Serverless computing:

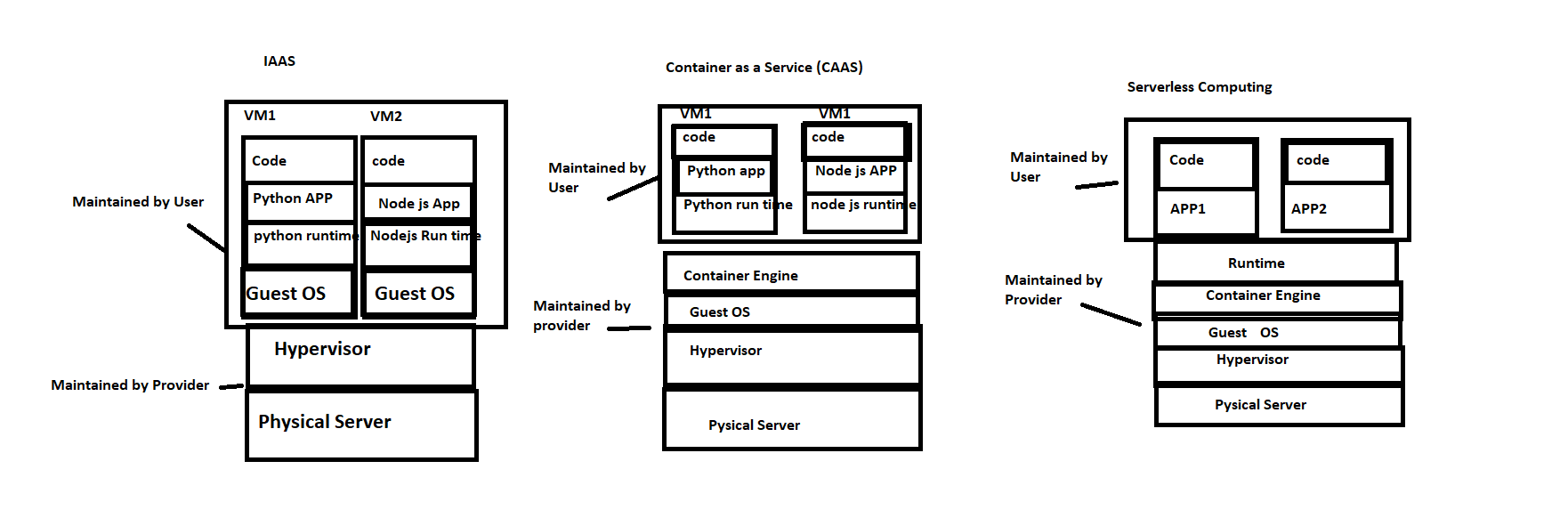
1. Serverless computing allows us to create, deploy and run applications without maintaining our own servers.
2. In serverless computing we basically focus on writing code but not on infrastructure.
3. Serverless computing means server will be there but we are not allowed to maintain. Maintenance is taken care by the provider

For example:

If I want to run a Python application in EC2, then I want to

1. Setup the network
2. Select the OS
3. Storage
4. Create an EC2 instance and connect to it.
5. Install Python environment, dependencies that need to run my application.

IAAS vs Container vs Serverless



IAAS: OS updates and problems are taken care by user who created and using those VM, all OS maintenance is taken care by user only.

Container Engine: Here OS related issues are maintained by service provider. Container maintenance should be taken care by user.

Example: EKS , ECS are the services for container engine.

Serverless Computing: Only App and code is maintained by user & rest of the things are maintained by service provider.

Serverless Services in different Cloud computing service provider

* AWS: Lambda
* Azure: Function
* GCP: functions

Pricing in Serverless computing:

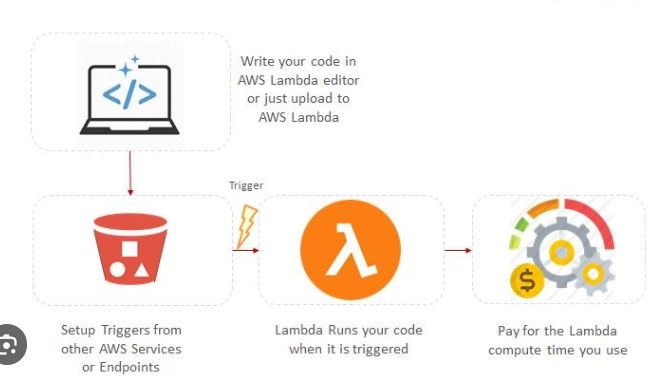
In EC2 & CAAS we need to start the service and we will be charged until we stop that service.

But in serverless, we don’t need to maintain or create any servers, we just need to write code and remaining things will be taken care by service provider and we will be charged for the time in which our code is executed.

What is AWS Lambda?

AWS lambda is a serverless compute service in AWS in which we just supply our code to perform task.

How lambda works?



Lambda default timeout is 3 seconds

Lambda maximum timeout is 15 minutes

Lambda service runs your function only when needed & scales automatically, you need to pay for the compute time.

Benefits:

1. No maintenance
2. Pay as you use
3. High availability
4. Scalability
5. Monitoring and logging

Lambda function code:



In lambda function we have 2 parameters

1. event
2. context

**event**: The data from the event that triggered.

**context**: the data about the execution environment. By default it holds the data for 15 minutes.

Create a lambda function with in AWS console:

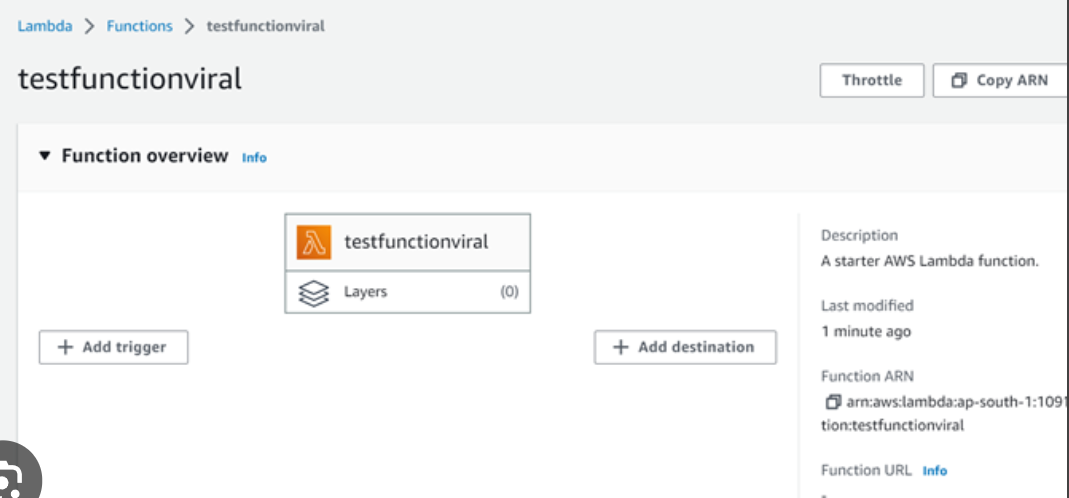
1. Search for lambda in AWS console.
2. Click on functions
3. Click on create function
4. Select author from scratch or use a blue print or container image (select a container image to deploy your function)
5. Basic information
6. Function name
7. Select run time (python 3.9, 3.8, 3.7 etc)
8. Select architecture (X86-64, arm64)
9. In permissions, we need to mention the IAM role.

Because when we run the lambda function, the code gets executed and some logs are generated & stored in cloud watch service for this we need to have access to IAM role.

1. Execution role
2. Create a new role with basic lambda permission
3. Use an existing role
4. Create a new role from AWS policy templates

Select any one of the execution role on your choice.

1. Now click on create function
2. Now your function is created and you will be navigated to the function page.

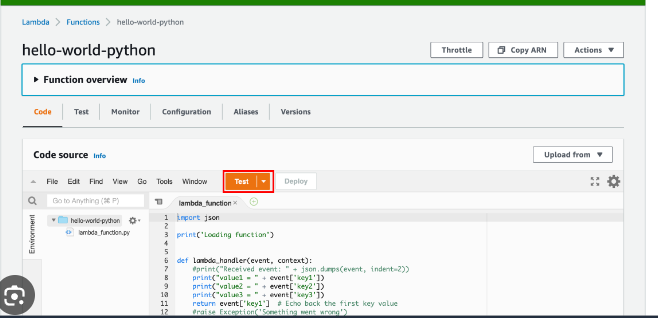


What are layers in Lambda?

1. A lambda layer is a .zip file archive that can contain additional dependencies and code.
2. My lambda code in python doesn’t have all dependencies which need to execute my code

If I want to install some dependencies then I will zip all those into a zip file & add it to the code as a layer

1. If I add package in my code, then my code size will become huge. This delays the code execution.
2. 50 mb is the max code limit. So, we need to put all the dependencies in layers.
3. When I want to upload code file then I need to zip the files & upload it using Upload from button on right side.



Adding Layers:

1. Select Layers from left side panel
2. Click on create layer
3. Enter layer name
4. You can upload the file in 2 ways.
5. You can upload using a .zip file
6. Upload dependencies file from amazon s3

If your file is larger than 10 mb consider uploading it using S3.

1. Select the compatible architecture. (X86\_64, arm64)
2. Select compatible runtime (Python 3.9, 3.8, 3.7, 3.6)
3. Click on create

Now your layer is created. So, you can select your function & click on layers.

Scroll down to layers & click on Add a layer.

1. Choose a layer
2. Aws layers
3. Custom layers – select this
4. Specify an ARN
5. Select from layers
6. Click on Add
7. You can add up to 5 layers not more than 5.
8. Layer size should be less than 250 mb

To remove a layer:

1. Click on edit layer
2. Select the layer and click on remove.

Triggers in Lambda:

We all know lambda follows event driven approach. Which means whenever an event is triggered then our lambda executes.

Here we can place S3, RDS, Dynamo DB, API as a trigger.

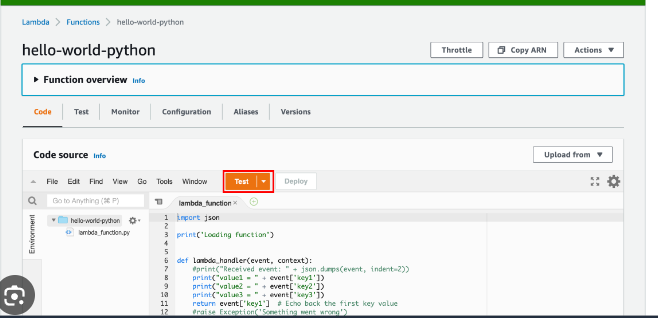
Example:

S3 event notification.

When ever a file is dropped in the s3 bucket then my lambda should be triggered.

Steps:

1. Create a lambda function
2. To read the file in s3 bucket we need to go for configuration.

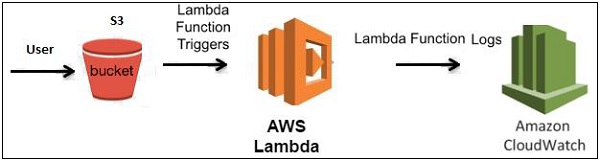


And then permissions, and give permission to the execution role.

1. S3 sends the metadata of the file which was dropped in bucket to the event parameter
2. Now navigate to lambda function, click on add trigger.
3. Select the trigger service as S3
4. Select bucket
5. Select event type
6. All object create events
7. Put -> Object update
8. Post -> Object created
9. Copy -> object copied

Here in event type, we need to mention, upon which event lambda should be triggered.

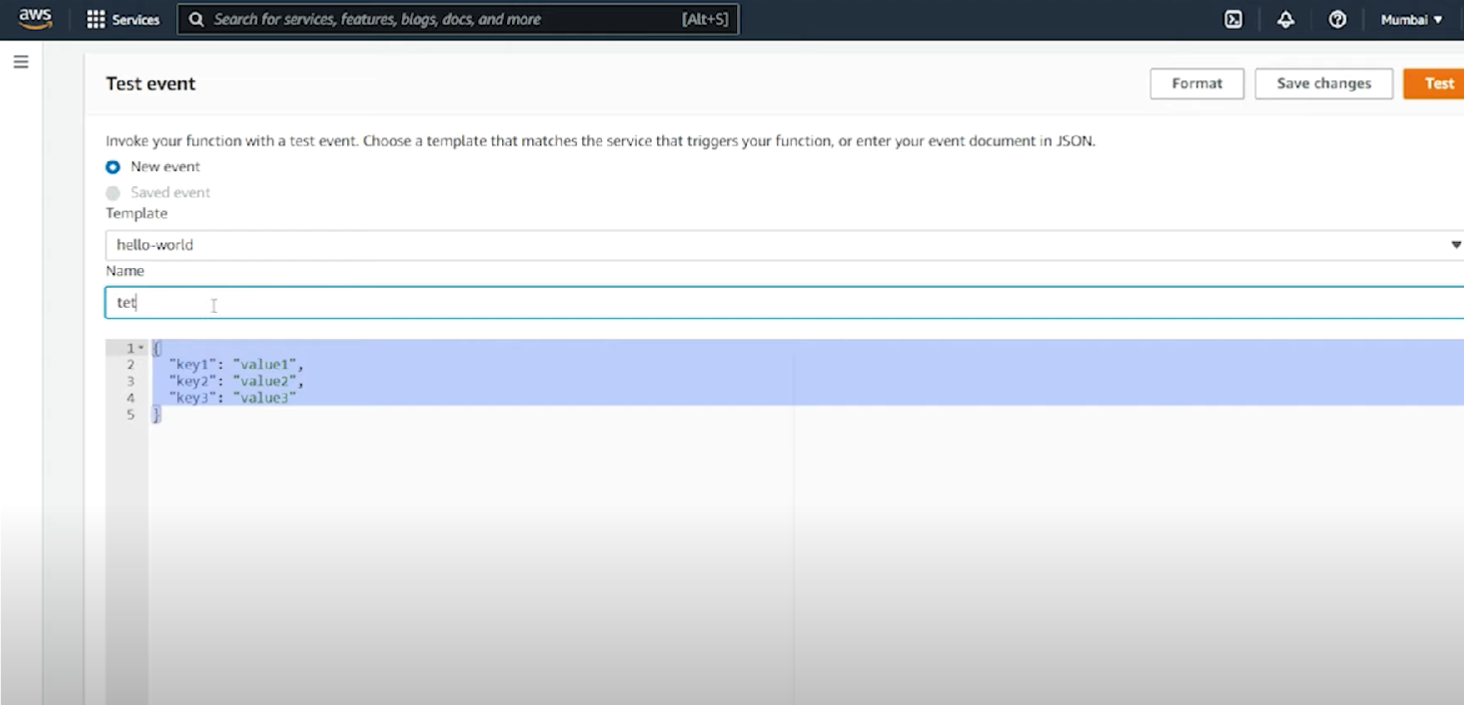
1. Prefix (optional): you can specify the folder in the bucket
2. Suffix(optional) : If I want to trigger my lambda when only a .png file is dropped then I want to provide suffix as .png.
3. Check the I acknowledge and click on Add.
4. Now your trigger is added you can check the trigger details in configurations tab -> triggers.



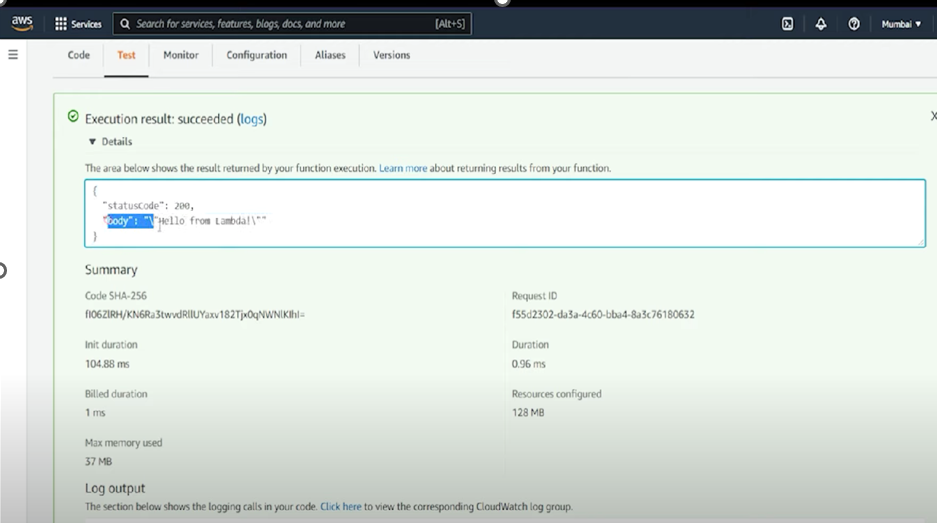
On-demand triggering of lambda function?

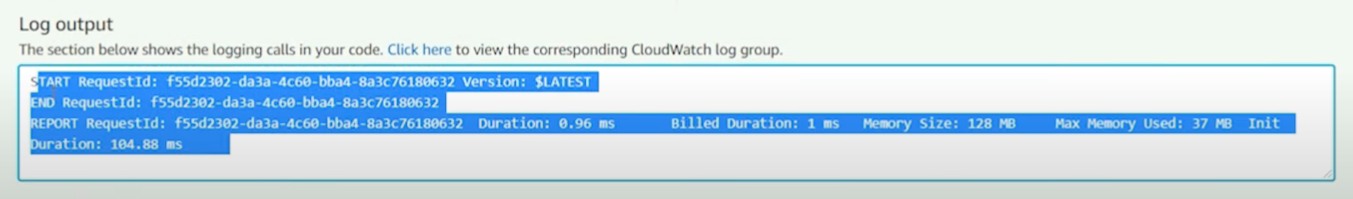
* To execute this function. We have to move to Test area.
* Lambda functions are event base.
* If some event occurs then only our lambda will execute.
* So, for every lambda we need to pass an event.

1. Click on Test and select the new event.



And click on Test you can see the execution result and logs.

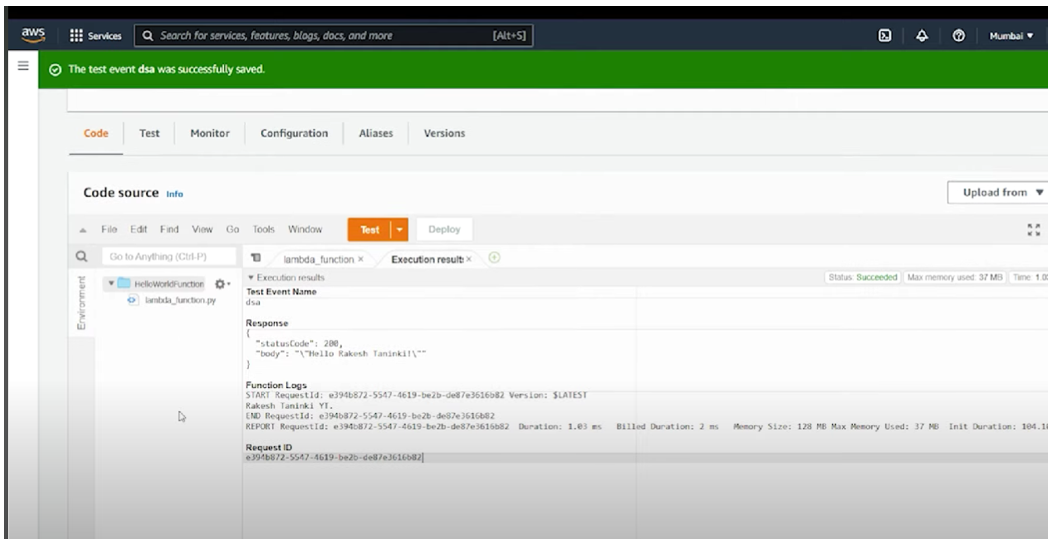




If you change/ modify code in lambda function, then you need to click on deploy. Then only our code gets modified.

To execute this lambda function, we can click on **Test button**.

If you click on Test button, then it will asks for event name and event and click on Test.



If you navigate to Monitor tab, then you can able to watch the logs

