

Assignment No- 15

Create a serverless computing service.

1. Search lambda, create function, give function name. Select on Enable function url in additional configurations.

Create function [Info](#)

Choose one of the following options to create your function.

☒ **Author from scratch**
Start with a simple Hello World example.

☐ **Use a blueprint**
Build a Lambda application from sample code and configuration presets for common use cases.

☐ **Container image**
Select a container image to deploy for your function.

Basic information

Function name
Enter a name that describes the purpose of your function.

Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, 0-9, hyphens (-), and underscores (_).

Runtime [Info](#)
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.

Architecture [Info](#)
Choose the instruction set architecture you want for your function code.
☒ x86_64
☐ arm64

Permissions [Info](#)
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.
[Change default execution role](#)

Additional Configurations
Use additional configurations to set up code signing, function URL, tags, and Amazon VPC access for your function.
☐ **Enable Code signing** [Info](#)
Use code signing configurations to ensure that the code has been signed by an approved source and has not been altered since signing.
☐ **Enable encryption with an AWS KMS customer managed key** [Info](#)
By default, Lambda encrypts the .zip file archive using an AWS owned key.
☒ **Enable function URL** [Info](#)
Use function URLs to assign HTTP(S) endpoints to your Lambda function.

Additional Configurations
Use additional configurations to set up code signing, function URL, tags, and Amazon VPC access for your function.
☐ **Enable Code signing** [Info](#)
Use code signing configurations to ensure that the code has been signed by an approved source and has not been altered since signing.
☐ **Enable encryption with an AWS KMS customer managed key** [Info](#)
By default, Lambda encrypts the .zip file archive using an AWS owned key.
☒ **Enable function URL** [Info](#)
Use function URLs to assign HTTP(S) endpoints to your Lambda function.

Auth type
Choose the auth type for your function. [Learn more](#)
☐ **AWS_IAM**
Only authenticated IAM users and roles can make requests to your function URL.
☒ **NONE**
Lambda won't perform IAM authentication on requests to your function URL. The URL endpoint will be public unless you implement your own authorization logic in your function.

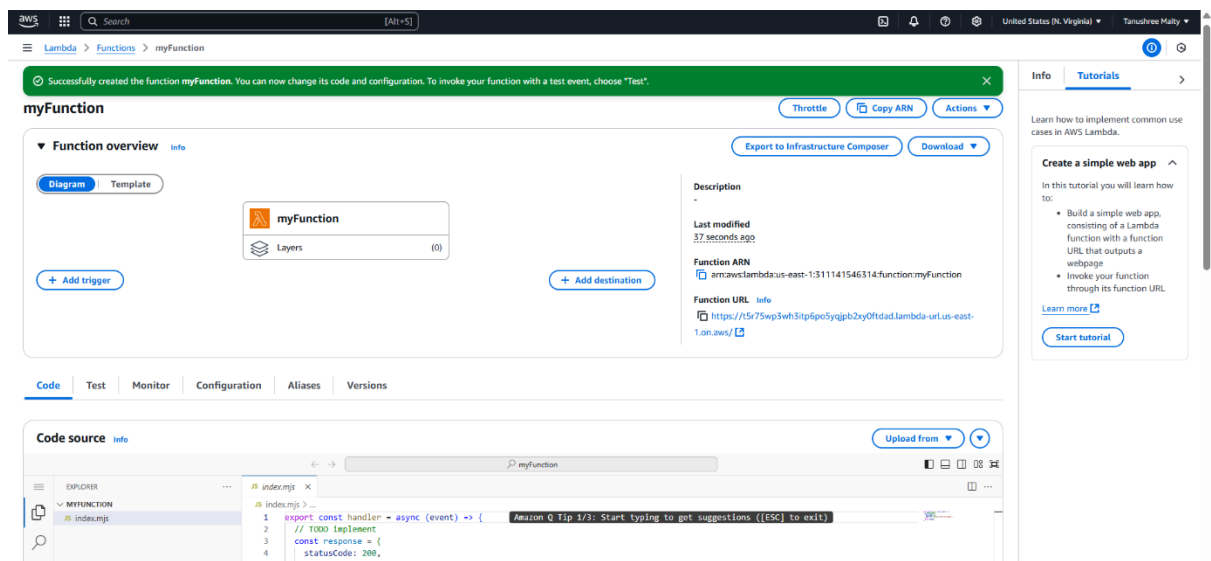
Function URL permissions

ⓘ When you choose auth type **NONE**, Lambda automatically creates the following resource-based policy and attaches it to your function. This policy makes your function public to anyone with the function URL. You can edit the policy later. To limit access to authenticated IAM users and roles, choose auth type **AWS_IAM**.

View policy statement

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "StatementId": "FunctionURLAllowPublicAccess",  
6       "Effect": "Allow",  
7       "Principal": "*",  
8       "Action": "lambda:InvokeFunctionUrl",  
9       "Resource": "arn:aws:lambda:us-east-1:311141546314:function:myFunction",  
10      "Condition": {  
11        "StringEquals": {  
12          "lambda:FunctionUrlAuthType": "NONE"  
13        }  
14      }  
15    }  
16  ]  
}
```

- Copy Function URL and paste it on new tab.



- Output is.

