# Serverless Computing (b)

## 1. Introduction

Serverless computing allows developers to build applications without managing infrastructure. AWS Lambda can be created and deployed using the AWS CLI, enabling automation and quick deployment. This assignment demonstrates creating a simple "Hello World" Lambda function using AWS CLI on Linux.

## 2. Steps to Create AWS Lambda Function via CLI

* Step 1: Install and Configure AWS CLI

Ensure AWS CLI v2 is installed. Configure it with your credentials using:  
aws configure  
Enter your Access Key, Secret Key, region (e.g., us-east-1), and output format (e.g., json).

* Step 2: Create Python Function File

Create a file named lambda\_function.py with the following code:

* Step 3: Zip the Python File

zip function.zip lambda\_function.py  
This creates a zip file function.zip to upload to AWS Lambda.

* Step 4: Create IAM Role (if not already created)

Create a role with basic Lambda execution permissions. Save the Role ARN for the next step.

* Step 5: Create Lambda Function

aws lambda create-function --function-name HelloLambdaCLI --runtime python3.9 --role <Your-Role-ARN> --handler lambda\_function.lambda\_handler --zip-file fileb://function.zip  
Replace <Your-Role-ARN> with your actual IAM Role ARN.

* Step 6: Test Lambda Function

Invoke the Lambda function:  
aws lambda invoke --function-name HelloLambdaCLI output.txt  
Check the file output.txt to see the output: {"statusCode": 200, "body": "Hello from AWS Lambda via Console!"}

## 3. Python Code

def lambda\_handler(event, context):  
 return {  
 'statusCode': 200,  
 'body': ‘Hello, World!  
 }

## 4. CLI Output Screenshot

[A screen shot of a computer

AI-generated content may be incorrect.

## 5. Conclusion

This assignment demonstrates creating and deploying a Lambda function using AWS CLI on Linux. Using the zipped file method ensures that larger, more complex functions can be deployed easily and automated in scripts.