

## Lesson 08 Demo 03

### Creating a Serverless Web App

**Objective:** To create a serverless web application on the Amazon Web Services (AWS) platform

**Tools required:** AWS WorkSpaces

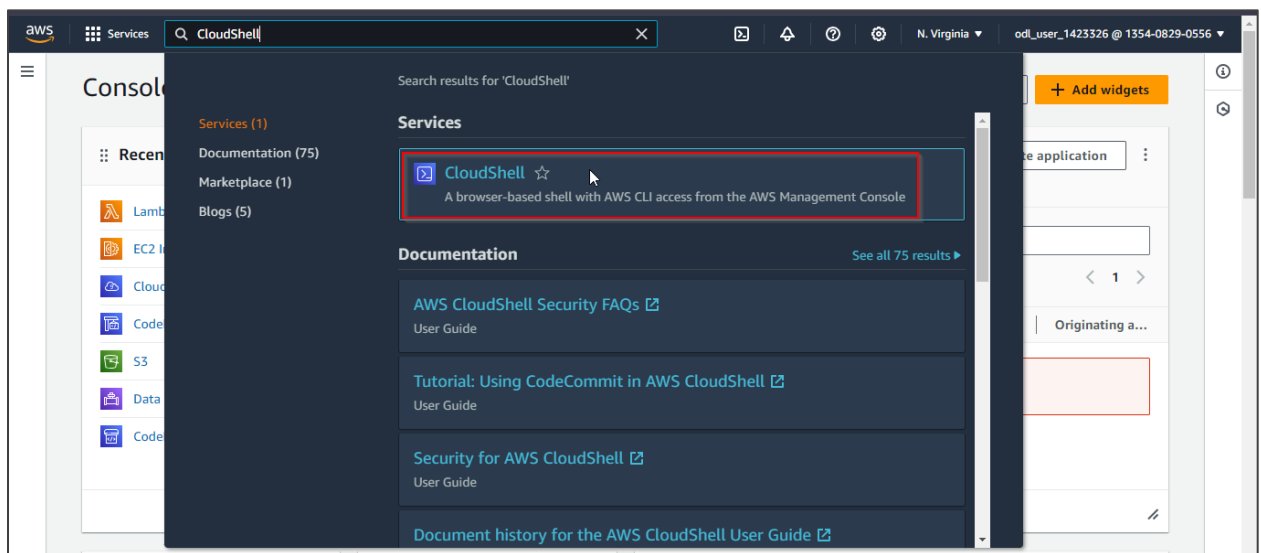
**Prerequisites:** None

Steps to be followed:

1. Develop a serverless web application

#### Step 1: Develop a serverless web application

- 1.1 On the AWS portal home screen, search for and select **CloudShell**



1.2 In the CloudShell, use the following command:

**sam init**



```
CloudShell
us-east-1 +
[cloudshell-user@ip-10-136-46-27 ~]$ sam init

SAM CLI now collects telemetry to better understand customer needs.

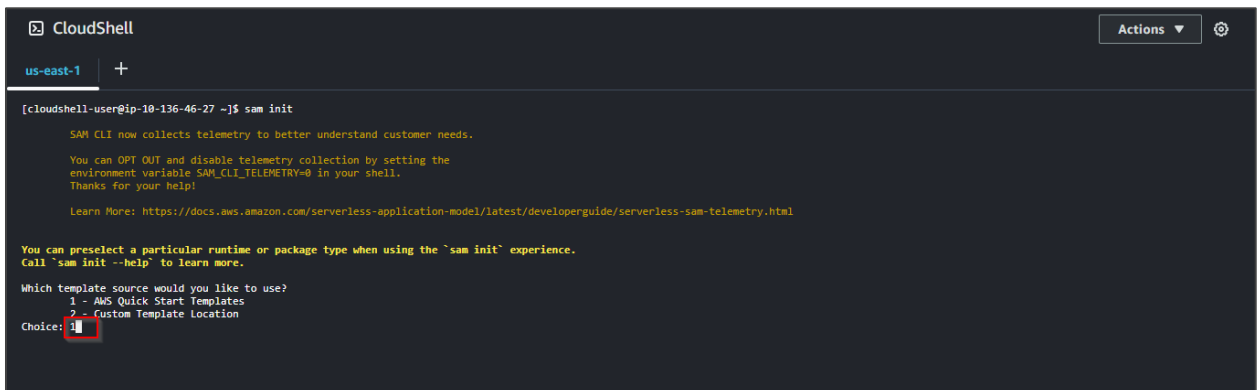
You can OPT OUT and disable telemetry collection by setting the
environment variable SAM_CLI_TELEMETRY=0 in your shell.
Thanks for your help!

Learn More: https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-telemetry.html

You can preselect a particular runtime or package type when using the `sam init` experience.
Call `sam init --help` to learn more.

Which template source would you like to use?
 1 - AWS Quick Start Templates
 2 - Custom Template Location
Choice: |
```

1.3 Choose option 1 for the AWS Quick Start Templates



```
CloudShell
us-east-1 +
[cloudshell-user@ip-10-136-46-27 ~]$ sam init

SAM CLI now collects telemetry to better understand customer needs.

You can OPT OUT and disable telemetry collection by setting the
environment variable SAM_CLI_TELEMETRY=0 in your shell.
Thanks for your help!

Learn More: https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-telemetry.html

You can preselect a particular runtime or package type when using the `sam init` experience.
Call `sam init --help` to learn more.

Which template source would you like to use?
 1 - AWS Quick Start Templates
 2 - Custom Template Location
Choice: 1
```

## 1.4 Select option 1 for the Hello World Example template

```

CloudShell
us-east-1 +

You can OPT OUT and disable telemetry collection by setting the
environment variable SAM_CLI_TELEMETRY=0 in your shell.
Thanks for your help!

Learn More: https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-telemetry.html

You can preselect a particular runtime or package type when using the `sam init` experience.
Call `sam init --help` to learn more.

Which template source would you like to use?
 1 - AWS Quick Start Templates
 2 - Custom Template Location
Choice: 1

Choose an AWS Quick Start application template
 1 - Hello World Example
 2 - Data processing
 3 - Hello World Example with Powertools for AWS Lambda
 4 - Multi-step workflow
 5 - Scheduled task
 6 - Standalone function
 7 - Serverless API
 8 - Infrastructure event management
 9 - Lambda Response Streaming
10 - Serverless Connector Hello World Example
11 - Multi-step workflow with Connectors
12 - GraphQL API Hello World Example
13 - Full Stack
14 - Lambda EFS example
15 - DynamoDB Example
16 - Machine Learning
Template: 1
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```

## 1.5 Enter n for Python and the zip package

```

aws Services Search [Alt+S] N. Virginia odl_user_1423326 @ 1354-0829-0556
CloudShell
us-east-1 +

Thanks for your help!

Learn More: https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-telemetry.html

You can preselect a particular runtime or package type when using the `sam init` experience.
Call `sam init --help` to learn more.

Which template source would you like to use?
 1 - AWS Quick Start Templates
 2 - Custom Template Location
Choice: 1

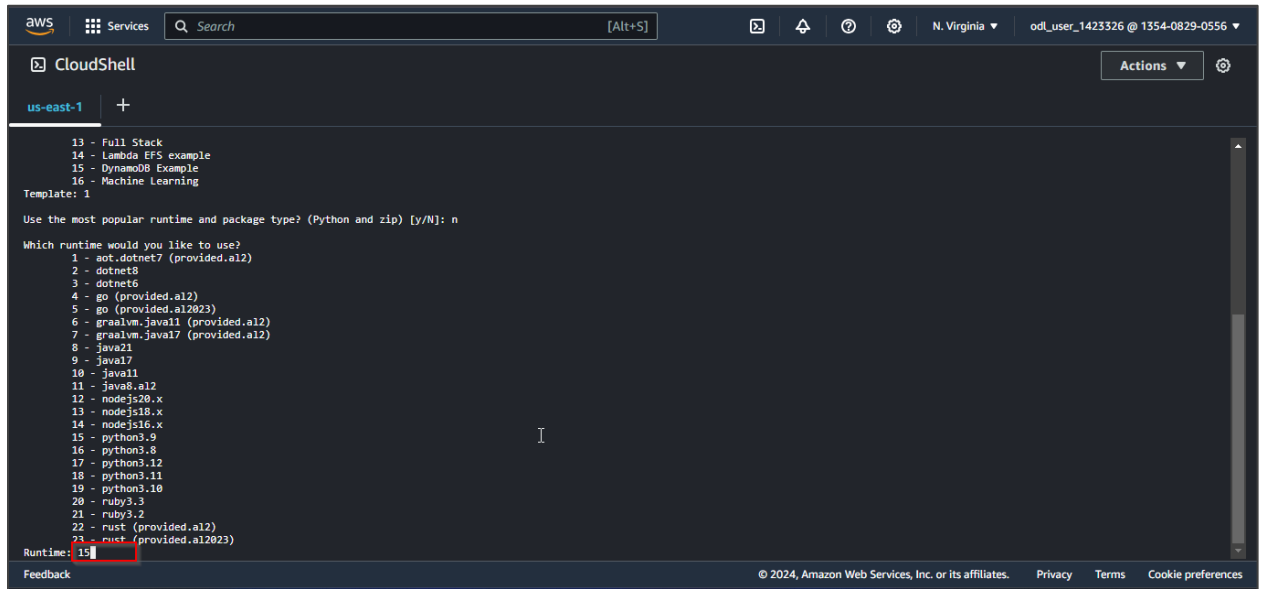
Choose an AWS Quick Start application template
 1 - Hello World Example
 2 - Data processing
 3 - Hello World Example with Powertools for AWS Lambda
 4 - Multi-step workflow
 5 - Scheduled task
 6 - Standalone function
 7 - Serverless API
 8 - Infrastructure event management
 9 - Lambda Response Streaming
10 - Serverless Connector Hello World Example
11 - Multi-step workflow with Connectors
12 - GraphQL API Hello World Example
13 - Full Stack
14 - Lambda EFS example
15 - DynamoDB Example
16 - Machine Learning
Template: 1

Use the most popular runtime and package type? (Python and zip) [y/N]: n
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

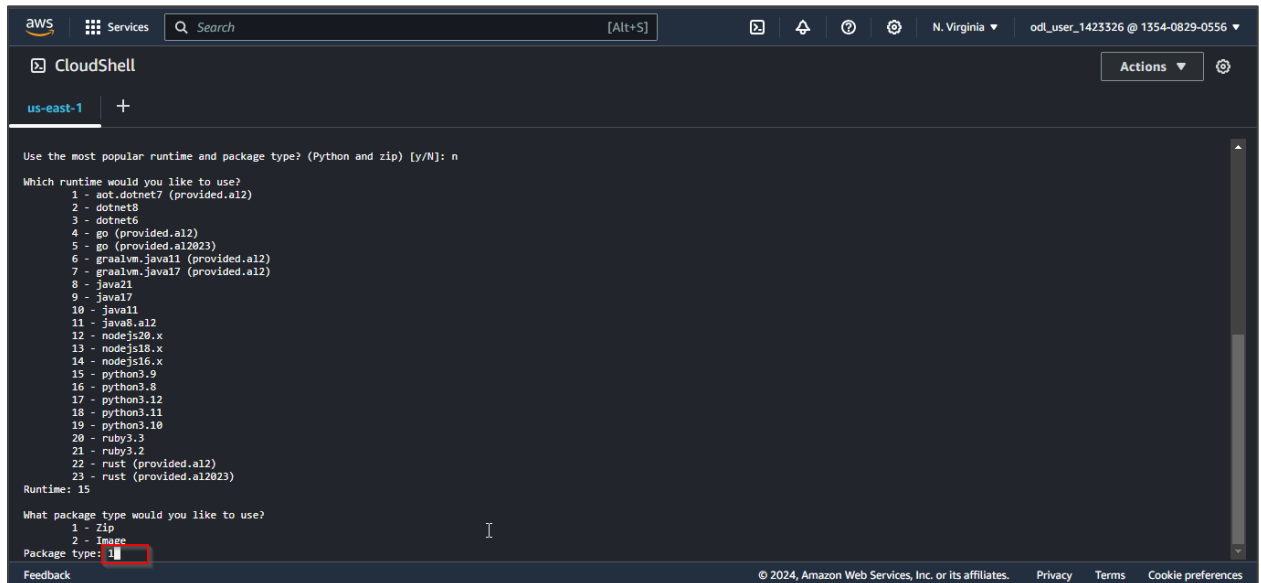
```

Opt for the non-zipped package and choose Python when prompted.

## 1.6 Set the runtime to Python 3.9 (Choose 15)



## 1.7 Choose package type 1, which is Zip



## 1.8 Enable X-ray tracing by entering y

```

CloudShell
us-east-1 +

2 - dotnet8
3 - dotnet6
4 - go (provided.al2)
5 - go (provided.al2023)
6 - graalvm.java11 (provided.al2)
7 - graalvm.java17 (provided.al2)
8 - java21
9 - java17
10 - java11
11 - java8.al2
12 - nodejs20.x
13 - nodejs18.x
14 - nodejs16.x
15 - python3.9
16 - python3.8
17 - python3.12
18 - python3.11
19 - python3.10
20 - ruby3.3
21 - ruby3.2
22 - rust (provided.al2)
23 - rust (provided.al2023)

Runtime: 15

What package type would you like to use?
1 - Zip
2 - Image
Package type: 1

Based on your selections, the only dependency manager available is pip.
We will proceed copying the template using pip.

Would you like to enable X-Ray tracing on the function(s) in your application? [y/N]: y

```

## 1.9 Disable CloudWatch monitoring by entering N

```

CloudShell
us-east-1 +

6 - graalvm.java11 (provided.al2)
7 - graalvm.java17 (provided.al2)
8 - java21
9 - java17
10 - java11
11 - java8.al2
12 - nodejs20.x
13 - nodejs18.x
14 - nodejs16.x
15 - python3.9
16 - python3.8
17 - python3.12
18 - python3.11
19 - python3.10
20 - ruby3.3
21 - ruby3.2
22 - rust (provided.al2)
23 - rust (provided.al2023)

Runtime: 15

What package type would you like to use?
1 - Zip
2 - Image
Package type: 1

Based on your selections, the only dependency manager available is pip.
We will proceed copying the template using pip.

Would you like to enable X-Ray tracing on the function(s) in your application? [y/N]: y
X-Ray will incur an additional cost. View https://aws.amazon.com/xray/pricing/ for more details

Would you like to enable monitoring using CloudWatch Application Insights?
For more info, please view https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/cloudwatch-application-insights.html [y/N]: N

```

## 1.10 Disable Structured Logging by entering N

```

CloudShell
us-east-1 +

8 - java21
9 - java17
10 - java11
11 - java8.a12
12 - nodejs20.x
13 - nodejs18.x
14 - nodejs16.x
15 - python3.9
16 - python3.8
17 - python3.12
18 - python3.11
19 - python3.10
20 - ruby3.3
21 - ruby3.2
22 - rust (provided.a12)
23 - rust (provided.a12023)
Runtime: 15

What package type would you like to use?
1 - Zip
2 - Image
Package type: 1

Based on your selections, the only dependency manager available is pip.
We will proceed copying the template using pip.

Would you like to enable X-Ray tracing on the function(s) in your application? [y/N]: y
X-Ray will incur an additional cost. View https://aws.amazon.com/xray/pricing/ for more details

Would you like to enable monitoring using CloudWatch Application Insights?
For more info, please view https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/cloudwatch-application-insights.html [y/N]: N

Would you like to set Structured Logging in JSON format on your Lambda functions? [y/N]: N

```

## 1.11 Enter test1 as the Project name (the default is sam-app)

```

AWS
Services Search [Alt+S] N. Virginia odl_user_1423326 @ 1354-0829-0556

CloudShell
us-east-1 +

10 - java11
11 - java8.a12
12 - nodejs20.x
13 - nodejs18.x
14 - nodejs16.x
15 - python3.9
16 - python3.8
17 - python3.12
18 - python3.11
19 - python3.10
20 - ruby3.3
21 - ruby3.2
22 - rust (provided.a12)
23 - rust (provided.a12023)
Runtime: 15

What package type would you like to use?
1 - Zip
2 - Image
Package type: 1

Based on your selections, the only dependency manager available is pip.
We will proceed copying the template using pip.

Would you like to enable X-Ray tracing on the function(s) in your application? [y/N]: y
X-Ray will incur an additional cost. View https://aws.amazon.com/xray/pricing/ for more details

Would you like to enable monitoring using CloudWatch Application Insights?
For more info, please view https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/cloudwatch-application-insights.html [y/N]: N

Would you like to set Structured Logging in JSON format on your Lambda functions? [y/N]: N

Project name [sam-app]: test1

```

1.12 Run the below command:

**cd test1**

The screenshot shows the AWS CloudShell interface. At the top, it says 'us-east-1' with a plus icon. Below that, a series of prompts are shown: 'Would you like to enable X-Ray tracing on the function(s) in your application? [y/N]: y', 'Would you like to enable monitoring using CloudWatch Application Insights? [y/N]: N', and 'Would you like to set Structured Logging in JSON format on your Lambda functions? [y/N]: N'. The 'Project name [sam-app]: test1' is entered. Then, the 'Generating application:' section shows details: Name: test1, Runtime: python3.9, Architectures: x86\_64, Dependency Manager: pip, Application Template: hello-world, Output Directory: ., and Configuration file: test1/samconfig.toml. Below this, 'Next steps can be found in the README file at test1/README.md'. A section 'Commands you can use next' lists three commands: '[\*] Create pipeline: cd test1 && sam pipeline init --bootstrap', '[\*] Validate SAM template: cd test1 && sam validate', and '[\*] Test Function in the Cloud: cd test1 && sam sync --stack-name {stack-name} --watch'. The prompt '[cloudshell-user@ip-10-136-46-27 Python-3.12.0]\$ cd test1' is shown with 'cd test1' highlighted by a red box.

1.13 Now, run the below command to build the project:

**sam build**

This screenshot is identical to the previous one, showing the same prompts and output in the AWS CloudShell. The prompt '[cloudshell-user@ip-10-136-46-27 Python-3.12.0]\$ cd test1' is shown. The prompt '[cloudshell-user@ip-10-136-46-27 test1]\$ sam build' is shown with 'sam build' highlighted by a red box.

```

CloudShell
us-east-1 +

Next steps can be found in the README file at test1/README.md

Commands you can use next
=====
[*] Create pipeline: cd test1 && sam pipeline init --bootstrap
[*] Validate SAM template: cd test1 && sam validate
[*] Test Function in the Cloud: cd test1 && sam sync --stack-name {stack-name} --watch

[cloudshell-user@ip-10-136-46-27 Python-3.12.0]$ cd test1
[cloudshell-user@ip-10-136-46-27 test1]$ sam build
Starting Build use cache
Manifest file is changed (new hash: 3298f13049d19cfaa37ca931dd4d421) or dependency folder (.aws-sam/deps/73c3ac14-4ad2-4b46-bdaf-5b1122ea2183) is missing for (HelloWorldFunction), downloading
dependencies and copying/building source
Building codeuri: /home/cloudshell-user/Python-3.12.0/test1/hello_world runtime: python3.9 metadata: {} architecture: x86_64 functions: HelloWorldFunction
Running PythonPipBuilder:CleanUp
Running PythonPipBuilder:ResolveDependencies
Running PythonPipBuilder:CopySource
Running PythonPipBuilder:CopySource
Build Succeeded

Built Artifacts  : .aws-sam/build
Built Template   : .aws-sam/build/template.yaml

Commands you can use next
=====
[*] Validate SAM template: sam validate
[*] Invoke Function: sam local invoke
[*] Test Function in the Cloud: sam sync --stack-name {(stack-name)} --watch
[*] Deploy: sam deploy --guided
[cloudshell-user@ip-10-136-46-27 test1]$
  
```

The build is successful.

1.14 Execute the command below and provide the values as shown in the screenshot:  
**sam deploy --guided**

```

CloudShell
us-east-1 +

Next steps can be found in the README file at test1/README.md

Commands you can use next
=====
[*] Create pipeline: cd test1 && sam pipeline init --bootstrap
[*] Validate SAM template: cd test1 && sam validate
[*] Test Function in the Cloud: cd test1 && sam sync --stack-name {stack-name} --watch

[cloudshell-user@ip-10-136-46-27 Python-3.12.0]$ cd test1
[cloudshell-user@ip-10-136-46-27 test1]$ sam build
Starting Build use cache
Manifest file is changed (new hash: 3298f13049d19cfaa37ca931dd4d421) or dependency folder (.aws-sam/deps/73c3ac14-4ad2-4b46-bdaf-5b1122ea2183) is missing for (HelloWorldFunction), downloading
dependencies and copying/building source
Building codeuri: /home/cloudshell-user/Python-3.12.0/test1/hello_world runtime: python3.9 metadata: {} architecture: x86_64 functions: HelloWorldFunction
Running PythonPipBuilder:CleanUp
Running PythonPipBuilder:ResolveDependencies
Running PythonPipBuilder:CopySource
Running PythonPipBuilder:CopySource
Build Succeeded

Built Artifacts  : .aws-sam/build
Built Template   : .aws-sam/build/template.yaml

Commands you can use next
=====
[*] Validate SAM template: sam validate
[*] Invoke Function: sam local invoke
[*] Test Function in the Cloud: sam sync --stack-name {(stack-name)} --watch
[*] Deploy: sam deploy --guided
[cloudshell-user@ip-10-136-46-27 test1]$ sam deploy --guided
  
```



## 1.15 Confirm the deployment of the changeset by entering y

```

aws Services Search [Alt+S]
CloudShell
us-east-1 +
Disable rollback : False
Deployment s3 bucket : aws-sam-cli-managed-default-samclisourcebucket-dqs3yy7p2rhy
Capabilities : [*CAPABILITY_IAM*]
Parameter overrides : {}
Signing Profiles : {}

Initiating deployment
=====
Uploading to test1/aed6dfdc2f878e5beadc782bc2ad672.template 1245 / 1245 (100.00%)

Waiting for changeset to be created..

CloudFormation stack changeset
=====
Operation LogicalResourceId ResourceType Replacement
+ Add HelloWorldFunctionHelloWorldPermissionProd AWS::Lambda::Permission N/A
+ Add HelloWorldFunctionRole AWS::IAM::Role N/A
+ Add HelloWorldFunction AWS::Lambda::Function N/A
+ Add ServerlessRestApiDeployment47fc2d5f9d AWS::ApiGateway::Deployment N/A
+ Add ServerlessRestApiProdStage AWS::ApiGateway::Stage N/A
+ Add ServerlessRestApi AWS::ApiGateway::RestApi N/A

Changeset created successfully. arn:aws:cloudformation:us-east-1:135488290556:changeSet/samcli-deploy1723449516/e9738edf-a7e4-4940-b743-9e43d029089d

Previewing CloudFormation changeset before deployment
=====
Deploy this changeset? [y/N]: y
Feedback
  
```

## 1.16 Copy the URL from the Outputs field, specifically the Value of HelloWorldApi

```

CREATE_COMPLETE AWS::CloudFormation::Stack yes

CloudFormation outputs from deployed stack

Outputs
=====
Key HelloWorldFunctionIamRole
Description Implicit IAM Role created for Hello World function
Value arn:aws:iam::596509374816:role/yes-HelloWorldFunctionRole-E779MYQZ8WUG

Key HelloWorldApi
Description API Gateway endpoint URL for Prod stage for Hello World function
Value https://t5qzfrys36.execute-api.us-east-1.amazonaws.com/Prod/hello/

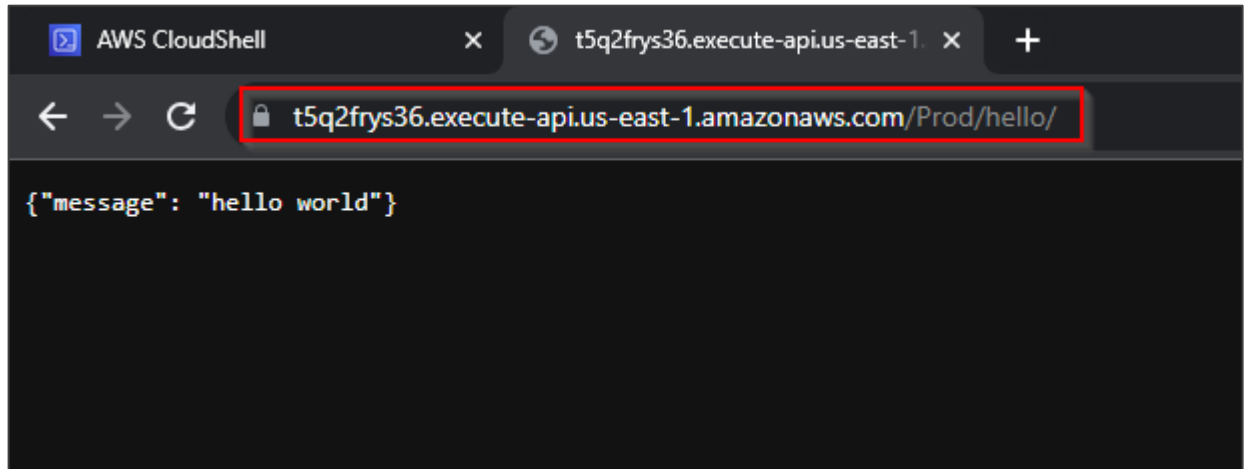
Key HelloWorldFunction
Description Hello World Lambda Function ARN
Value arn:aws:lambda:us-east-1:596509374816:function:yes-HelloWorldFunction-OuUSU7kk63IC

Successfully created/updated stack - yes in us-east-1

[cloudshell-user@ip-10-2-110-224 test1]$
  
```

1.17 Open a new browser tab and paste the URL to access the output:

**`https://t5q2frys36.execute-api.us-east-1.amazonaws.com/Prod/hello/`**



By following these steps, you have successfully created a serverless web application on AWS.