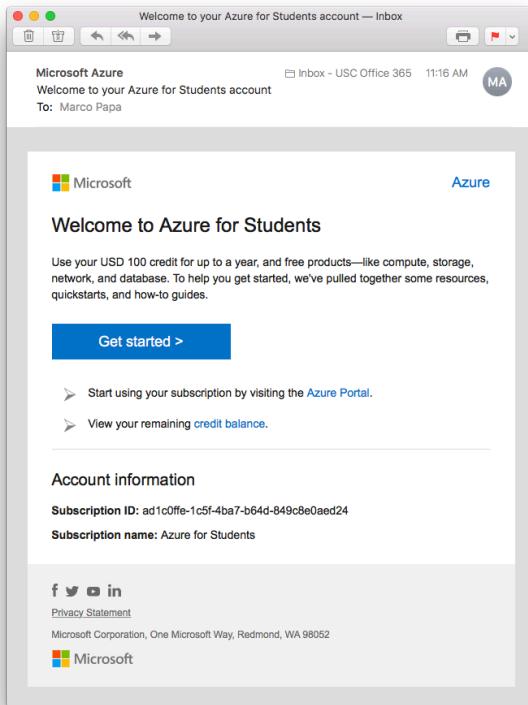


Agree to the free subscription offer by selecting the checkbox. Click **Sign Up**.

You will be taken to the Microsoft Azure “Education” Portal.



You should also receive a confirmation “**Welcome to Azure for Students**” e-mail.



2. Create a Python App in Azure App Service on Linux

App Service on Linux provides a highly scalable, self-patching web hosting service using the Linux operating system. Microsoft Azure provides a **Quickstart tutorial** that shows how to deploy a Python app to Azure App Service on Linux. The tutorial is available at:

<https://docs.microsoft.com/en-us/azure/app-service/containers/quickstart-python>

You can follow the steps in the tutorial using a Mac, Windows, or Linux machine.

1. To complete the Quickstart, you need to install the following pre-requisite components:
 - a. **Git**. If you need to install Git, please go here:
<https://git-scm.com/>
 - b. **Python 3.7**. If you need to install Python, please see the instruction for Python on CGP.
 - c. **Azure CLI**. Instructions are provided here:
<https://docs.microsoft.com/en-us/cli/azure/install-azure-cli>

2. The Quickstart tutorial above allows you to create the web app with built-in image using the **Azure CLI in Cloud Shell**, and you use Git to deploy the Python code to the web app. After you have installed the Azure CLI, test that is running, by typing ‘az’:

```
hidden_service$ az
Welcome to Azure CLI!
Use "az -h" to see available commands or go to https://aka.ms/CLI.
Telemetry
The Azure CLI collects usage data in order to improve your experience.
The data is anonymous and does not include command-line argument values.
The data is collected by Microsoft.
You can change your telemetry settings with "az config".
Welcome to the cool new Azure CLI!
Use "az --version" to display the current version.
Here are the basic commands:
account      : Manage Azure subscription information.
acr          : Manage Private registries with Azure Container Registries.
ad           : Manage Azure Active Directory Graph entities needed for Role Based Access Control.
advisor     : Manage Azure Advisor.
aks          : Manage Azure Kubernetes Service resources.
ams          : Manage Azure Media Service resources.
apis         : Manage Azure API Management services.
appserviceconfig : Manage App Service plans.
appservice   : Manage App Service plans.
batch        : Manage Azure Batch.
blob         : Manage Azure Blob storage.
billing      : Manage Azure Billing.
cdn          : Manage Microsoft Azure Bot Service.
```

3. The Quickstart gives step by step instructions for all of the following:

- **Download the sample**

```
Last login: Sat Feb 15 16:06:52 on ttys006
The default interactive shell is now zsh.
To update your account to use zsh, please run 'chsh -s /bin/zsh'.
For more details, please visit https://support.apple.com/kb/HT200859.
server:~ marcopappa$ git clone https://github.com/Azure-Samples/python-docs-hello-world
Cloning into 'python-docs-hello-world'...
remote: Total 58 files, received 58, 0 bytes (0 delta 0), pack-reused 58
Unpacking objects: 100% (58/58), done.
server:~ marcopappa$ pwd
/Users/marcopappa
server:~ marcopappa$ cd python-docs-hello-world
server:~/python-docs-hello-world marcopappa$
```

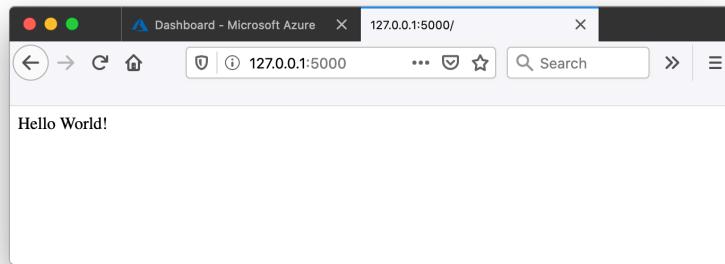
- **Build the sample**

```
python-docs-hello-world$ python3 -m venv venv
python-docs-hello-world$ source venv/bin/activate
(venv) server:python-docs-hello-world marcopappa$ pip install -r requirements.txt
Collecting click==6.7
  Downloading https://files.pythonhosted.org/packages/34/c1/8886f99713dd9993c5366c362b2f908f18269f8d792aff1bf0d708775a77/click-6.7-py2.7-py3-none-any.whl (71kB) | 71kB 3.2MB/s
Requirement already satisfied: Flask==0.10.2 in /usr/local/lib/python3.7/site-packages (from -r requirements.txt (line 2)) (0.10.2)
Collecting itsdangerous==0.24
  Downloading https://files.pythonhosted.org/packages/dc/b/a/680c0db945c0ff6d688d997513lab3f25b2f2bcfe1ab221165194b2d4/itsdangerous-0.24.tar.gz (46kB) | 51kB 3.2MB/s
Collecting Jinja2==2.10
  Downloading https://files.pythonhosted.org/packages/77/ff/ae64accdfc95f27a016a70ed8e8686763ba4d277a78ca76f32659220a731/jinja2-2.10-py2.7-py3-none-any.whl (126kB) | 126kB 3.2MB/s
Collecting MarkupSafe==1.0
  Downloading https://files.pythonhosted.org/packages/4d/de/32d741db316d8fdb7688822d37801ef7aa448255de9699ab4bfcfd4712b/MarkupSafe-1.0.tar.gz (46kB) | 51kB 3.2MB/s
Collecting Werkzeug==0.14.1
  Downloading https://files.pythonhosted.org/packages/28/c4/12e3e56473e52375aa9c4764e79dbf3feaf6682bef8d0aae04fe335243/Werkzeug-0.14.1.tar.gz (322kB) | 322kB 3.2MB/s
Building wheels for collected packages: itsdangerous, MarkupSafe
Building wheel for itsdangerous: setup.py: error: invalid command: 'egg_info'
Created wheel for itsdangerous: filename=itsdangerous-0.24-cp37-none-any.whl size=10621 sha256=83d73787cabfa81b44dacf19be76a4712bf4644
09e7ff48a767903169abeb33208
Building wheel for MarkupSafe (setup.py) ... done
Created wheel for MarkupSafe: filename=MarkupSafe-1.0-cp37-cp37m-macosx_10_15_x86_64.whl size=15563 sha256=4084587a7f892b712a783d645
Stored in directory: /Users/marcopappa/Library/Caches/pip/wheels/33/56/0be49a5c612fffe1c5a632146b16596f9e64676768661e4e46
Successfully built itsdangerous MarkupSafe
Installing collected packages: click, itsdangerous, MarkupSafe, Jinja2, Werkzeug
  Found existing installation: Click 7.0
    Uninstalling Click-7.0:
      Successfully uninstalled Click-7.0
  Found existing installation: itsdangerous 1.1.0
    Uninstalling itsdangerous-1.1.0:
      Successfully uninstalled itsdangerous-1.1.0
  Found existing installation: MarkupSafe 1.0.1
    Uninstalling MarkupSafe-1.0.1:
      Successfully uninstalled MarkupSafe-1.0.1
  Found existing installation: Jinja2 2.10.3
    Uninstalling Jinja2-2.10.3:
      Successfully uninstalled Jinja2-2.10.3
  Found existing installation: Werkzeug 0.16.0
    Uninstalling Werkzeug-0.16.0:
      Successfully uninstalled Werkzeug-0.16.0
  Successfully installed Jinja2-2.10 MarkupSafe-1.0 Werkzeug-0.14.1 click-6.7 itsdangerous-0.24
(venv) server:python-docs-hello-world marcopappa$ export FLASK_APP=application.py
```

- Run the sample locally at <http://127.0.0.1:5000/>

```
(venv) server:python-docs-hello-world marcopapas$ flask run
 * Serving Flask app "application.py"
 * Environment: production
WARNING: Do not use the development server in a production environment.
Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

- Browse the sample output locally (Press CTRL+C to quit)



- Sign in to Azure. You will need to go through USC authentication.

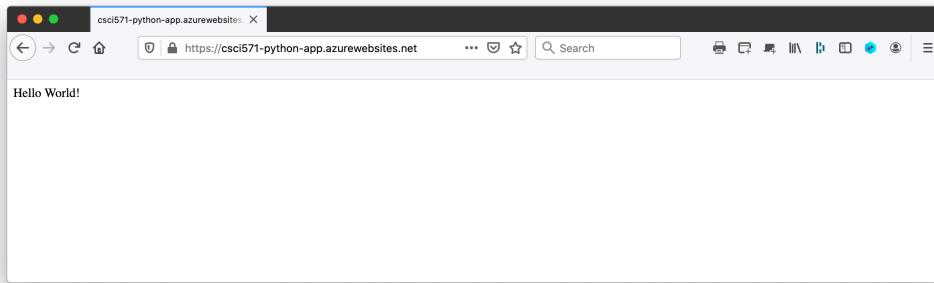
```
(venv) server:python-docs-hello-world marcopapas$ az login
You have logged in. Now let us find all the subscriptions to which you have access...
[{"cloudName": "AzureCloud", "id": "ad9e0ffe-1c5f-4ba7-b64d-849c8e9aed24", "isDefault": true, "name": "Azure for Students", "state": "Enabled", "tenantId": "9d9aaac1-389f-4cb1-a113-081be6cc25fc", "user": {"name": "papagusc.edu", "type": "user"}}, {"cloudName": "AzureCloud", "id": "3752d048-8acf-4b05-9006-78c1e614af92", "isDefault": false, "name": "Azure for Students Starter", "state": "Enabled", "tenantId": "9d9aaac1-389f-4cb1-a113-081be6cc25fc", "user": {"name": "papagusc.edu", "type": "user"}}]
```

- Deploy the sample to Azure

```
^C(venv) server:python-docs-hello-world marcopapas$ webapp cs1571-python-app up --sku F1 -n csc1571-python-app -l westus
webapp cs1571-python-app doesn't exist
Creating webapp 'cs1571-python-app'...
Configuring default logs for the app if not already enabled
Creating zip with contents of dir /Users/marcopapa/python-docs-hello-world ...
Getting scm site credentials for zip deployment
Starting zip deployment. This operation can take a while to complete ...
Deployment endpoint responded with status code 202
You can launch the app: http://csc1571-python-app.azurewebsites.net
{
  "URL": "http://csc1571-python-app.azurewebsites.net",
  "appServicePlan": "papa_asp_Linux_westus_0",
  "location": "westus",
  "name": "csc1571-python-app",
  "os": "Linux",
  "resourceGroup": "papa_rg_Linux_westus",
  "runtime_version": "python3.7",
  "runtime_version_detected": "",
  "sku": "FREE",
  "src_path": "/Users/marcopapa/python-docs-hello-world"
}
(venv) server:python-docs-hello-world marcopapas$
```

- **Browse** the app **on Azure** (the full URL for your application is listed in the “URL” value in the displayed object) as in:

`http://csci571-python-app.azurewebsites.net`

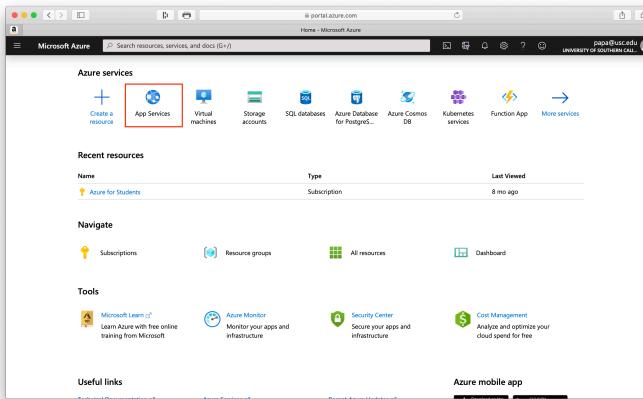


- **Redeploy Updates** to your app after changes are performed. Use the same AZ CLI command that you used to deploy the initial version of your app. See the Quickstart tutorial for more details.

3. Manage your Azure web app

Go to the **Azure portal** to manage the web app you created.

<https://portal.azure.com/>



From the list of **Azure services**, click the **App Services** icon, and then click the name of your Azure web app.

The screenshot shows the Microsoft Azure portal's App Services overview. A single web app, 'csci571-python-app', is listed. It is running in the West US location under a free app service plan. The portal interface includes standard navigation and filtering tools.

Click the app **Name**. You see your web app's **Overview** page. Here, you can perform basic management tasks like browse, stop, start, restart, and delete.

This screenshot shows the detailed overview of the 'csci571-python-app'. The left sidebar lists various configuration options. The main area displays the app's configuration, including its resource group, status, location, and subscription details. Below this are sections for 'Diagnose and solve problems' and 'App Service Advisor'. The bottom half of the screen features three real-time monitoring charts: 'Http 5xx' (showing error counts over time), 'Data In' (showing incoming data volume in bytes), and 'Data Out' (showing outgoing data volume in bytes).

The left menu provides different pages for configuring your app.

Have fun exploring Azure Web Services!!