



Module	Assessment Type
Distributed and Cloud Systems Programming	Individual Report

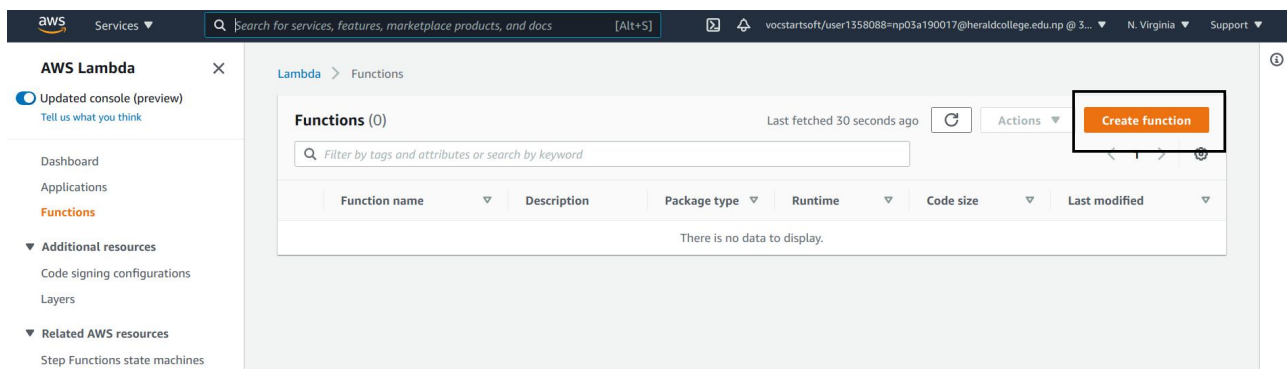
Workshop 1

Student Id : 2049867 (NP03A190017)
Student Name : Roshan Parajuli
Section : L5CG3
Module Leader : Rupak Koirala
Lecturer /Tutor : Saroj Sharma
Submitted on : 2020-03-30

Creating a AWS Lambda “Hello, World!” program.

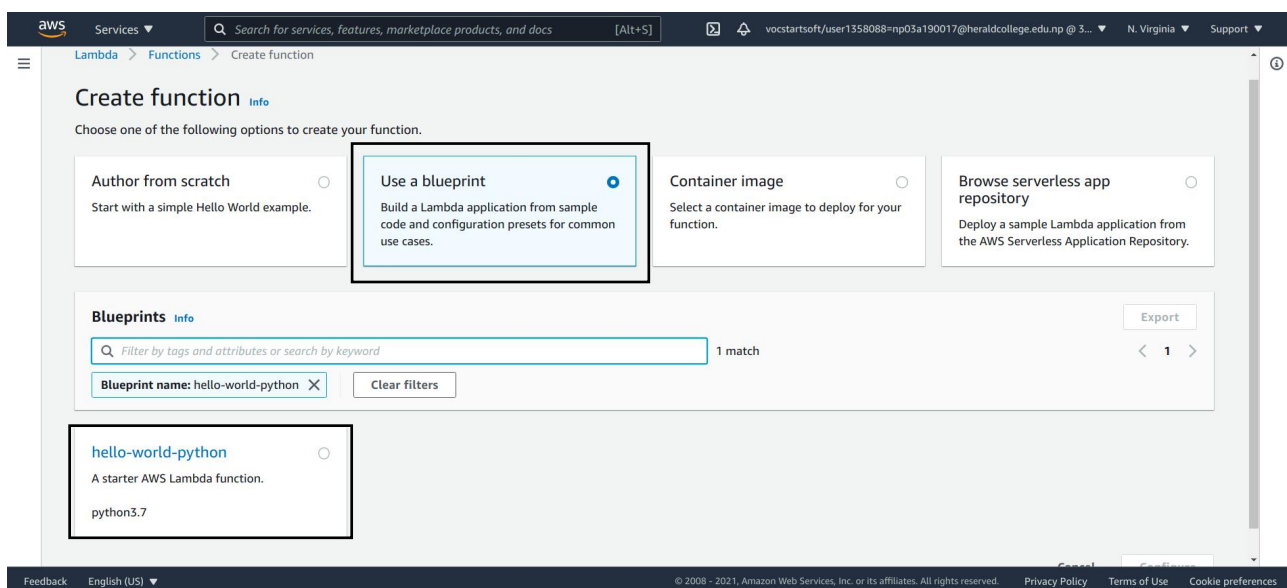
AWS Lambda is a serverless compute service that runs the code in response to events and automatically manages the underlying compute resources for the user. In short, Lambda function is created here to run the code without provisioning and managing the server.

1. Creating the AWS Lambda function



2. Choosing the blueprint

After selecting the lambda service from the AWS search, a new lambda function is made from the existing blueprint here.



3. Creating a new execution role.

Basic information info

Function name
hello-world-python

Execution role
Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#).

☐ Create a new role with basic Lambda permissions

☐ Use an existing role

☒ Create a new role from AWS policy templates

Role creation might take a few minutes. Please do not delete the role or edit the trust or permissions policies in this role.

Role name
Enter a name for your new role.
lambda_basic_execution

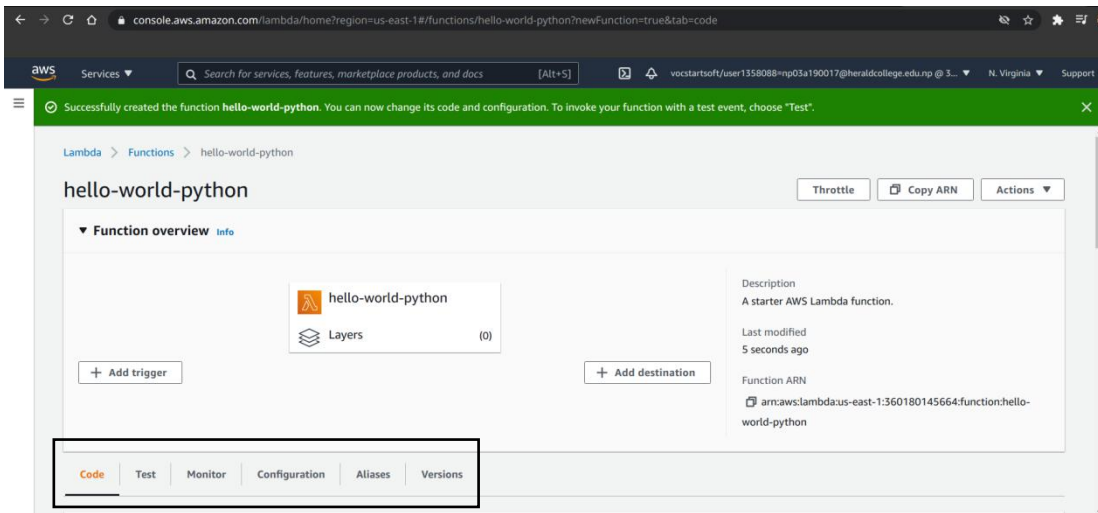
Use only letters, numbers, hyphens, or underscores with no spaces.

Policy templates - optional info

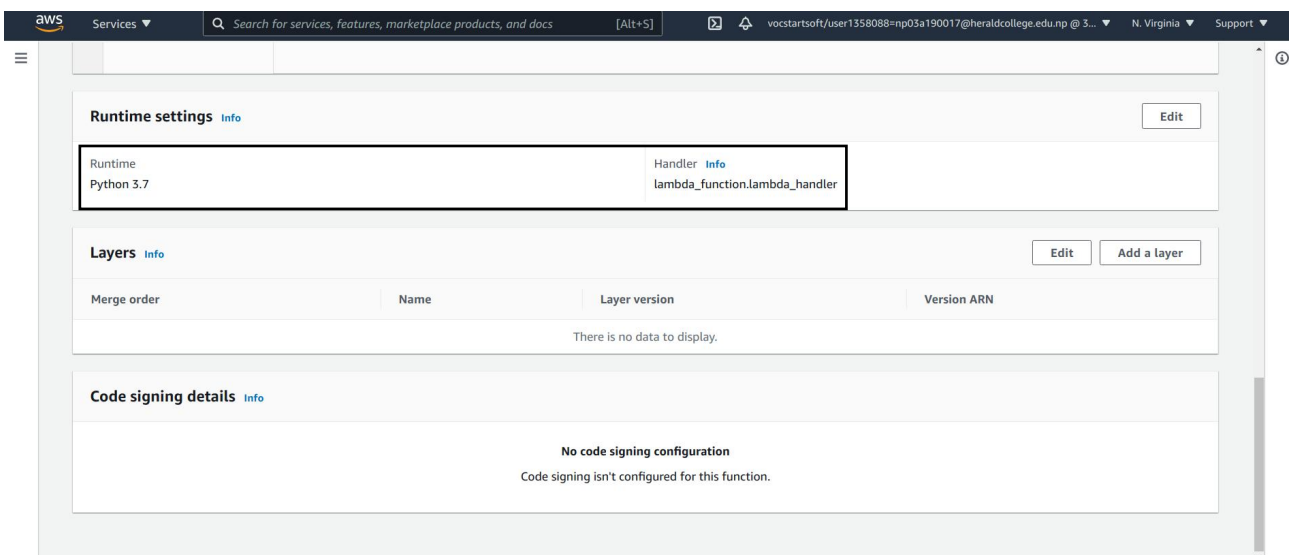
A Lambda function's execution role is an AWS Identity and Access Management (IAM) role that grants the function permission to access AWS services and resources. Here, a new role is made with the name `lambda_basic_execution` to limit as well as grant permissions to the lambda function which is just created.

Role name	Trusted entities	Last activity
<input type="checkbox"/> AWSServiceRoleForAWSCloud9	AWS service: cloud9 (Service-Linked role)	None
<input type="checkbox"/> AWSServiceRoleForCloudWatchEvents	AWS service: events (Service-Linked role)	None
<input type="checkbox"/> AWSServiceRoleForElasticCache	AWS service: elasticache (Service-Linked role)	None
<input type="checkbox"/> AWSServiceRoleForOrganizations	AWS service: organizations (Service-Linked r...	None
<input type="checkbox"/> AWSServiceRoleForSupport	AWS service: support (Service-Linked role)	None
<input type="checkbox"/> AWSServiceRoleForTrustedAdvisor	AWS service: trustedadvisor (Service-Linked ...)	None
<input type="checkbox"/> EMR_AutoScaling_DefaultRole	AWS service: application-autoscaling and 1 more	None
<input type="checkbox"/> EMR_DefaultRole	AWS service: elasticmapreduce	None
<input type="checkbox"/> EMR_EC2_DefaultRole	AWS service: ec2	None
<input checked="" type="checkbox"/> lambda_basic_execution	AWS service: lambda	6 days
<input type="checkbox"/> robomaker_students	AWS service: greengrass and 3 more	None
<input type="checkbox"/> vocareum	Account: 769722020711	You need permissions
<input type="checkbox"/> vocstartsoft	Account: 769722020711	You need permissions

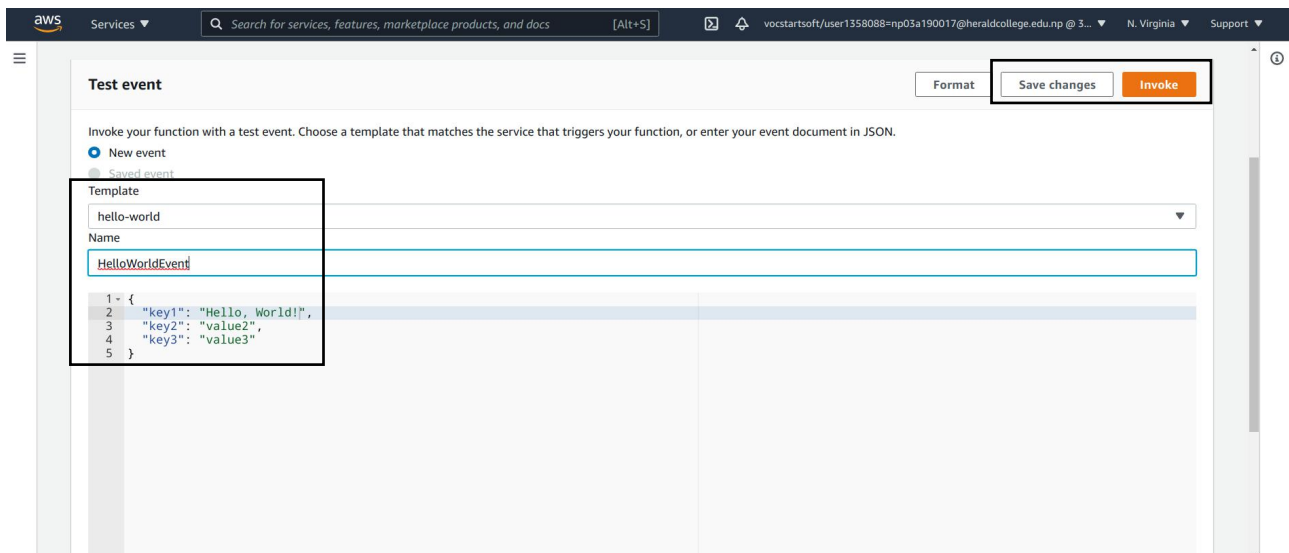
Here, we can alter the permissions of the role which alters the permission of the lambda function as this role is attached to it.



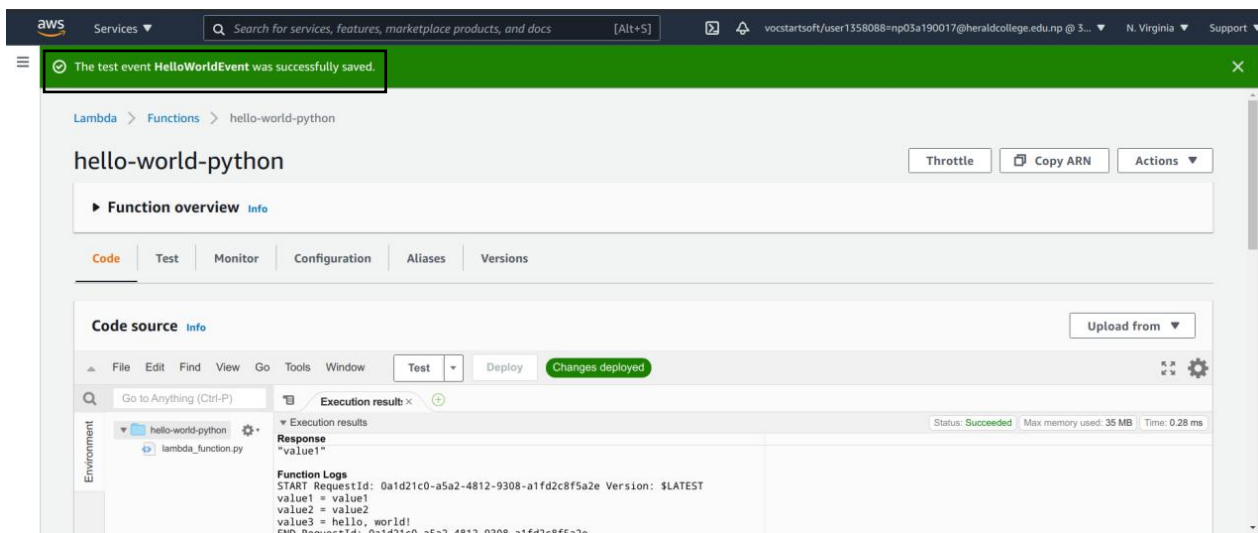
The new lambda function is created with the help of blueprint. This screen acts as a dashboard of Lambda function where its functionalities can be discovered including code, test, monitoring, configuration and so on.



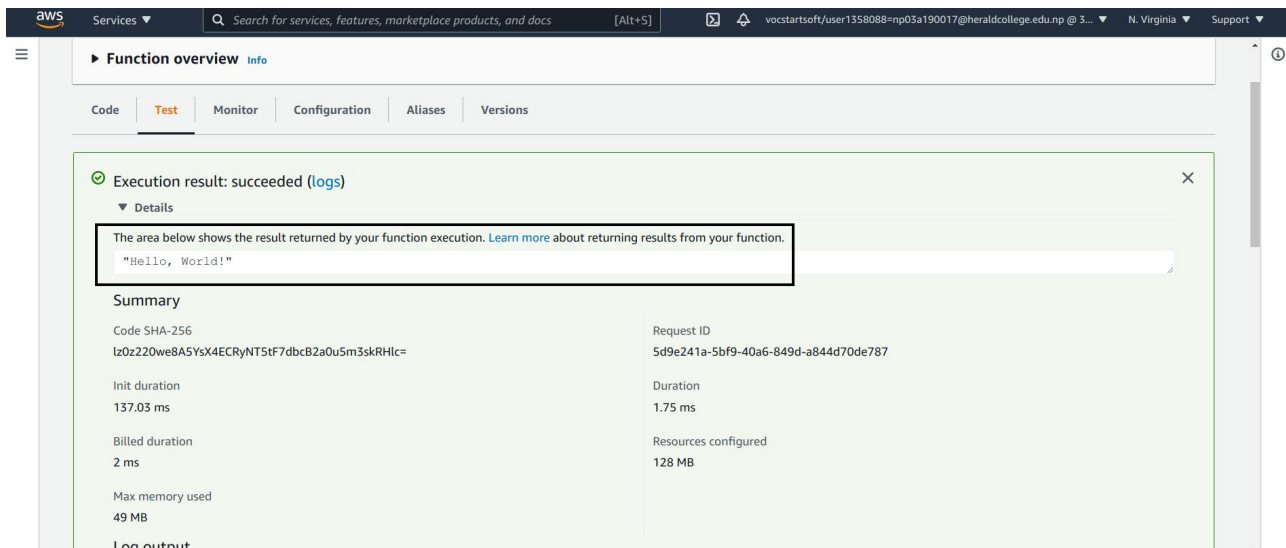
The runtime settings of the lambda function is default i.e. python 3.7 and the handler is also the default lambda_function.lambda_handler.



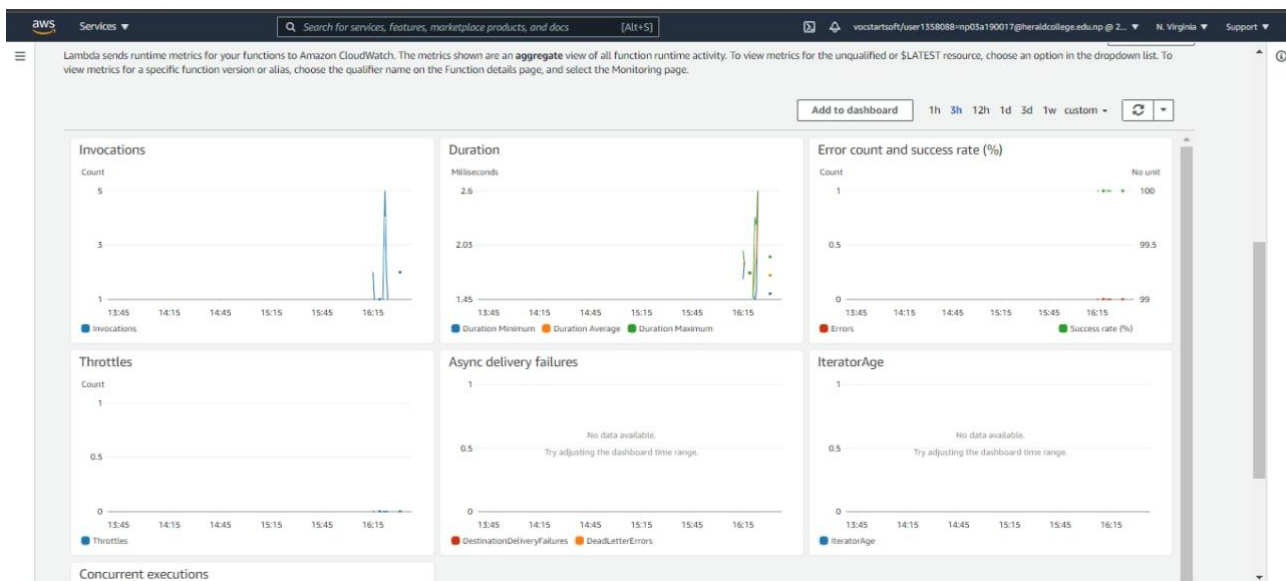
Here, a new event is created namely “HelloWorldEvent” so that it could invoke the lambda function, accomplish the goal and the monitoring could be done.



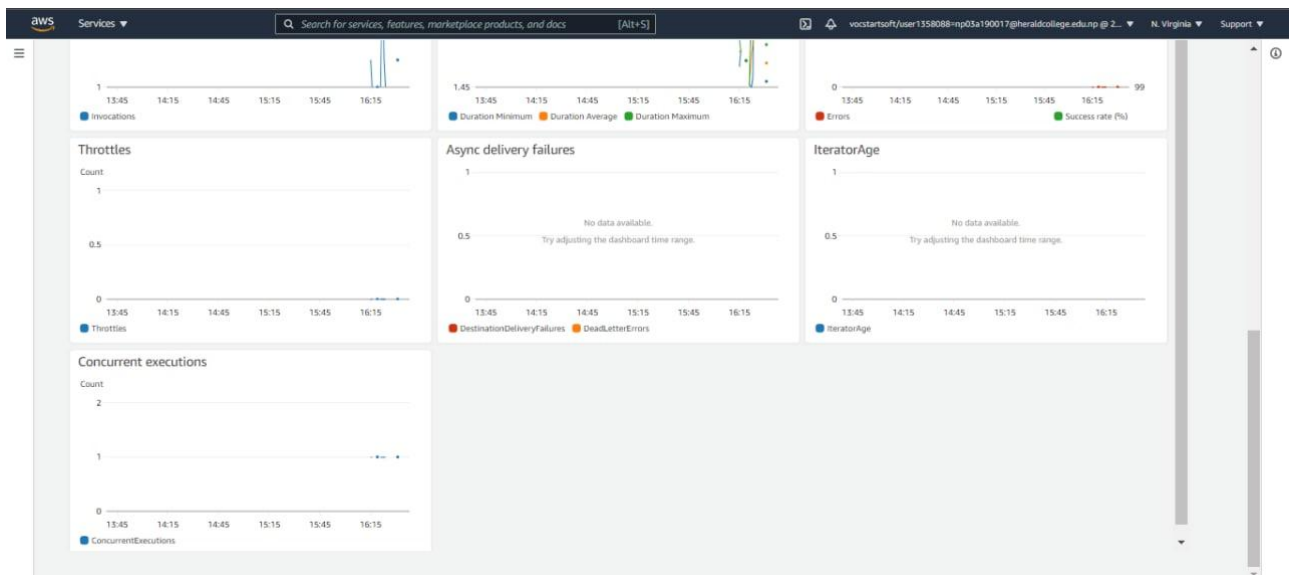
HelloWorldEvent event is successfully saved and now can be invoked multiple number of times.



After invoking the function, the execution result portion shows the summary of the basic attributes. Our goal is achieved i.e. "Hello World" is printed in the screen which was the value of key1 in the json file. We can also see other attributes such as the time it took for initialization (Init duration), the total time it was billed for (billed duration), resources used, etc.





With the help of cloudwatch, we can monitor the details of the lambda function which was just created. It includes properties such as the number of times the function was invoked, the total duration it was invoked for, error count and success rate and a lot of other properties.



hello-world-python

▼ Function overview Info

 hello-world-python

 Layers (0)

+ Add trigger

+ Add destination

Description
A starter AWS Lambda function.

Last modified
53 minutes ago

Function ARN
arn:aws:lambda:us-east-1:220065220104:function:hello-world-python

Throttle

Copy ARN

Actions

- Publish new version
- Create alias
- Export function
- Capabilities
- Deploy to Lambda@Edge
- Delete function**

Finally, the lambda function was deleted as a part of the workshop. The deletion of lambda function does not affect the IAM role which would still be present and can be assigned to another function.