lab

lab title

Creating a Python Development Environment with AWS Cloud9

V1.05

Course title

BackSpace Academy
AWS Certified Associate

Table of Contents

Contents

About the Lab

Please note that not all AWS services are supported in all regions. Please use the US-East-1 (North Virginia) region for this lab.

These lab notes are to support the AWS Cloud9 lab in the Setting Up section of the AWS Certified Developer Associate Course.

Please note that AWS services change on a weekly basis and it is extremely important you check the version number on this document to ensure you have the lastest version with any updates or corrections.

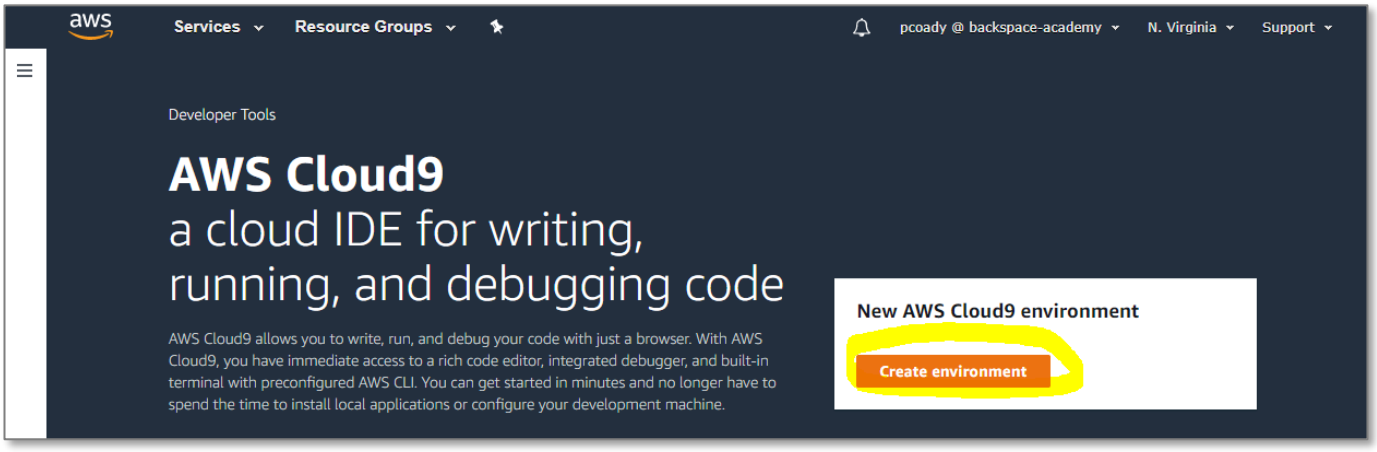
Creating a Cloud9 Development Environment on EC2

In this section, we will use the Cloud9 service to create a development environment on an EC2 instance.

From the AWS console click 'Services"

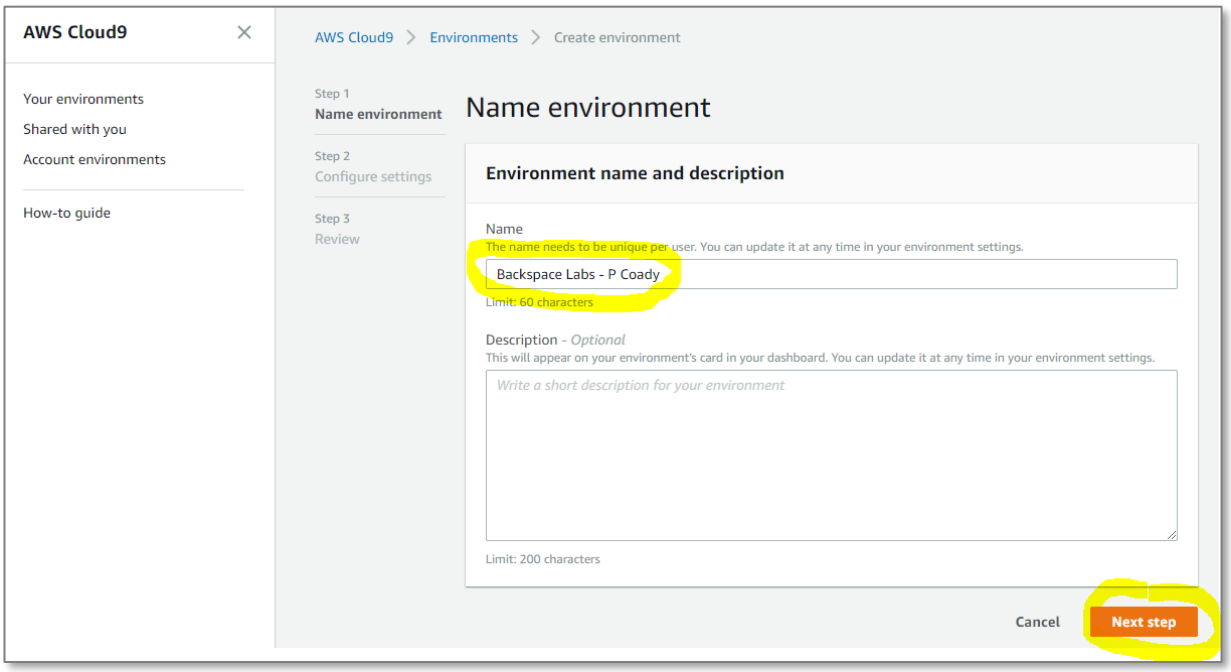
Click 'Cloud9"

Click 'Create Environment"

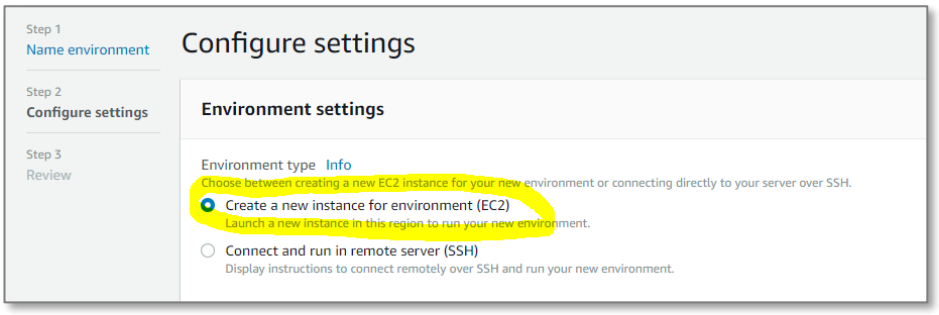


Give your environment a unique name.

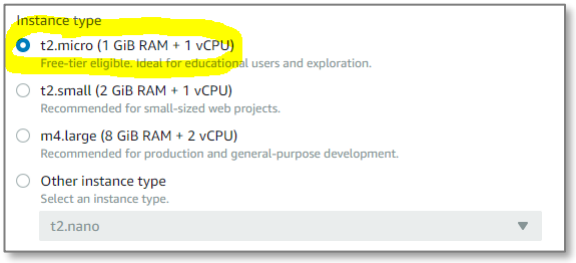
Click 'Next Step"



Select EC2 environment.

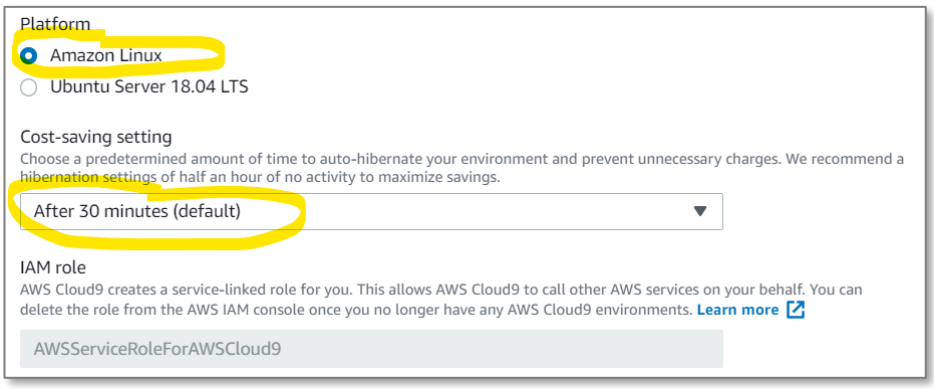


Select t2 micro to stay in the free tier



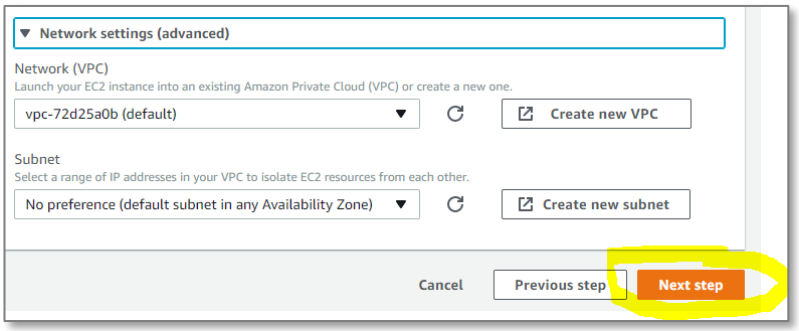
Leave Platform as Amazon Linux. (Amazon Linux has the CLI, NodeJS and Python SDK pre-installed).

Leave hibernation setting at 30 mins

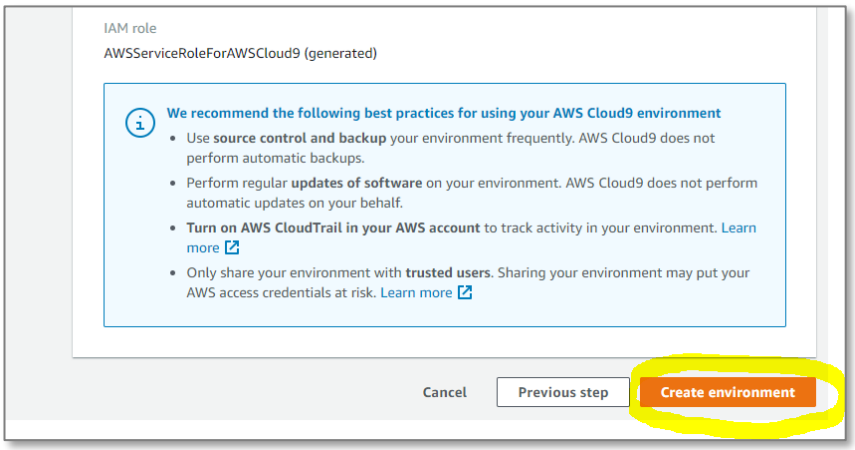


Leave Network settings as default

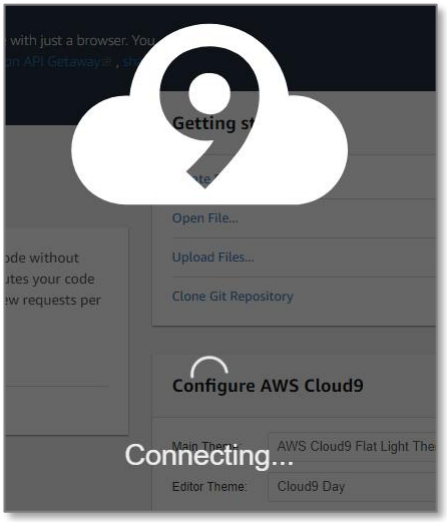
Click 'Next Step"



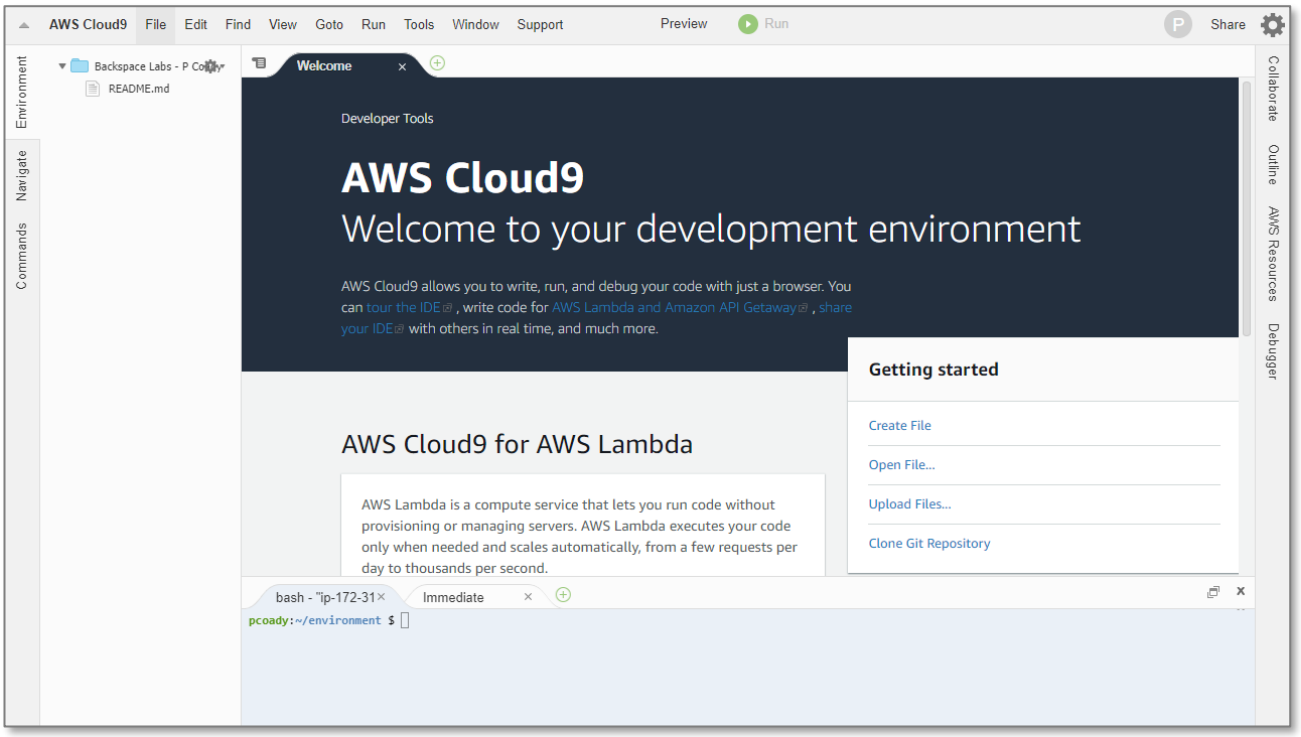
Click 'Create Environment"



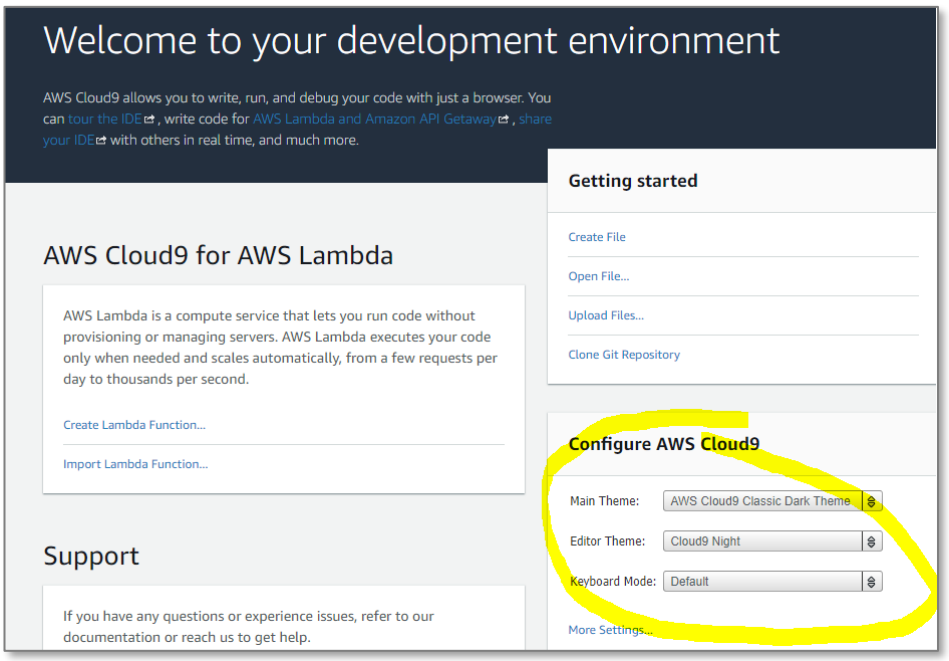
The environment creation process will begin



After some time, your environment will be created.



You can customize the look and feel of the IDE.



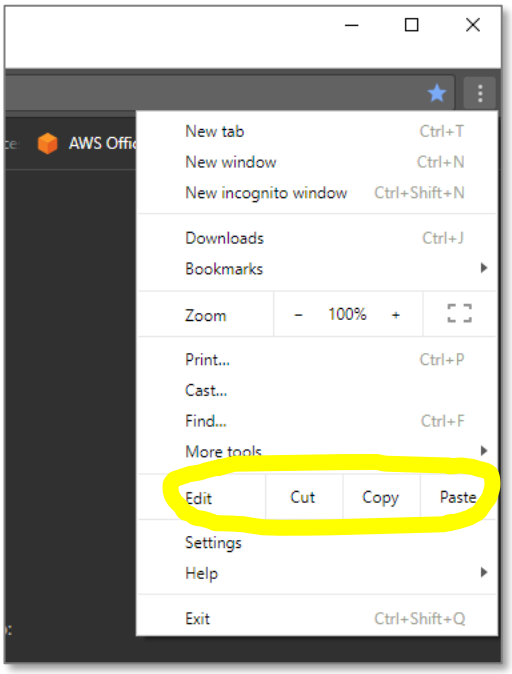
Sending Commands to a Cloud9 EC2 Instance

In this section, we will use the Cloud9 service to send Linux commands to the Cloud9 EC2 instance.

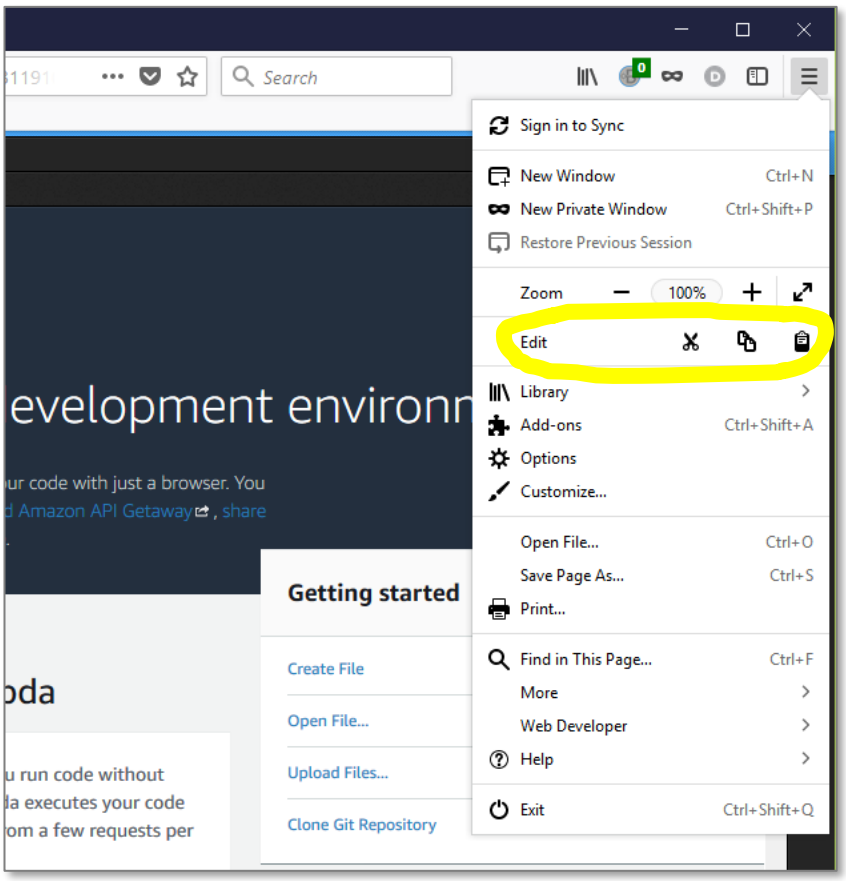
Please note:

Cut and Paste (right click or Cloud9 menu) may not work directly in Cloud9. If you cannot paste into Cloud9 then use the browser paste menu item or use ctrl-v (Windows) / cmd-v (MAC).

e.g. for Chrome:



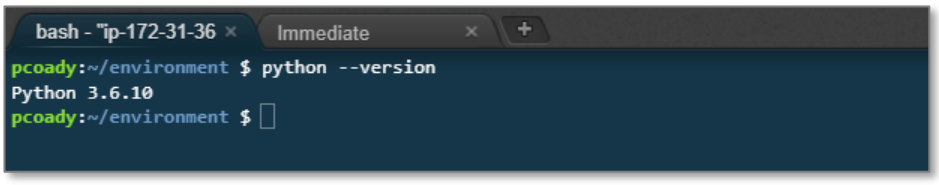
e.g. for Firefox



At the bottom of the screen will be the Linux terminal console pane.

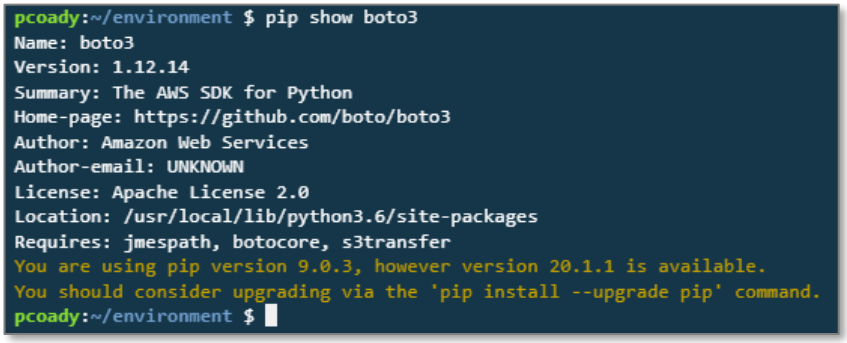
Check that Python is already installed.

```
python --version
```



Check that AWS Python SDK (Boto3) is already installed.

```
pip show boto3
```



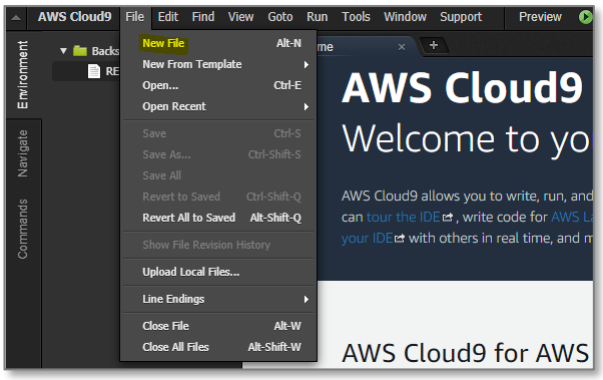
Update Amazon Linux (CentOS) operating system

```
sudo yum update -y
```

Running Code on your Cloud9 EC2 Instance

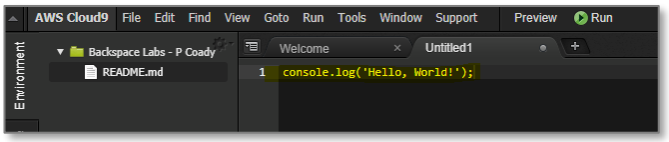
In this section, we will use the Cloud9 service to run Python code on the Cloud9 EC2 instance.

Select 'File" – 'New File"

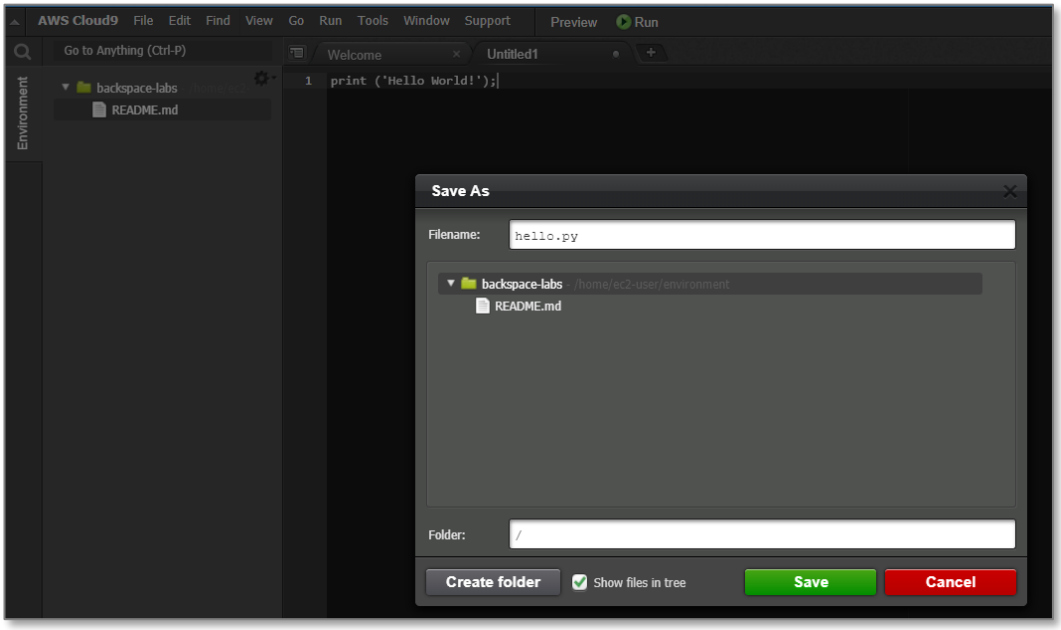


Create the classic 'Hello World ' application

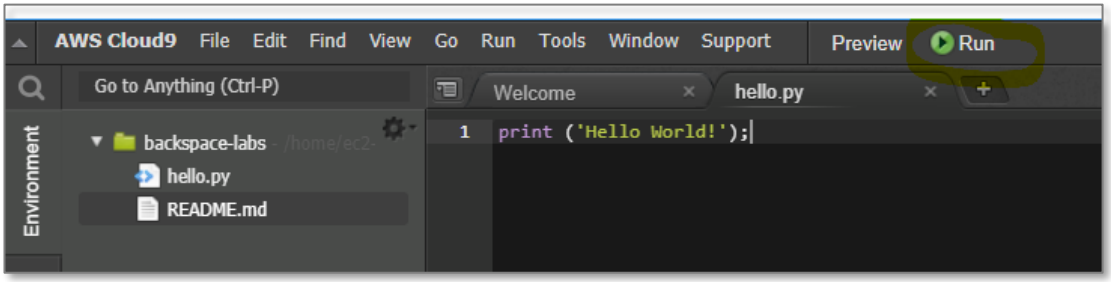
```
print ('Hello World!');
```



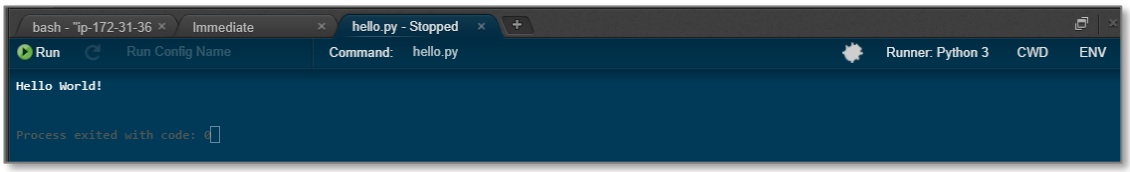
Save as 'hello.py"



Click *Run*



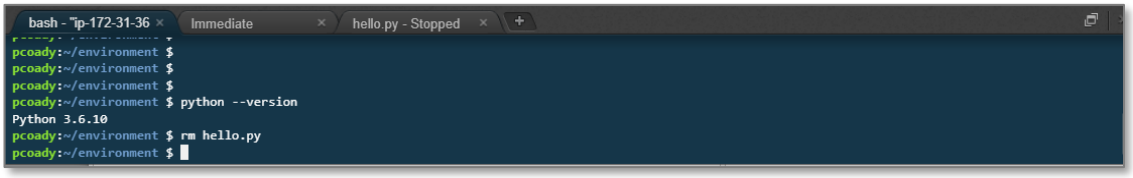
You can see the console output from your application



Close the application tab to remove it from the IDE.

Go back to the Linux console and remove the hello.js file

```
rm hello.py
```

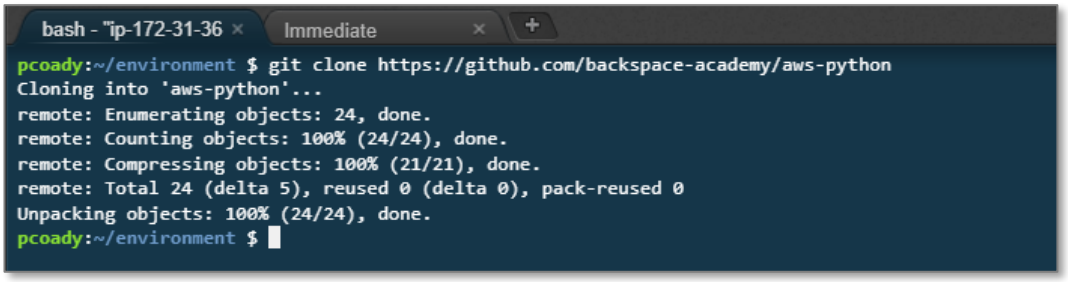


Cloning a GitHub Repository to your Cloud9 EC2 Instance

In this section, we will use the Cloud9 service to clone a GitHub repository to the Cloud9 EC2 instance. We will then run the code in the repository.

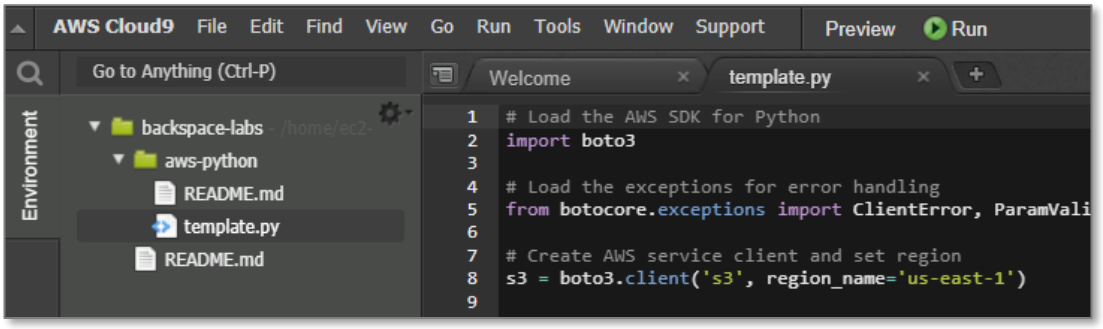
Go back to the Linux console and clone the following sample Python app repository.

```
git clone https://github.com/backspace-academy/aws-cloud9-python
```



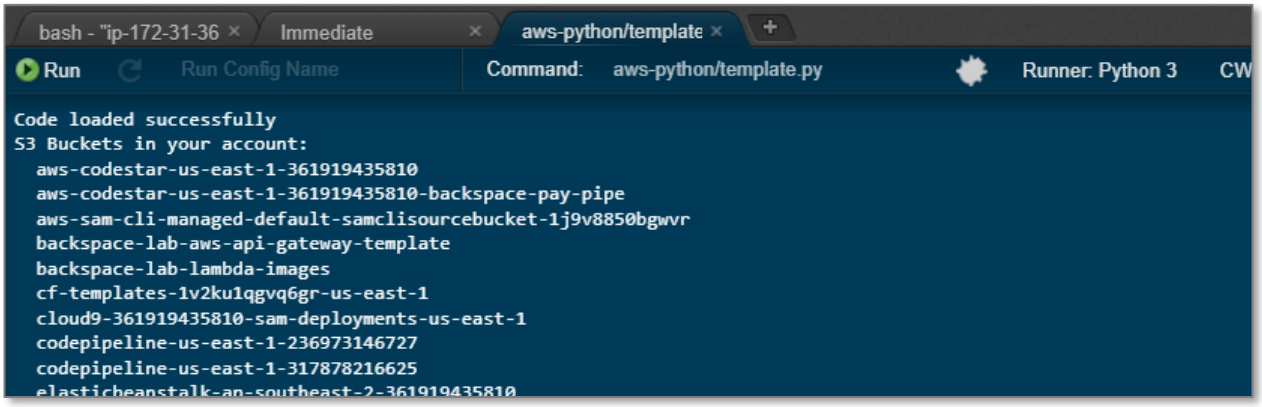
Open *template.py*

Click *Run*



Note: If you have an error stating your credentials have expired, close the run console, refresh the browser, and try again.

You should see a list of buckets in your account in the run console



Editing Files

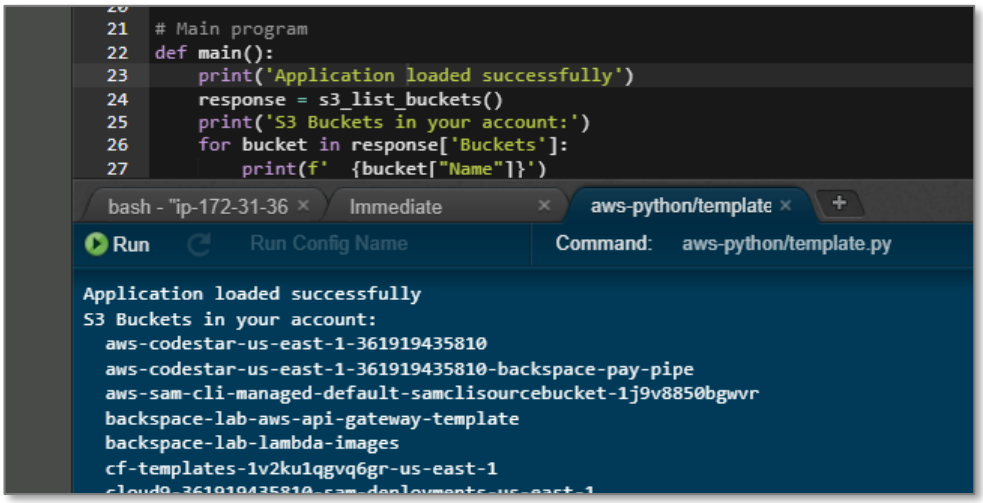
Go back to the *template.py* edit window

Change the print message to 'Application loaded successfully'

Select *File > Save*

Click *Run*

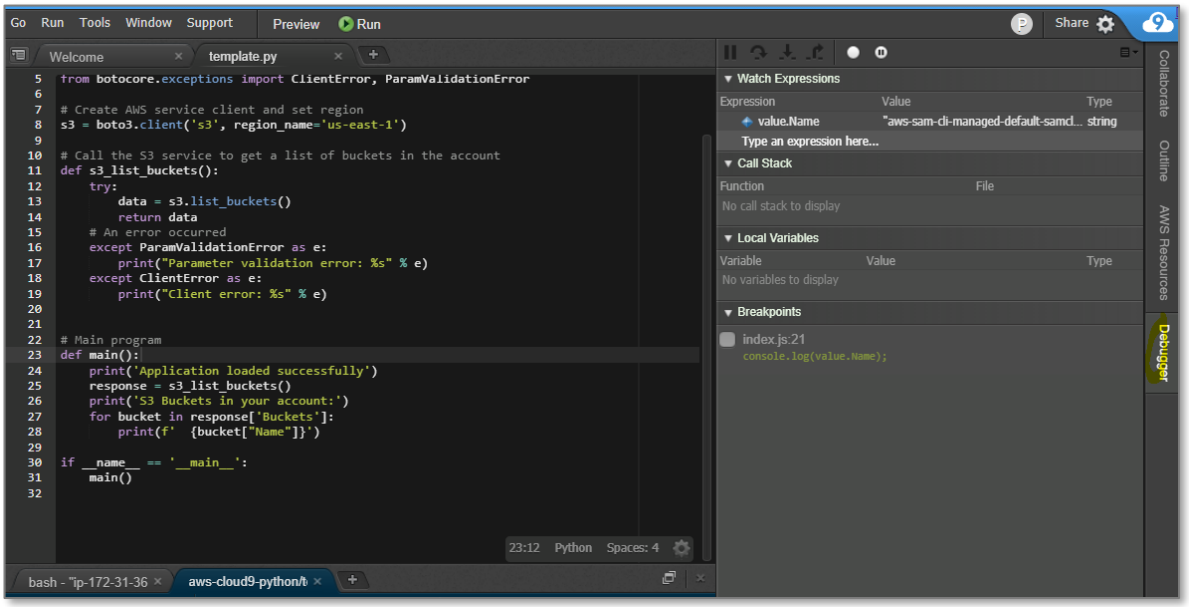
The Linux terminal will show the app running again with the different message



Using the Cloud9 Debugger

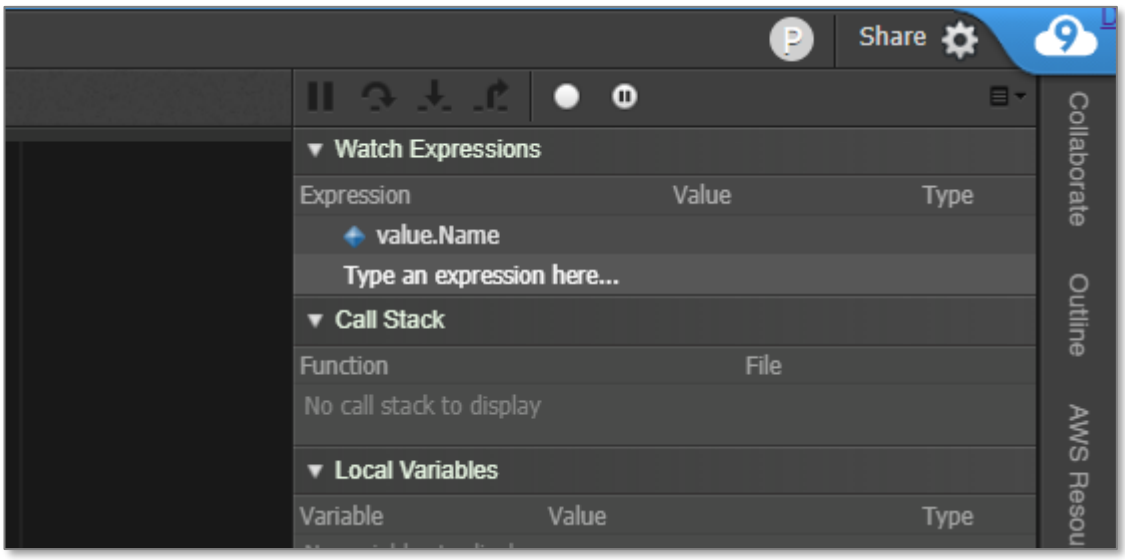
In this section, we will use the Cloud9 debugger to create breakpoints in our code and create a watch expression.

Open the debugger tab on the right hand side



Double click on the *Watch Expressions* input field

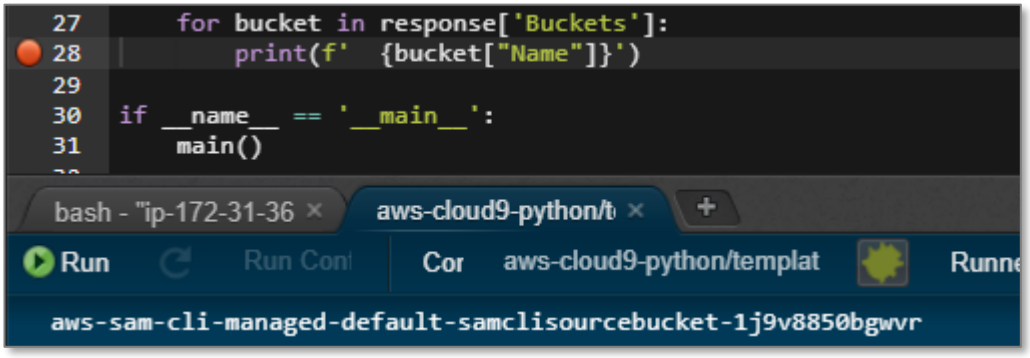
Enter *value.Name*



Click next to the line number for the *console.log(value.Name).* This will create a breakpoint and pause code execution every time a bucket name is output to the console.

A red breakpoint dot will appear next to the line number and the breakpoint will be added to the breakpoints list on the left of the window.

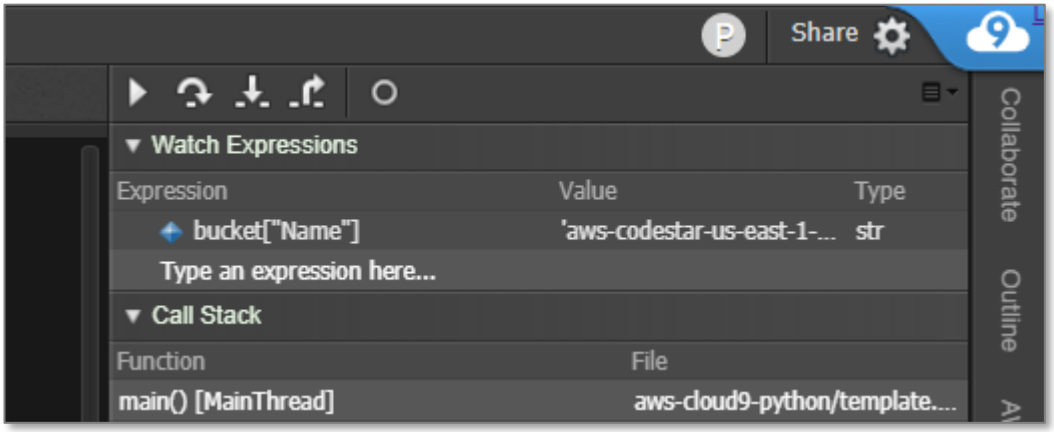
In the terminal window make sure *Run in Debug Mode*  is selected.



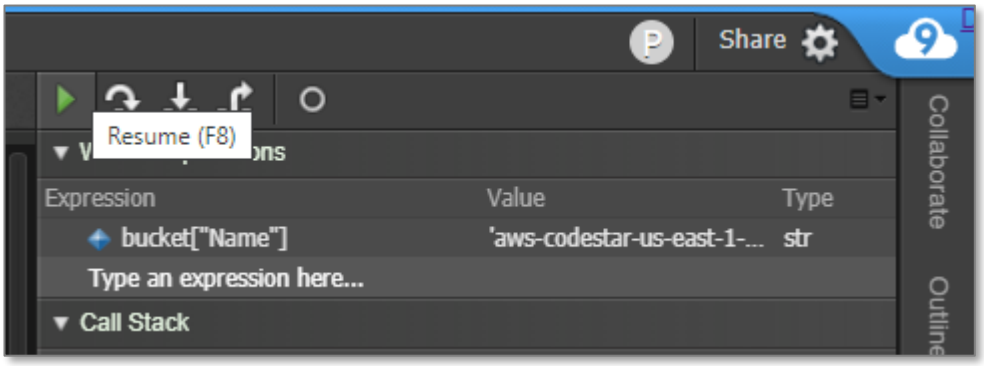
Click *Run*

Note: If you have an error stating your credentials have expired or invalid, refresh the browser to renew the security token, and try again.

The code will run and pause at the breakpoint. The watch expression will have the value of the first basket in the list.

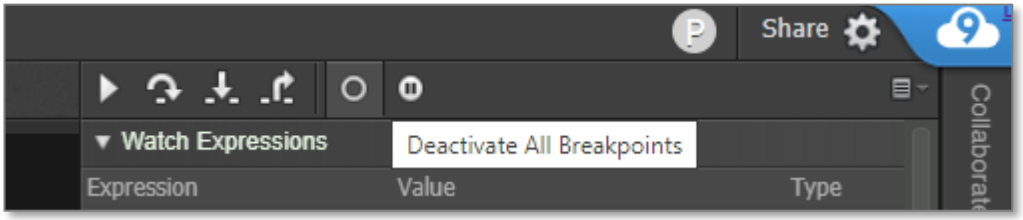


Click the resume button to keep running.



The watch expression value will change.

Click *Deactivate All Breakpoints* to remove the breakpoint.



Click *Resume* again to finish running the code.

Cleaning Up

Your Cloud 9 EC2 development environment will automatically go into hibernation after 30 mins of inactivity. You will not need to terminate it to save costs.

You should clean up the code in environment ready for the next lab by deleting the watch expression, breakpoints and the cloned repository.