Quality Assurance via CI/CD on the Cloud

Bedir Asici

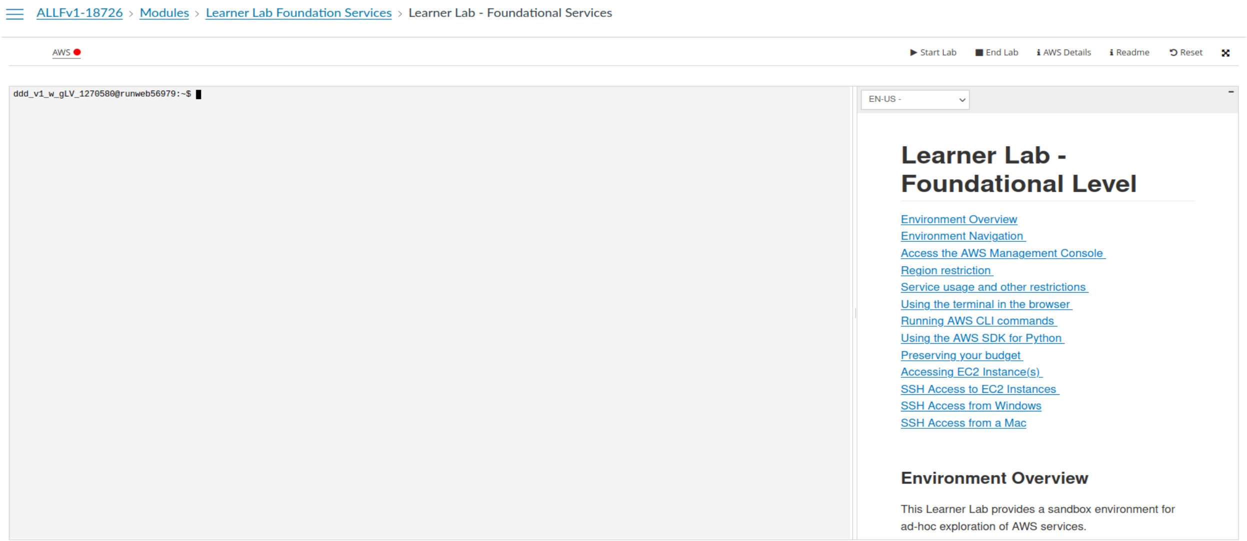
Introduction

Using the cloud 9 ide thorough aws provides a method to develop projects in a maintained and controllable environment. Resources, permissions and other limiting factors can be solved through unique ways when the user is developing in the environment with others as all the mentioned factors can be controlled centrally. Some location sensitivity has been found for example the cloud 9 ide location has to match the bucket location if the required information is in a specific bucket.

Task One : preparing IDE

Aws was logged into using the student access invitation. Once the lab was started, the aws popup link was used to access the aws console. Once open, an environment was created using the appropriate details, following this cloud 9 ide was accessed. Caller identity was shown (Figure 9)

-Figure 1- Locating the lab startup screen


ALLFv1-18726 › Modules › Learner Lab Foundation Services › Learner Lab - Foundational Services
AWS
▶ Start Lab
■ End Lab
ℹ AWS Details
ℹ Readme
Reset
EN-US -

```
ddd_v1_w_gLV_1270580@runweb56979:~$ █
```

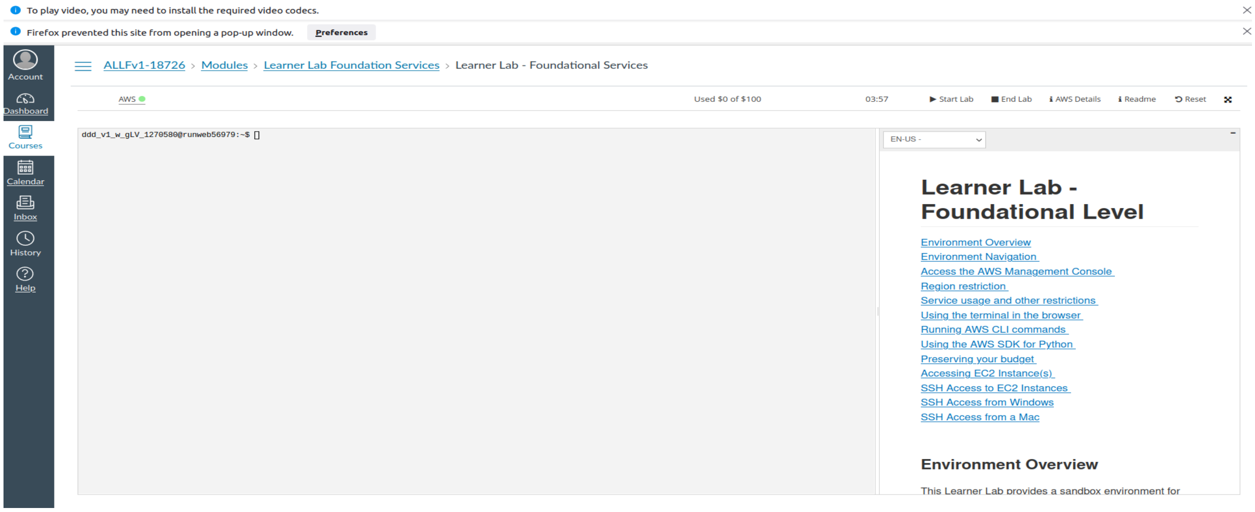
Learner Lab - Foundational Level

Environment Overview
Environment Navigation
Access the AWS Management Console
Region restriction
Service usage and other restrictions
Using the terminal in the browser
Running AWS CLI commands
Using the AWS SDK for Python
Preserving your budget
Accessing EC2 Instance(s)
SSH Access to EC2 Instances
SSH Access from Windows
SSH Access from a Mac

Environment Overview

This Learner Lab provides a sandbox environment for ad-hoc exploration of AWS services.

-Figure 2- Start lab and launch AWS by clicking <u>AWS</u>



To play video, you may need to install the required video codecs.

Firefox prevented this site from opening a pop-up window.

```
ddd_v1_w_gLV_1270580@runweb56979:~$ []
```

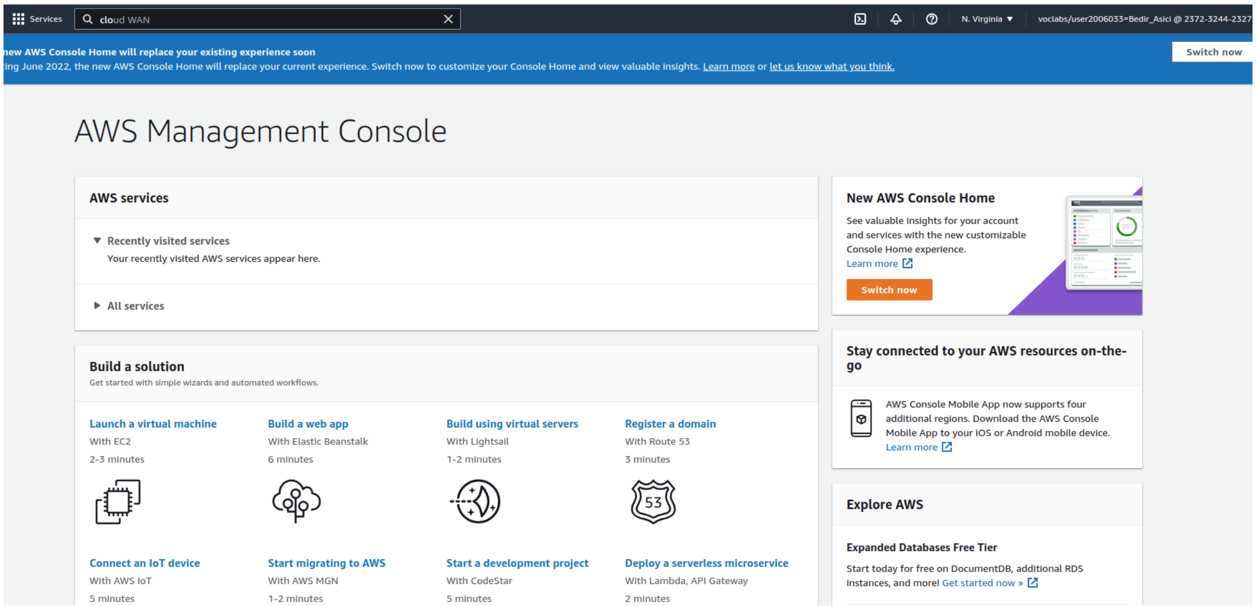
Learner Lab - Foundational Level

Environment Overview
Environment Navigation
Access the AWS Management Console
Region restriction
Service usage and other restrictions
Using the terminal in the browser
Running AWS CLI commands
Using the AWS SDK for Python
Preserving your budget
Accessing EC2 Instance(s)
SSH Access to EC2 Instances
SSH Access from Windows
SSH Access from a Mac

Environment Overview

This Learner Lab provides a sandbox environment for

-*Figure 3- Console home screen*



new AWS Console Home will replace your existing experience soon

ing June 2022, the new AWS Console Home will replace your current experience. Switch now to customize your Console Home and view valuable insights. Learn more or let us know what you think.

AWS Management Console

AWS services

▼ Recently visited services

Your recently visited AWS services appear here.

▶ All services

Build a solution

Get started with simple wizards and automated workflows.

Launch a virtual machine
With EC2
2-3 minutes

Build a web app
With Elastic Beanstalk
6 minutes

Build using virtual servers
With Lightsail
1-2 minutes

Register a domain
With Route 53
3 minutes

Connect an IoT device
With AWS IoT
5 minutes

Start migrating to AWS
With AWS MGN
1-2 minutes

Start a development project
With CodeStar
5 minutes

Deploy a serverless microservice
With Lambda, API Gateway
2 minutes

New AWS Console Home

See valuable insights for your account and services with the new customizable Console Home experience.
Learn more

Stay connected to your AWS resources on-the-go

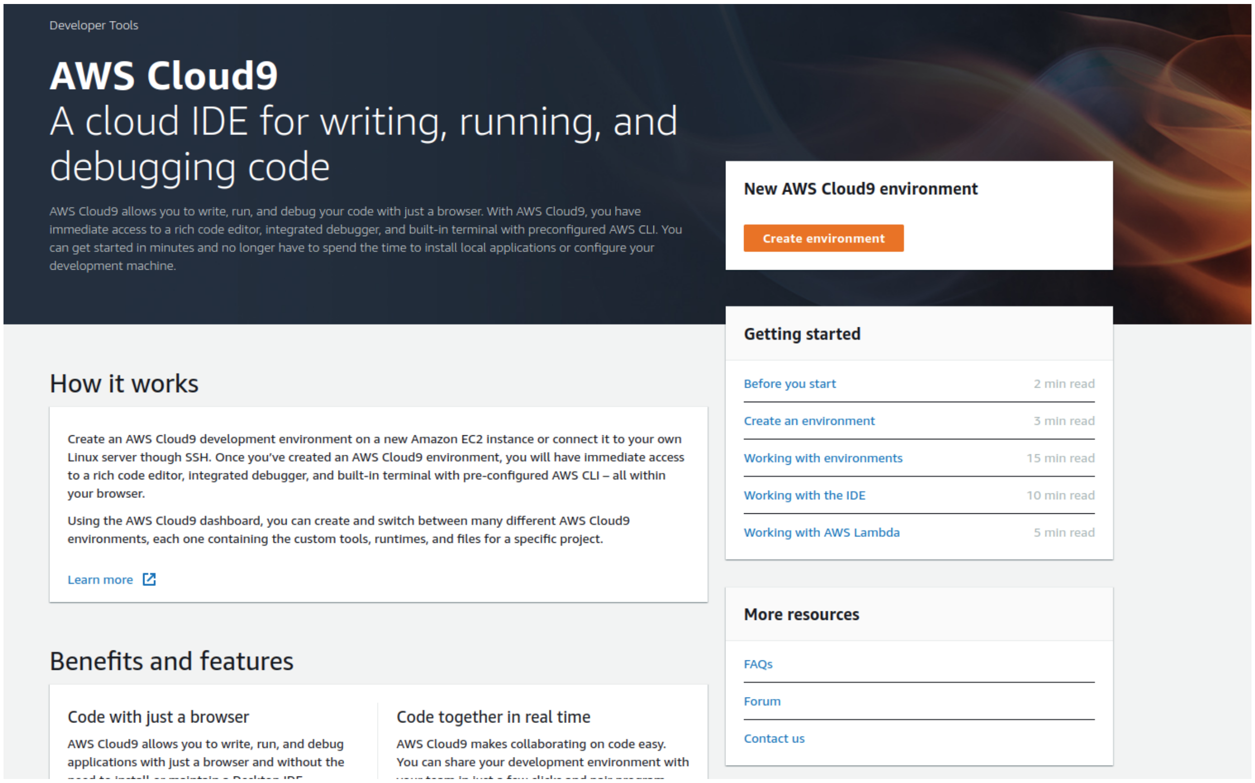
AWS Console Mobile App now supports four additional regions. Download the AWS Console Mobile App to your iOS or Android mobile device.
Learn more

Explore AWS

Expanded Databases Free Tier

Start today for free on DocumentDB, additional RDS instances, and more! Get started now »

-*Figure 4- Cloud9 IDE home screen*



Developer Tools

AWS Cloud9

A cloud IDE for writing, running, and debugging code

AWS Cloud9 allows you to write, run, and debug your code with just a browser. With AWS Cloud9, you have immediate access to a rich code editor, integrated debugger, and built-in terminal with preconfigured AWS CLI. You can get started in minutes and no longer have to spend the time to install local applications or configure your development machine.

New AWS Cloud9 environment

Getting started

Before you start	2 min read
Create an environment	3 min read
Working with environments	15 min read
Working with the IDE	10 min read
Working with AWS Lambda	5 min read

How it works

Create an AWS Cloud9 development environment on a new Amazon EC2 instance or connect it to your own Linux server though SSH. Once you've created an AWS Cloud9 environment, you will have immediate access to a rich code editor, integrated debugger, and built-in terminal with pre-configured AWS CLI – all within your browser.

Using the AWS Cloud9 dashboard, you can create and switch between many different AWS Cloud9 environments, each one containing the custom tools, runtimes, and files for a specific project.

Learn more

More resources

FAQs

Forum

Contact us

Benefits and features

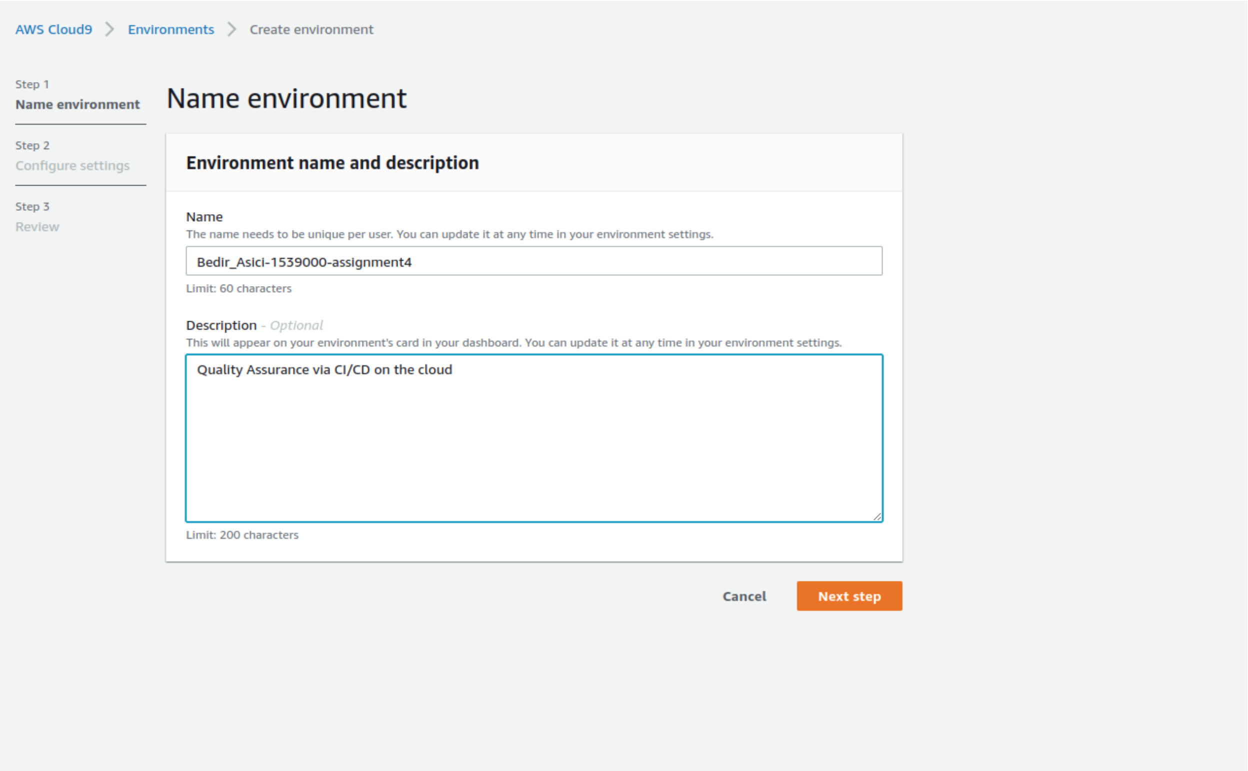
Code with just a browser

AWS Cloud9 allows you to write, run, and debug applications with just a browser and without the

Code together in real time

AWS Cloud9 makes collaborating on code easy. You can share your development environment with

-Figure 5- Creating environment



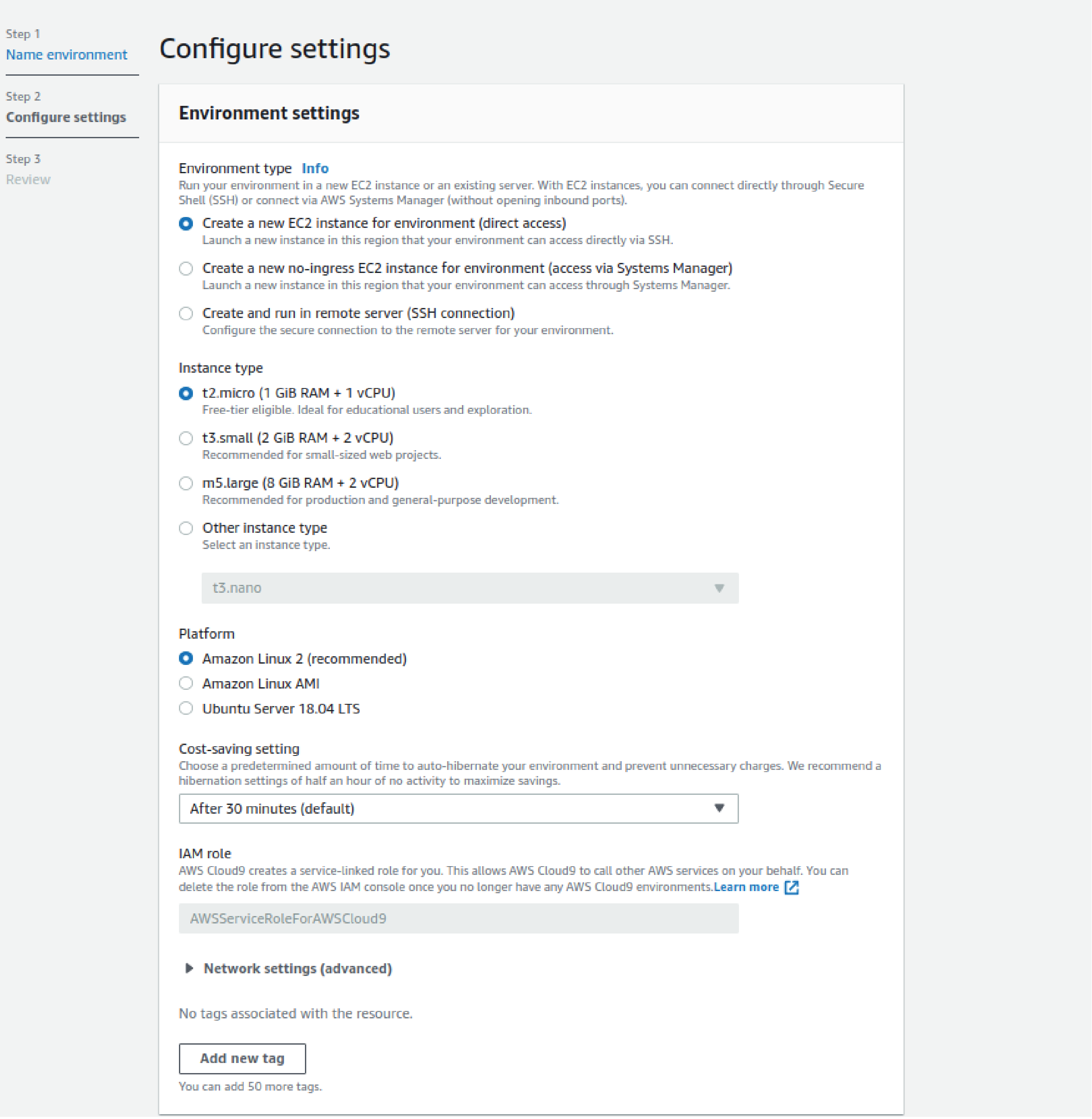
Name environment

Environment name and description

Name
The name needs to be unique per user. You can update it at any time in your environment settings.
Bedir_Asici-1539000-assignment4
Limit: 60 characters

Description - *Optional*
This will appear on your environment's card in your dashboard. You can update it at any time in your environment settings.
Quality Assurance via CI/CD on the cloud
Limit: 200 characters

-Figure 6- Environment settings



Step 1
Name environment

Step 2
Configure settings

Step 3
Review

Configure settings

Environment settings

Environment type Info
Run your environment in a new EC2 instance or an existing server. With EC2 instances, you can connect directly through Secure Shell (SSH) or connect via AWS Systems Manager (without opening inbound ports).

● Create a new EC2 instance for environment (direct access)
Launch a new instance in this region that your environment can access directly via SSH.

○ Create a new no-ingress EC2 instance for environment (access via Systems Manager)
Launch a new instance in this region that your environment can access through Systems Manager.

○ Create and run in remote server (SSH connection)
Configure the secure connection to the remote server for your environment.

Instance type

● t2.micro (1 GiB RAM + 1 vCPU)
Free-tier eligible. Ideal for educational users and exploration.

○ t3.small (2 GiB RAM + 2 vCPU)
Recommended for small-sized web projects.

○ m5.large (8 GiB RAM + 2 vCPU)
Recommended for production and general-purpose development.

○ Other instance type
Select an instance type.

t3.nano ▼

Platform

● Amazon Linux 2 (recommended)

○ Amazon Linux AMI

○ Ubuntu Server 18.04 LTS

Cost-saving setting
Choose a predetermined amount of time to auto-hibernate your environment and prevent unnecessary charges. We recommend a hibernation settings of half an hour of no activity to maximize savings.

After 30 minutes (default) ▼

IAM role
AWS Cloud9 creates a service-linked role for you. This allows AWS Cloud9 to call other AWS services on your behalf. You can delete the role from the AWS IAM console once you no longer have any AWS Cloud9 environments.Learn more

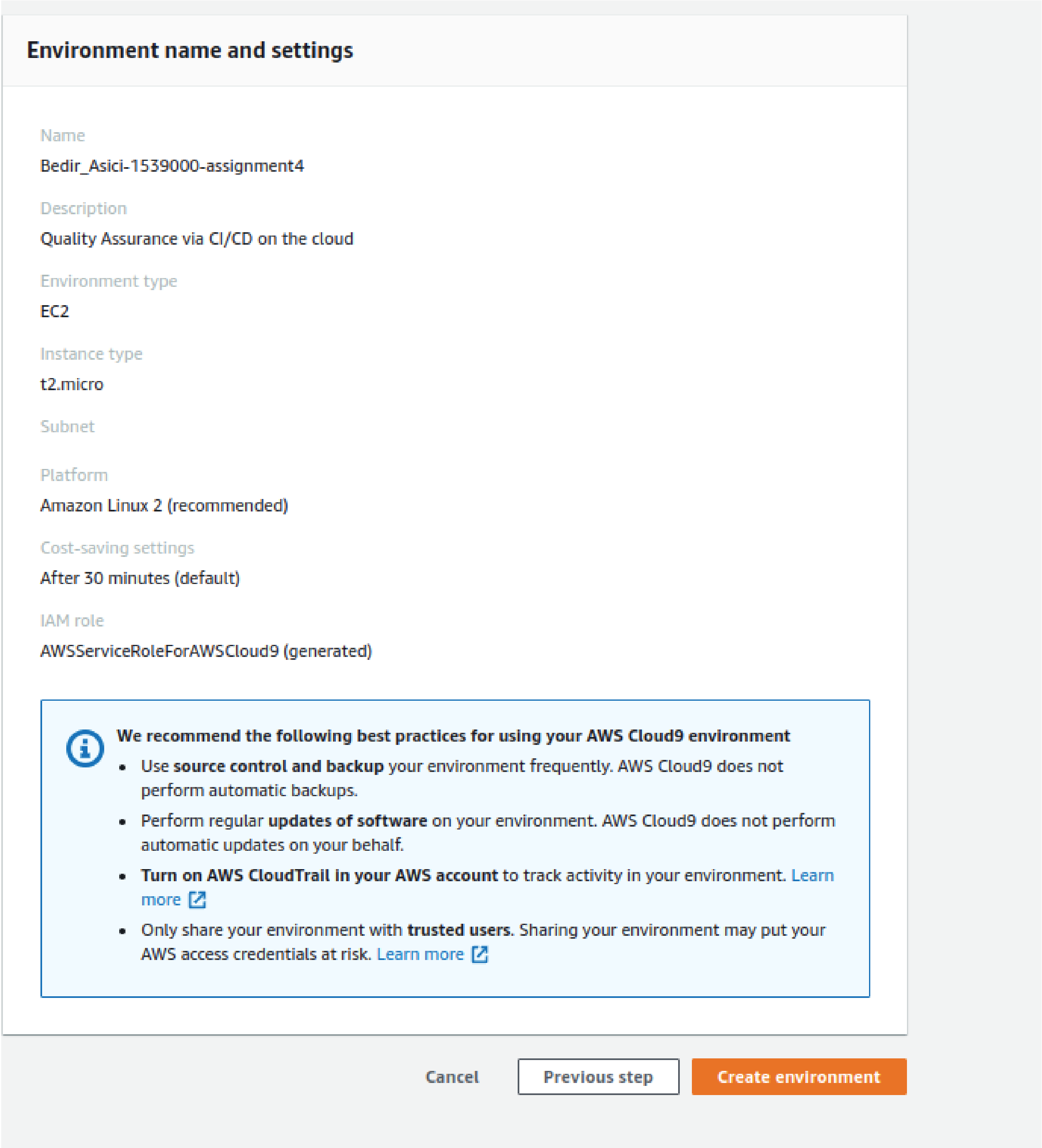
AWSServiceRoleForAWSCloud9

▶ Network settings (advanced)

No tags associated with the resource.

You can add 50 more tags.

-Figure 7- Showing result of settings



Environment name and settings

Name

Bedir_Asici-1539000-assignment4

Description

Quality Assurance via CI/CD on the cloud

Environment type

EC2

Instance type

t2.micro

Subnet

Platform

Amazon Linux 2 (recommended)

Cost-saving settings

After 30 minutes (default)

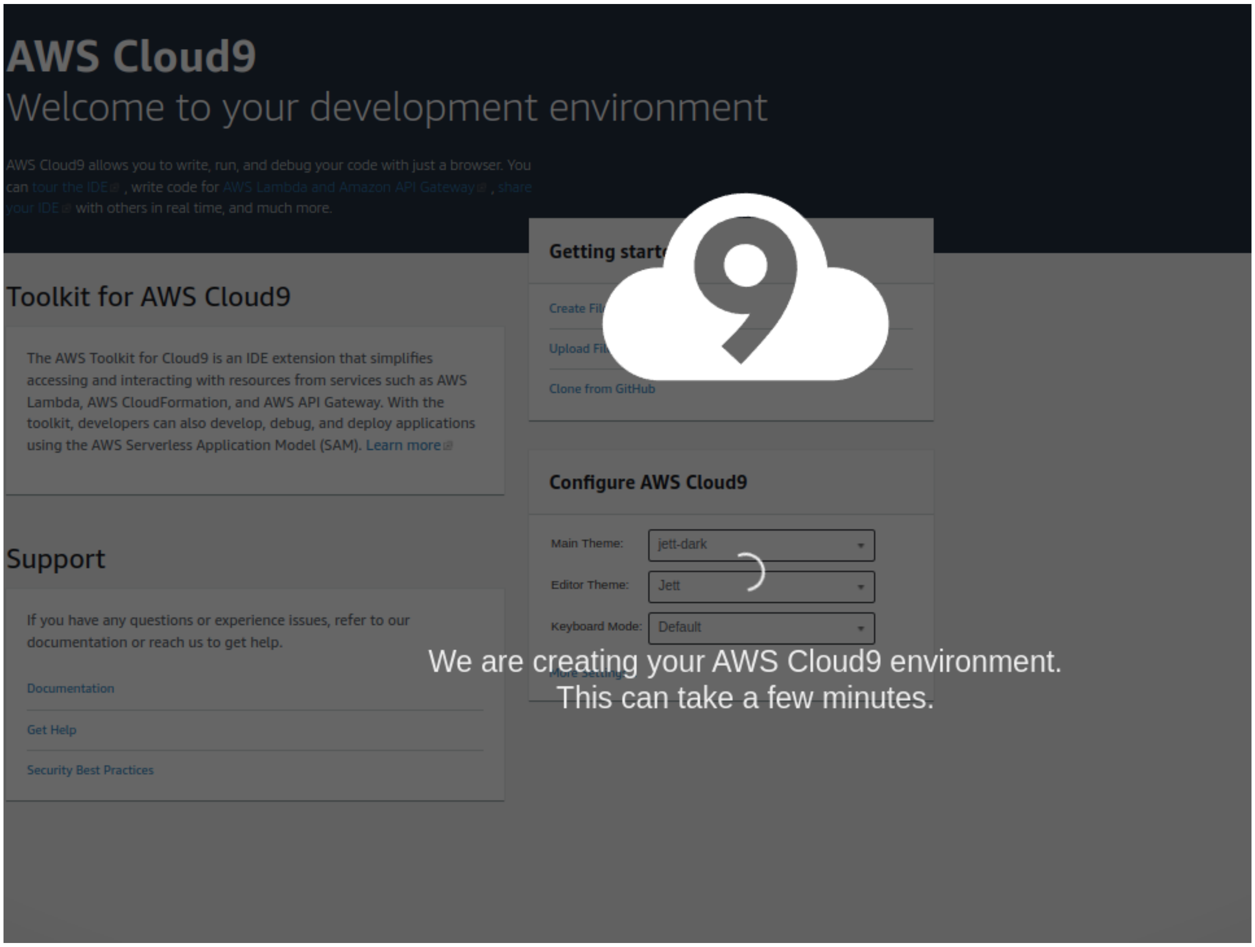
IAM role

AWSServiceRoleForAWSCloud9 (generated)

We recommend the following best practices for using your AWS Cloud9 environment

- Use **source control and backup** your environment frequently. AWS Cloud9 does not perform automatic backups.
- Perform regular **updates of software** on your environment. AWS Cloud9 does not perform automatic updates on your behalf.
- **Turn on AWS CloudTrail in your AWS account** to track activity in your environment. Learn more
- Only share your environment with **trusted users**. Sharing your environment may put your AWS access credentials at risk. Learn more

-Figure 8- Launching Cloud9 IDE



AWS Cloud9

Welcome to your development environment

AWS Cloud9 allows you to write, run, and debug your code with just a browser. You can tour the IDE, write code for AWS Lambda and Amazon API Gateway, share your IDE with others in real time, and much more.

Toolkit for AWS Cloud9

The AWS Toolkit for Cloud9 is an IDE extension that simplifies accessing and interacting with resources from services such as AWS Lambda, AWS CloudFormation, and AWS API Gateway. With the toolkit, developers can also develop, debug, and deploy applications using the AWS Serverless Application Model (SAM). Learn more

Support

If you have any questions or experience issues, refer to our documentation or reach us to get help.

Configure AWS Cloud9

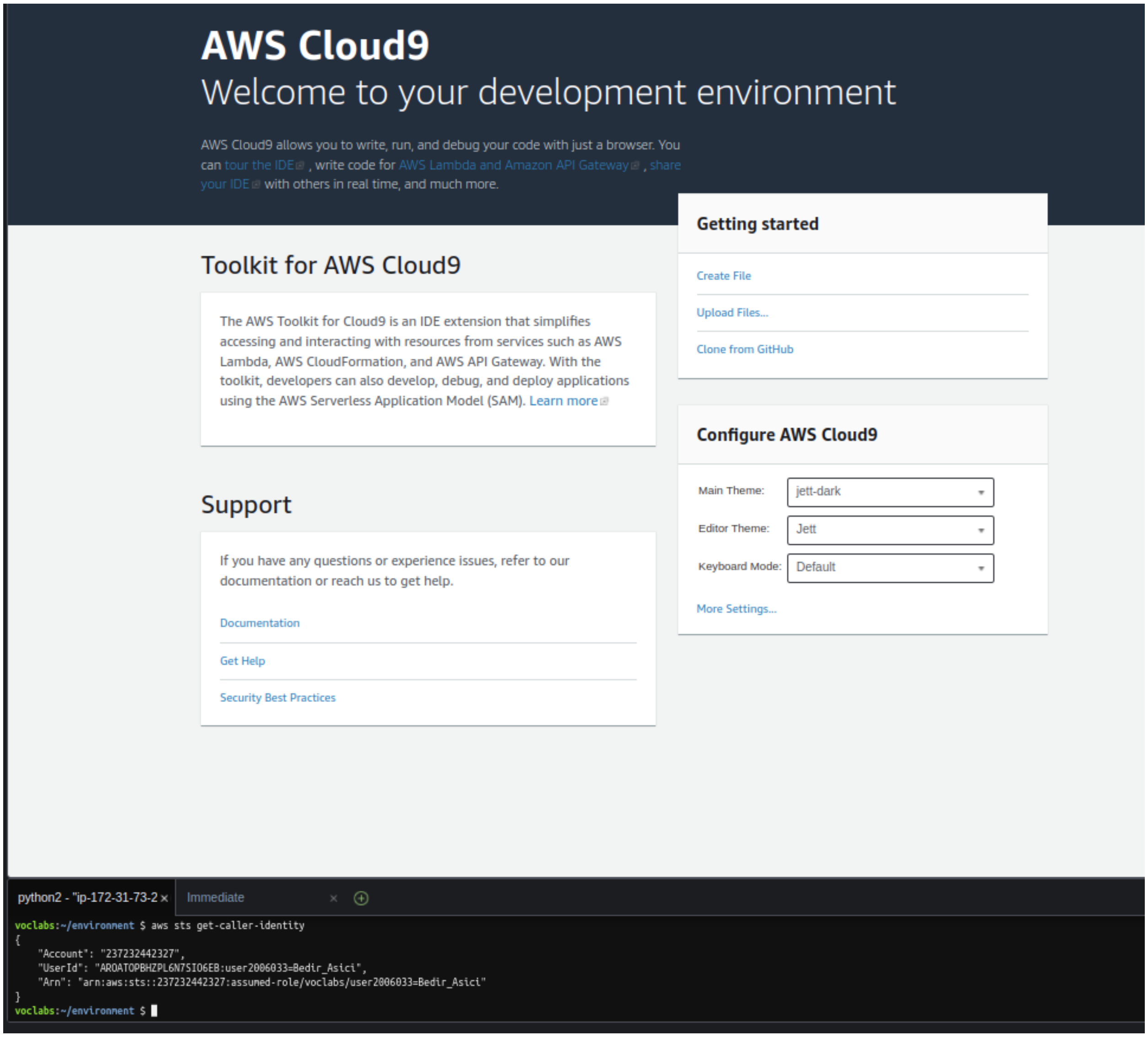
Main Theme: jett-dark

Editor Theme: Jett

Keyboard Mode: Default

We are creating your AWS Cloud9 environment.
This can take a few minutes.

*-Figure 9- Running aws sts -get-caller-identity from ide termina*l



AWS Cloud9

Welcome to your development environment

AWS Cloud9 allows you to write, run, and debug your code with just a browser. You can tour the IDE, write code for AWS Lambda and Amazon API Gateway, share your IDE with others in real time, and much more.

Toolkit for AWS Cloud9

The AWS Toolkit for Cloud9 is an IDE extension that simplifies accessing and interacting with resources from services such as AWS Lambda, AWS CloudFormation, and AWS API Gateway. With the toolkit, developers can also develop, debug, and deploy applications using the AWS Serverless Application Model (SAM). Learn more

Support

If you have any questions or experience issues, refer to our documentation or reach us to get help.

Documentation

Get Help

Security Best Practices

Getting started

Create File

Upload Files...

Clone from GitHub

Configure AWS Cloud9

Main Theme: jett-dark

Editor Theme: Jett

Keyboard Mode: Default

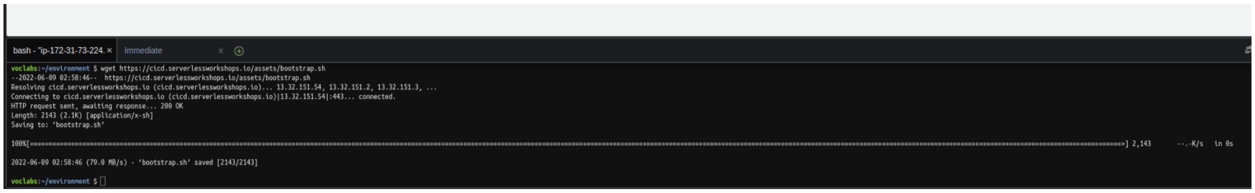
More Settings...

```
voclabs:~/environment $ aws sts get-caller-identity
{
    "Account": "237232442327",
    "UserId": "AROATOPBHZPL6N7SIO6EB:user2006033=Bedir_Asici",
    "Arn": "arn:aws:sts::237232442327:assumed-role/voclabs/user2006033=Bedir_Asici"
}
voclabs:~/environment $ 
```

Task Two: Initializing AWS Serverless Architecture Model

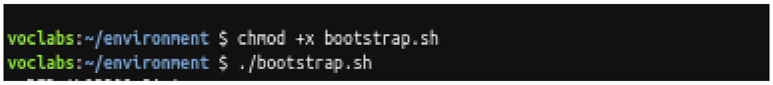
The instructions were followed to set up a Helloworld template application.

-Figure 10- Using wget https://cicd.serverlessworkshops.io/assets/bootstrap.sh



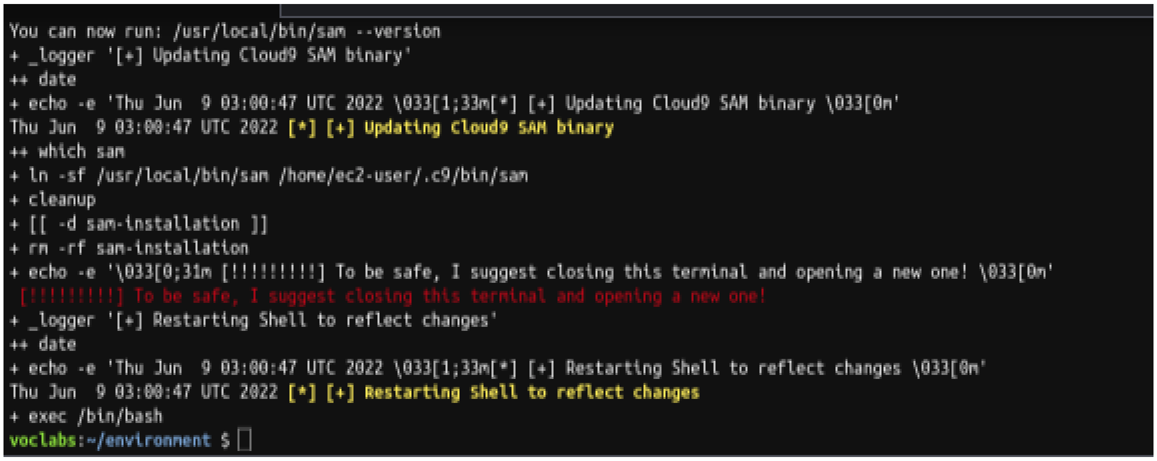
```
voclabs:~/environment $ wget https://cicd.serverlessworkshops.io/assets/bootstrap.sh
--2022-06-09 02:58:46--  https://cicd.serverlessworkshops.io/assets/bootstrap.sh
Resolving cicd.serverlessworkshops.io (cicd.serverlessworkshops.io)... 13.32.151.54, 13.32.151.2, 13.32.151.3, ...
Connecting to cicd.serverlessworkshops.io (cicd.serverlessworkshops.io)|13.32.151.54|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2143 (2.1K) [application/x-sh]
Saving to: ‘bootstrap.sh’

100%[==================================================================================================================================================================================================================>] 2,143      --.-K/s   in 0s

2022-06-09 02:58:46 (79.0 MB/s) - ‘bootstrap.sh’ saved [2143/2143]

voclabs:~/environment $ 
```

-Figure 11- Using chmod +x bootstrap.sh



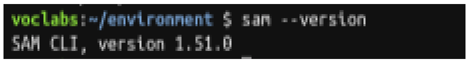
```
voclabs:~/environment $ chmod +x bootstrap.sh
voclabs:~/environment $ ./bootstrap.sh
```

-Figure 12- Results



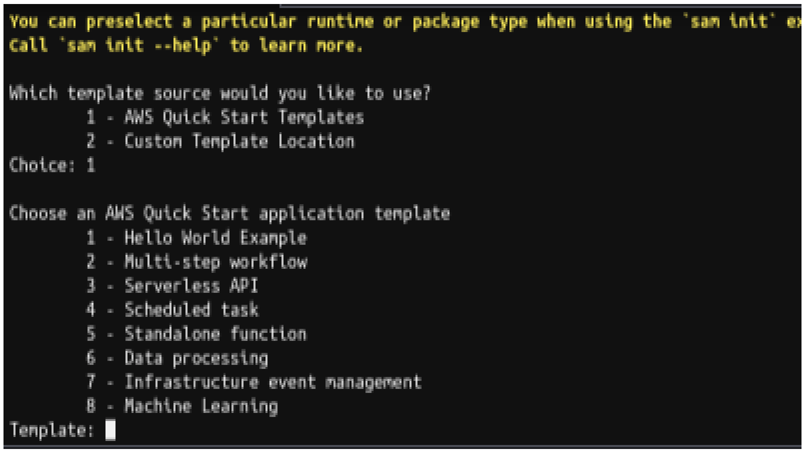
```
You can now run: /usr/local/bin/sam --version
+ _logger '[+] Updating Cloud9 SAM binary'
++ date
+ echo -e 'Thu Jun  9 03:00:47 UTC 2022 \033[1;33m[*] [+] Updating Cloud9 SAM binary \033[0m'
Thu Jun  9 03:00:47 UTC 2022 [*] [+] Updating Cloud9 SAM binary
++ which sam
+ ln -sf /usr/local/bin/sam /home/ec2-user/.c9/bin/sam
+ cleanup
+ [[ -d sam-installation ]]
+ rm -rf sam-installation
+ echo -e '\033[0;31m [!!!!!!!!!] To be safe, I suggest closing this terminal and opening a new one! \033[0m'
 [!!!!!!!!!] To be safe, I suggest closing this terminal and opening a new one!
+ _logger '[+] Restarting Shell to reflect changes'
++ date
+ echo -e 'Thu Jun  9 03:00:47 UTC 2022 \033[1;33m[*] [+] Restarting Shell to reflect changes \033[0m'
Thu Jun  9 03:00:47 UTC 2022 [*] [+] Restarting Shell to reflect changes
+ exec /bin/bash
voclabs:~/environment $ 
```

-Figure 13- Getting the version



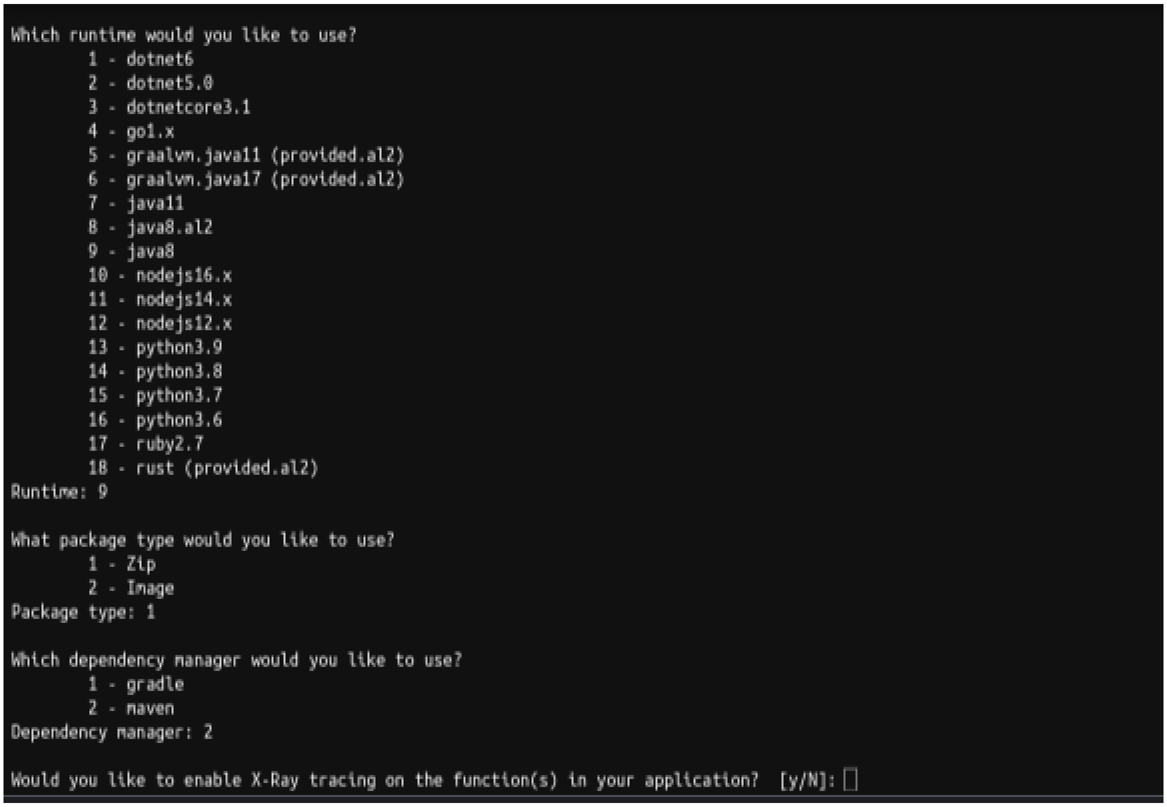
```
voclabs:~/environment $ sam --version
SAM CLI, version 1.51.0
```

-Figure 14- Using sam init



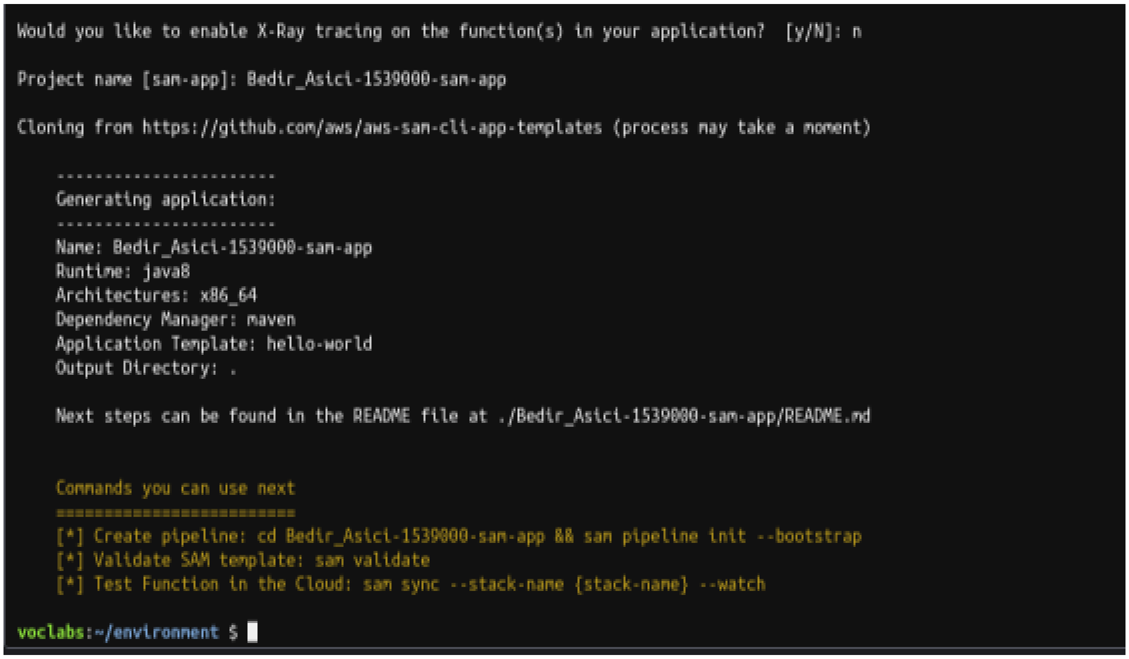
```
You can preselect a particular runtime or package type when using the `sam init` ex
Call `sam init --help` to learn more.

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 - Multi-step workflow
        3 - Serverless API
        4 - Scheduled task
        5 - Standalone function
        6 - Data processing
        7 - Infrastructure event management
        8 - Machine Learning
Template: 
```

-Figure 15- Results, picking appropriate options



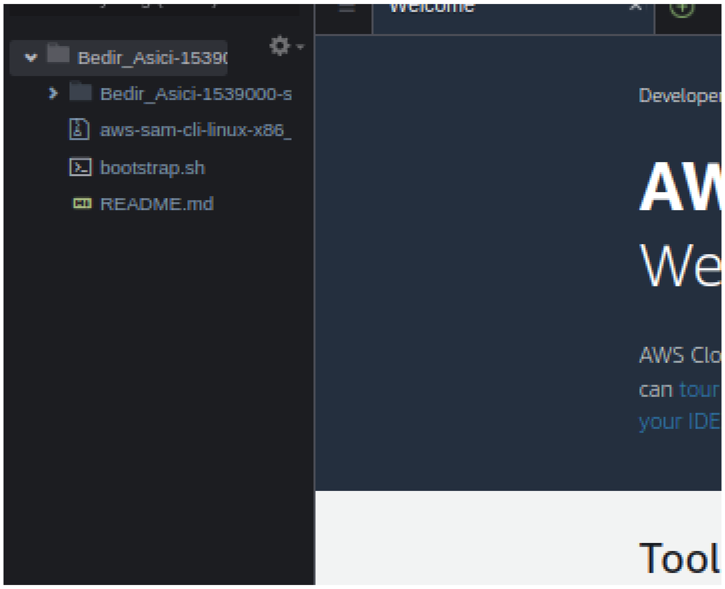
```
Which runtime would you like to use?
        1 - dotnet6
        2 - dotnet5.0
        3 - dotnetcore3.1
        4 - go1.x
        5 - graalvm.java11 (provided.al2)
        6 - graalvm.java17 (provided.al2)
        7 - java11
        8 - java8.al2
        9 - java8
        10 - nodejs16.x
        11 - nodejs14.x
        12 - nodejs12.x
        13 - python3.9
        14 - python3.8
        15 - python3.7
        16 - python3.6
        17 - ruby2.7
        18 - rust (provided.al2)
Runtime: 9

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

Which dependency manager would you like to use?
        1 - gradle
        2 - maven
Dependency manager: 2

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

-Figure 16- Results continuation



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

Project name [sam-app]: Bedir_Asici-1539000-sam-app

Cloning from https://github.com/aws/aws-sam-cli-app-templates (process may take a moment)

    -----------------------
    Generating application:
    -----------------------
    Name: Bedir_Asici-1539000-sam-app
    Runtime: java8
    Architectures: x86_64
    Dependency Manager: maven
    Application Template: hello-world
    Output Directory: .

    Next steps can be found in the README file at ./Bedir_Asici-1539000-sam-app/README.md


    Commands you can use next
    =========================
    [*] Create pipeline: cd Bedir_Asici-1539000-sam-app && sam pipeline init --bootstrap
    [*] Validate SAM template: sam validate
    [*] Test Function in the Cloud: sam sync --stack-name {stack-name} --watch

voclabs:~/environment $
```

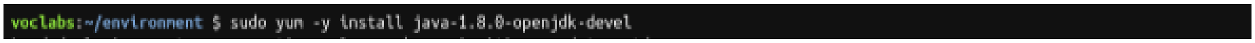
-Figure 17- Showing directory structure



Task Three: Deploy Locally

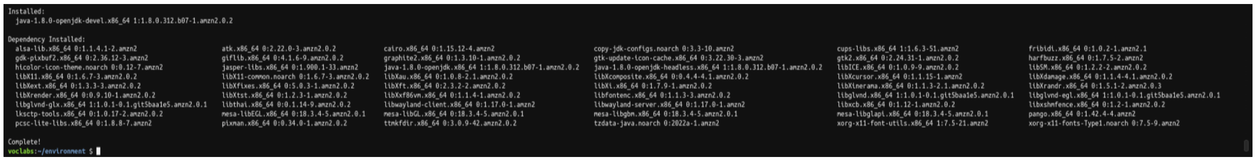
The required java SDK as well as a maven were installed, after which the application was deployed locally.

-Figure 18- Using sudo yum to install open jdk



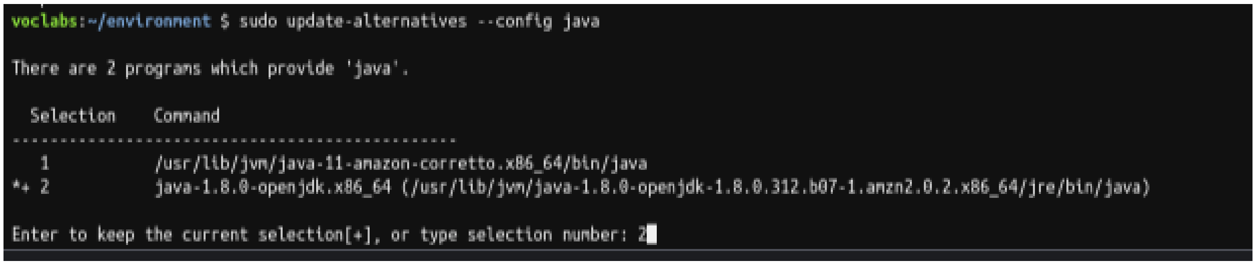
```
voclabs:~/environment $ sudo yum -y install java-1.8.0-openjdk-devel
```

-Figure 19- Result



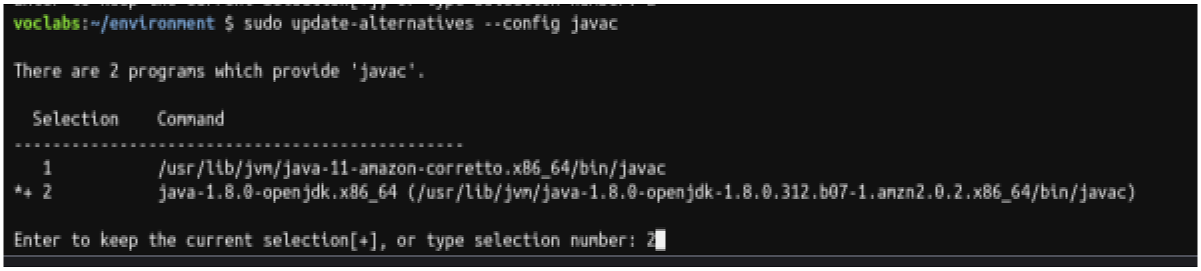
```
Installed:
  java-1.8.0-openjdk-devel.x86_64 1:1.8.0.312.b07-1.amzn2.0.2

Dependency Installed:
  alsa-lib.x86_64 0:1.1.4.1-2.amzn2                          atk.x86_64 0:2.22.0-3.amzn2.0.2                    cairo.x86_64 0:1.15.12-4.amzn2                                copy-jdk-configs.noarch 0:3.3-10.amzn2                                cups-libs.x86_64 1:1.6.3-51.amzn2                           fribidi.x86_64 0:1.0.2-1.amzn2.1
  gdk-pixbuf2.x86_64 0:2.36.12-3.amzn2                       giflib.x86_64 0:4.1.6-9.amzn2.0.2                  graphite2.x86_64 0:1.3.10-1.amzn2.0.2                         gtk-update-icon-cache.x86_64 0:3.22.30-3.amzn2                        gtk2.x86_64 0:2.24.31-1.amzn2.0.2                           harfbuzz.x86_64 0:1.7.5-2.amzn2
  hicolor-icon-theme.noarch 0:0.12-7.amzn2                   jasper-libs.x86_64 0:1.900.1-33.amzn2              java-1.8.0-openjdk.x86_64 1:1.8.0.312.b07-1.amzn2.0.2        java-1.8.0-openjdk-headless.x86_64 1:1.8.0.312.b07-1.amzn2.0.2        libICE.x86_64 0:1.0.9-9.amzn2.0.2                           libSM.x86_64 0:1.2.2-2.amzn2.0.2
  libX11.x86_64 0:1.6.7-3.amzn2.0.2                          libX11-common.noarch 0:1.6.7-3.amzn2.0.2           libXau.x86_64 0:1.0.8-2.1.amzn2.0.2                           libXcomposite.x86_64 0:0.4.4-4.1.amzn2.0.2                            libXcursor.x86_64 0:1.1.15-1.amzn2                          libXdamage.x86_64 0:1.1.4-4.1.amzn2.0.2
  libXext.x86_64 0:1.3.3-3.amzn2.0.2                         libXfixes.x86_64 0:5.0.3-1.amzn2.0.2               libXft.x86_64 0:2.3.2-2.amzn2.0.2                             libXi.x86_64 0:1.7.9-1.amzn2.0.2                                      libXinerama.x86_64 0:1.1.3-2.1.amzn2.0.2                    libXrandr.x86_64 0:1.5.1-2.amzn2.0.3
  libXrender.x86_64 0:0.9.10-1.amzn2.0.2                     libXtst.x86_64 0:1.2.3-1.amzn2.0.2                 libXxf86vm.x86_64 0:1.1.4-1.amzn2.0.2                         libfontenc.x86_64 0:1.1.3-3.amzn2.0.2                                 libglvnd.x86_64 1:1.0.1-0.1.git5baa1e5.amzn2.0.1            libglvnd-egl.x86_64 1:1.0.1-0.1.git5baa1e5.amzn2.0.1
  libglvnd-glx.x86_64 1:1.0.1-0.1.git5baa1e5.amzn2.0.1       libthai.x86_64 0:0.1.14-9.amzn2.0.2                libwayland-client.x86_64 0:1.17.0-1.amzn2                     libwayland-server.x86_64 0:1.17.0-1.amzn2                             libxcb.x86_64 0:1.12-1.amzn2.0.2                            libxshmfence.x86_64 0:1.2-1.amzn2.0.2
  lksctp-tools.x86_64 0:1.0.17-2.amzn2.0.2                   mesa-libEGL.x86_64 0:18.3.4-5.amzn2.0.1            mesa-libGL.x86_64 0:18.3.4-5.amzn2.0.1                        mesa-libgbm.x86_64 0:18.3.4-5.amzn2.0.1                               mesa-libglapi.x86_64 0:18.3.4-5.amzn2.0.1                   pango.x86_64 0:1.42.4-4.amzn2
  pcsc-lite-libs.x86_64 0:1.8.8-7.amzn2                      pixman.x86_64 0:0.34.0-1.amzn2.0.2                 ttmkfdir.x86_64 0:3.0.9-42.amzn2.0.2                          tzdata-java.noarch 0:2022a-1.amzn2                                    xorg-x11-font-utils.x86_64 1:7.5-21.amzn2                   xorg-x11-fonts-Type1.noarch 0:7.5-9.amzn2

Complete!
voclabs:~/environment $ 
```

-Figure 20- Seeing if the java is up to date



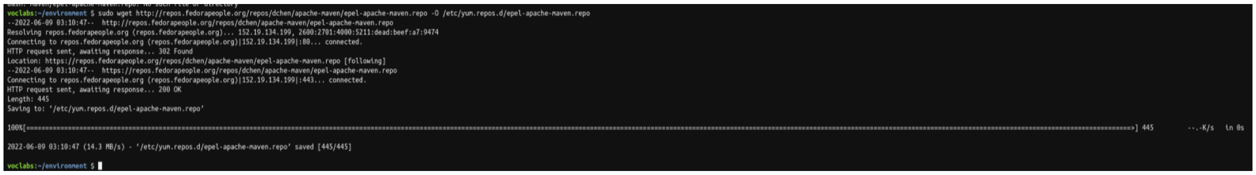
```
voclabs:~/environment $ sudo update-alternatives --config java

There are 2 programs which provide 'java'.

  Selection    Command
-----------------------------------------------
   1           /usr/lib/jvm/java-11-amazon-corretto.x86_64/bin/java
*+ 2           java-1.8.0-openjdk.x86_64 (/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.312.b07-1.amzn2.0.2.x86_64/jre/bin/java)

Enter to keep the current selection[+], or type selection number: 2
```

-Figure 21- Result



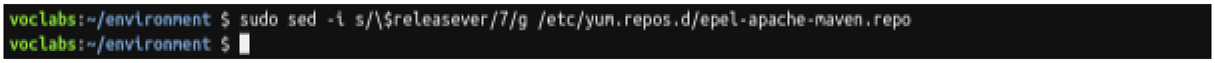
```
Enter to keep the current selection[+], or type selection number: 2
voclabs:~/environment $ sudo update-alternatives --config javac

There are 2 programs which provide 'javac'.

  Selection    Command
-----------------------------------------------
   1           /usr/lib/jvm/java-11-amazon-corretto.x86_64/bin/javac
*+ 2           java-1.8.0-openjdk.x86_64 (/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.312.b07-1.amzn2.0.2.x86_64/bin/javac)

Enter to keep the current selection[+], or type selection number: 2
```

-Figure 22-Result continued



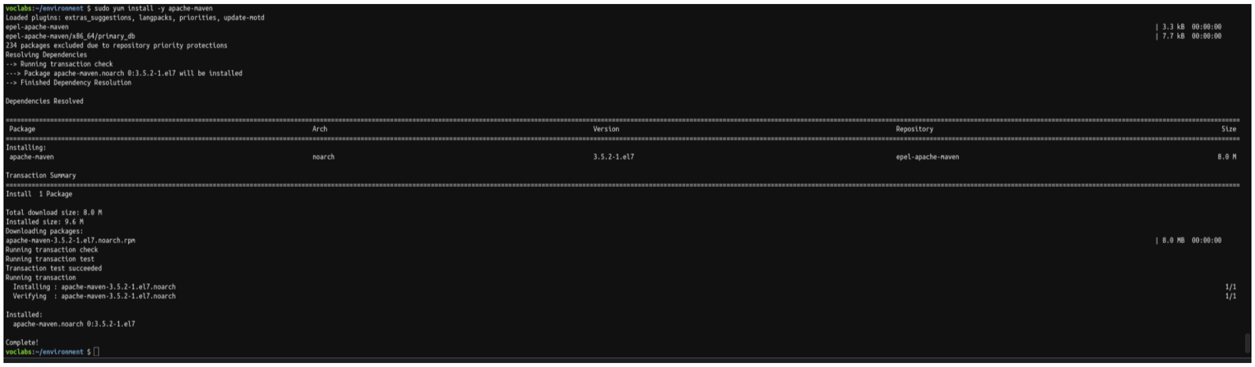
```
bash: maven/epel-apache-maven.repo: No such file or directory
voclabs:~/environment $ sudo wget http://repos.fedorapeople.org/repos/dchen/apache-maven/epel-apache-maven.repo -O /etc/yum.repos.d/epel-apache-maven.repo
--2022-06-09 03:10:47--  http://repos.fedorapeople.org/repos/dchen/apache-maven/epel-apache-maven.repo
Resolving repos.fedorapeople.org (repos.fedorapeople.org)... 152.19.134.199, 2600:2701:4000:5211:dead:beef:a7:9474
Connecting to repos.fedorapeople.org (repos.fedorapeople.org)|152.19.134.199|:80... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://repos.fedorapeople.org/repos/dchen/apache-maven/epel-apache-maven.repo [following]
--2022-06-09 03:10:47--  https://repos.fedorapeople.org/repos/dchen/apache-maven/epel-apache-maven.repo
Connecting to repos.fedorapeople.org (repos.fedorapeople.org)|152.19.134.199|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 445
Saving to: '/etc/yum.repos.d/epel-apache-maven.repo'

100%[==================================================================================================>] 445       --.-K/s   in 0s

2022-06-09 03:10:47 (14.3 MB/s) - '/etc/yum.repos.d/epel-apache-maven.repo' saved [445/445]

voclabs:~/environment $
```

-Figure 23- Setting up maven pt.1



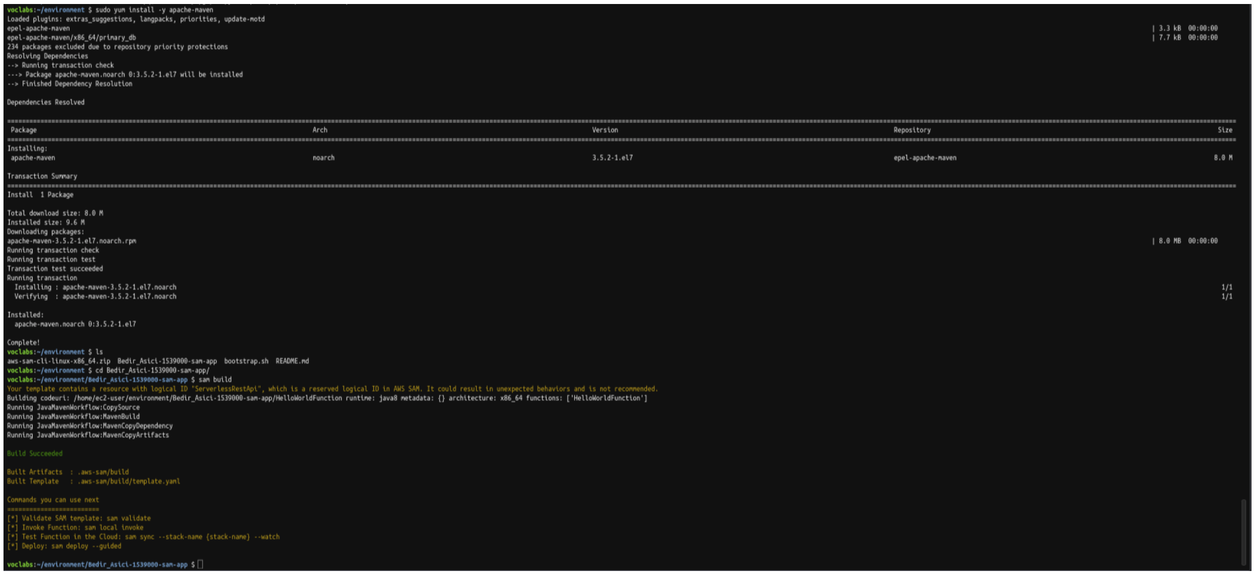
```
voclabs:~/environment $ sudo sed -i s/\$releasever/7/g /etc/yum.repos.d/epel-apache-maven.repo
voclabs:~/environment $
```

-Figure 24-*Setting up maven pt.2*



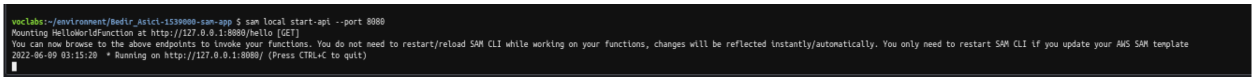
```
voclabs:~/environment $ sudo yum install -y apache-maven
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
epel-apache-maven                                                                                                    | 3.3 kB  00:00:00
epel-apache-maven/x86_64/primary_db                                                                                  | 7.7 kB  00:00:00
234 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
---> Package apache-maven.noarch 0:3.5.2-1.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

==========================================================================================================================================
 Package                          Arch                          Version                          Repository                          Size
==========================================================================================================================================
Installing:
 apache-maven                     noarch                        3.5.2-1.el7                      epel-apache-maven                  8.0 M

Transaction Summary
==========================================================================================================================================
Install  1 Package

Total download size: 8.0 M
Installed size: 9.6 M
Downloading packages:
apache-maven-3.5.2-1.el7.noarch.rpm                                                                                  | 8.0 MB  00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : apache-maven-3.5.2-1.el7.noarch                                                                                          1/1
  Verifying  : apache-maven-3.5.2-1.el7.noarch                                                                                          1/1

Installed:
  apache-maven.noarch 0:3.5.2-1.el7

Complete!
voclabs:~/environment $
```

-Figure 25-*Using sam build*



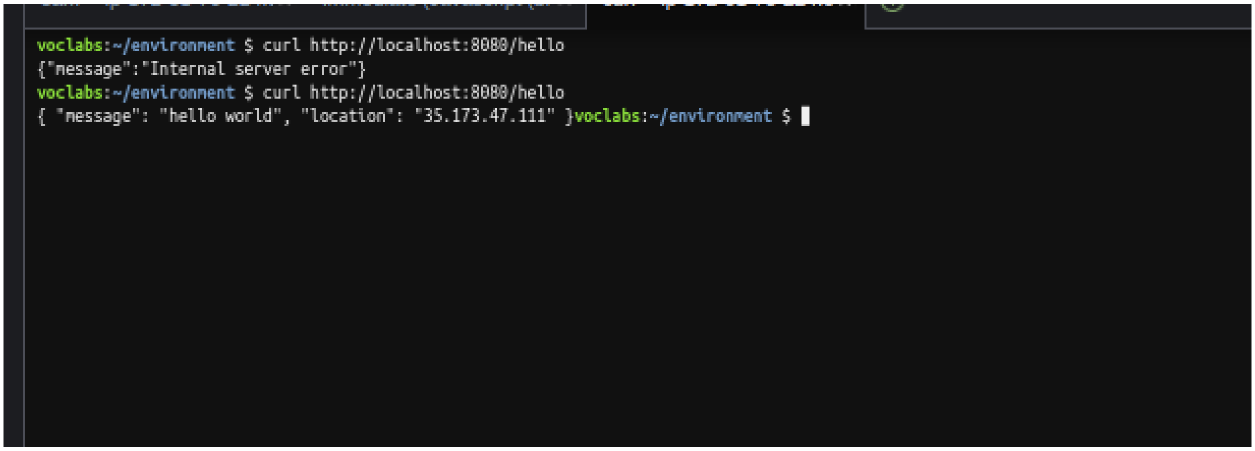
```
voclabs:~/environment $ sudo yum install -y apache-maven
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
epel-apache-maven                                                                                                    | 3.3 kB  00:00:00
epel-apache-maven/x86_64/primary_db                                                                                  | 7.7 kB  00:00:00
234 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
---> Package apache-maven.noarch 0:3.5.2-1.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

==========================================================================================================================================
 Package                          Arch                          Version                          Repository                          Size
==========================================================================================================================================
Installing:
 apache-maven                     noarch                        3.5.2-1.el7                      epel-apache-maven                  8.0 M

Transaction Summary
==========================================================================================================================================
Install  1 Package

Total download size: 8.0 M
Installed size: 9.6 M
Downloading packages:
apache-maven-3.5.2-1.el7.noarch.rpm                                                                                  | 8.0 MB  00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : apache-maven-3.5.2-1.el7.noarch                                                                                          1/1
  Verifying  : apache-maven-3.5.2-1.el7.noarch                                                                                          1/1

Installed:
  apache-maven.noarch 0:3.5.2-1.el7

Complete!
voclabs:~/environment $ ls
aws-sam-cli-linux-x86_64.zip  Bedir_Asici-1539000-sam-app  bootstrap.sh  README.md
voclabs:~/environment $ cd Bedir_Asici-1539000-sam-app/
voclabs:~/environment/Bedir_Asici-1539000-sam-app $ sam build
Your template contains a resource with logical ID "ServerlessRestApi", which is a reserved logical ID in AWS SAM. It could result in unexpected behaviors and is not recommended.
Building codeuri: /home/ec2-user/environment/Bedir_Asici-1539000-sam-app/HelloWorldFunction runtime: java8 metadata: {} architecture: x86_64 functions: ['HelloWorldFunction']
Running JavaMavenWorkflow:CopySource
Running JavaMavenWorkflow:MavenBuild
Running JavaMavenWorkflow:MavenCopyDependency
Running JavaMavenWorkflow:MavenCopyArtifacts

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

voclabs:~/environment/Bedir_Asici-1539000-sam-app $
```

-Figure 26-Launching application locally



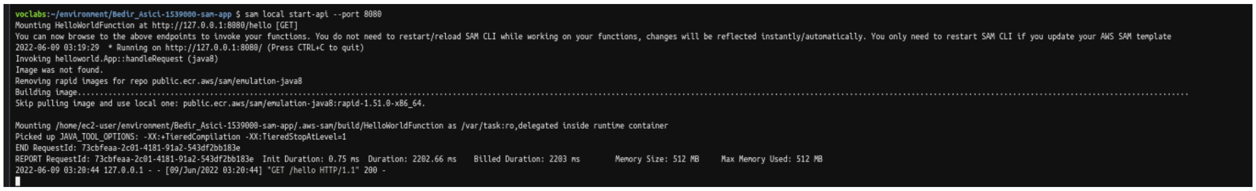
```
voclabs:~/environment/Bedir_Asici-1539000-sam-app $ sam local start-api --port 8080
Mounting HelloWorldFunction at http://127.0.0.1:8080/hello [GET]
You can now browse to the above endpoints to invoke your functions. You do not need to restart/reload SAM CLI while working on your functions, changes will be reflected instantly/automatically. You only need to restart SAM CLI if you update your AWS SAM template
2022-06-09 03:15:20  * Running on http://127.0.0.1:8080/ (Press CTRL+C to quit)
```

-Figure 27-Curling to query launched application



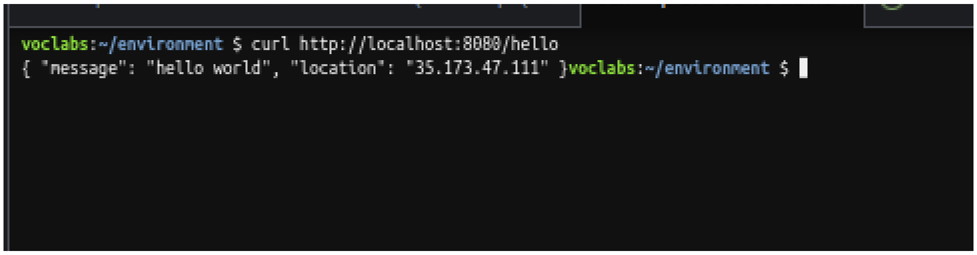
```
voclabs:~/environment $ curl http://localhost:8080/hello
{"message":"Internal server error"}
voclabs:~/environment $ curl http://localhost:8080/hello
{ "message": "hello world", "location": "35.173.47.111" }voclabs:~/environment $
```

-Figure 28-Showing application running



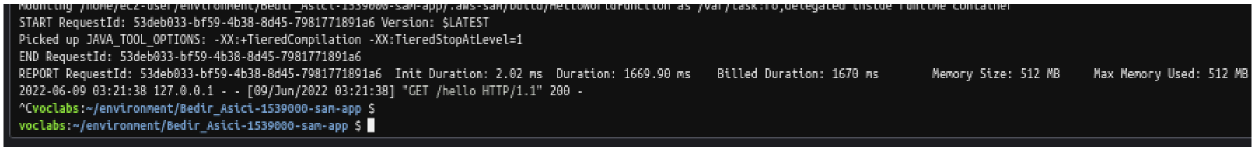
```
voclabs:~/environment/Bedir_Asici-1539000-sam-app $ sam local start-api --port 8080
Mounting HelloWorldFunction at http://127.0.0.1:8080/hello [GET]
You can now browse to the above endpoints to invoke your functions. You do not need to restart/reload SAM CLI while working on your functions, changes will be reflected instantly/automatically. You only need to restart SAM CLI if you update your AWS SAM template
2022-06-09 03:19:29  * Running on http://127.0.0.1:8080/ (Press CTRL+C to quit)
Invoking helloworld.App::handleRequest (java8)
Image was not found.
Removing rapid images for repo public.ecr.aws/sam/emulation-java8
Building image.......................................................................................................................................
Skip pulling image and use local one: public.ecr.aws/sam/emulation-java8:rapid-1.51.0-x86_64.

Mounting /home/ec2-user/environment/Bedir_Asici-1539000-sam-app/.aws-sam/build/HelloWorldFunction as /var/task:ro,delegated inside runtime container
Picked up JAVA_TOOL_OPTIONS: -XX:+TieredCompilation -XX:TieredStopAtLevel=1
END RequestId: 73cbfeaa-2c01-4181-91a2-543df2bb183e
REPORT RequestId: 73cbfeaa-2c01-4181-91a2-543df2bb183e  Init Duration: 0.75 ms  Duration: 2202.66 ms    Billed Duration: 2203 ms        Memory Size: 512 MB     Max Memory Used: 512 MB
2022-06-09 03:20:44 127.0.0.1 - - [09/Jun/2022 03:20:44] "GET /hello HTTP/1.1" 200 -
```

-Figure 29-Another example of server working



```
voclabs:~/environment $ curl http://localhost:8080/hello
{ "message": "hello world", "location": "35.173.47.111" }voclabs:~/environment $
```

-Figure 30-Exiting server

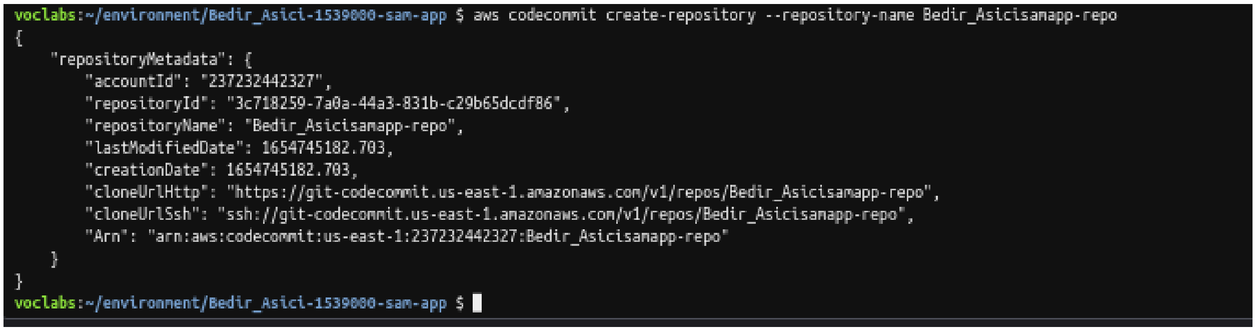


```
Mounting /home/ec2-user/environment/Bedir_Asici-1539000-sam-app/.aws-sam/build/HelloWorldFunction as /var/task:ro,delegated inside runtime container
START RequestId: 53deb033-bf59-4b38-8d45-7981771891a6 Version: $LATEST
Picked up JAVA_TOOL_OPTIONS: -XX:+TieredCompilation -XX:TieredStopAtLevel=1
END RequestId: 53deb033-bf59-4b38-8d45-7981771891a6
REPORT RequestId: 53deb033-bf59-4b38-8d45-7981771891a6  Init Duration: 2.02 ms  Duration: 1669.90 ms    Billed Duration: 1670 ms        Memory Size: 512 MB     Max Memory Used: 512 MB
2022-06-09 03:21:38 127.0.0.1 - - [09/Jun/2022 03:21:38] "GET /hello HTTP/1.1" 200 -
^Cvoclabs:~/environment/Bedir_Asici-1539000-sam-app $
voclabs:~/environment/Bedir_Asici-1539000-sam-app $
```

Task Four : Configuring git repository

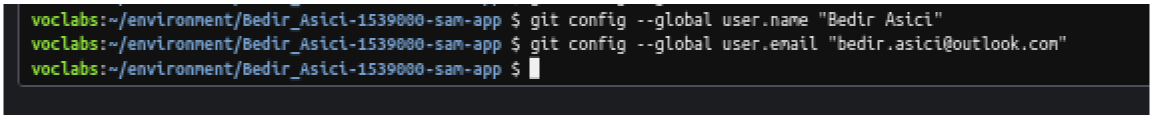
Using a git repository for development is beneficial as version control gives more flexibility to what you can do with code, rolling back changes, seeing processes and more.

-Figure 31-Showing repository



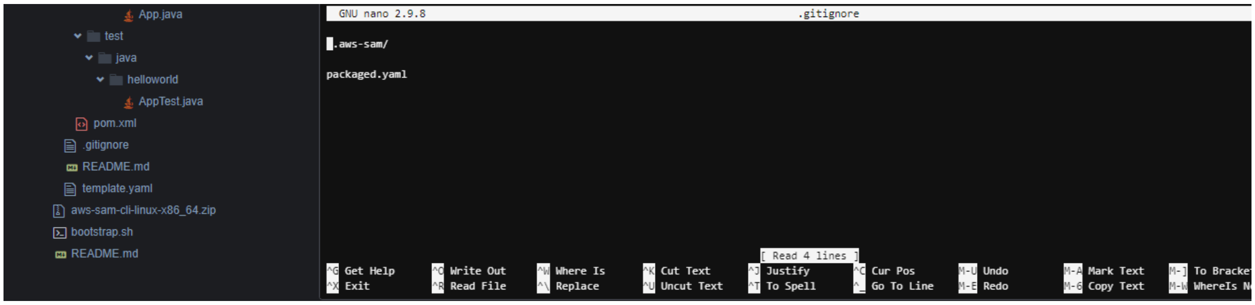
```
voclabs:~/environment/Bedir_Asici-1539000-sam-app $ aws codecommit create-repository --repository-name Bedir_Asicisamapp-repo
{
    "repositoryMetadata": {
        "accountId": "237232442327",
        "repositoryId": "3c718259-7a0a-44a3-831b-c29b65dcdf86",
        "repositoryName": "Bedir_Asicisamapp-repo",
        "lastModifiedDate": 1654745182.703,
        "creationDate": 1654745182.703,
        "cloneUrlHttp": "https://git-codecommit.us-east-1.amazonaws.com/v1/repos/Bedir_Asicisamapp-repo",
        "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/Bedir_Asicisamapp-repo",
        "Arn": "arn:aws:codecommit:us-east-1:237232442327:Bedir_Asicisamapp-repo"
    }
}
voclabs:~/environment/Bedir_Asici-1539000-sam-app $
```

-Figure 32-Setting details to the repository



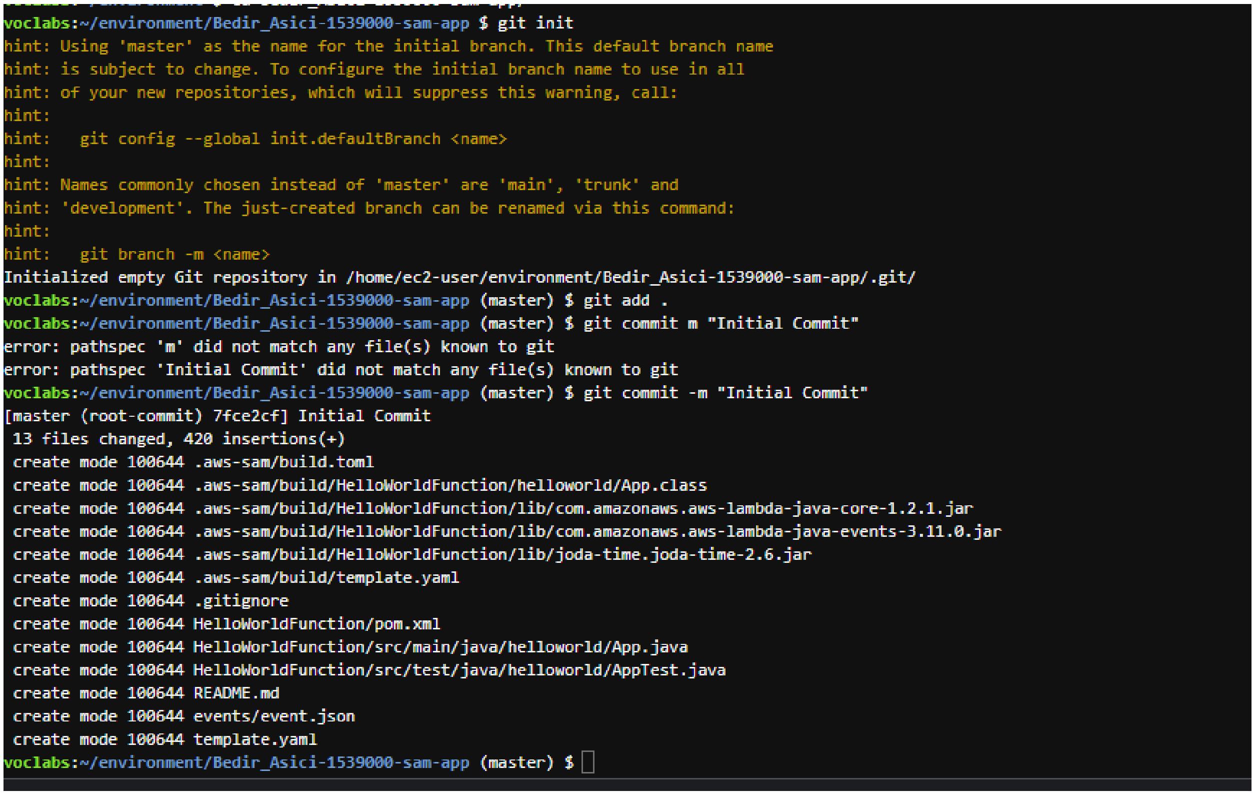
```
voclabs:~/environment/Bedir_Asici-1539000-sam-app $ git config --global user.name "Bedir Asici"
voclabs:~/environment/Bedir_Asici-1539000-sam-app $ git config --global user.email "bedir.asici@outlook.com"
voclabs:~/environment/Bedir_Asici-1539000-sam-app $
```

-Figure 33-Setting up the .gitignore file



```
.aws-sam/

packaged.yaml
```

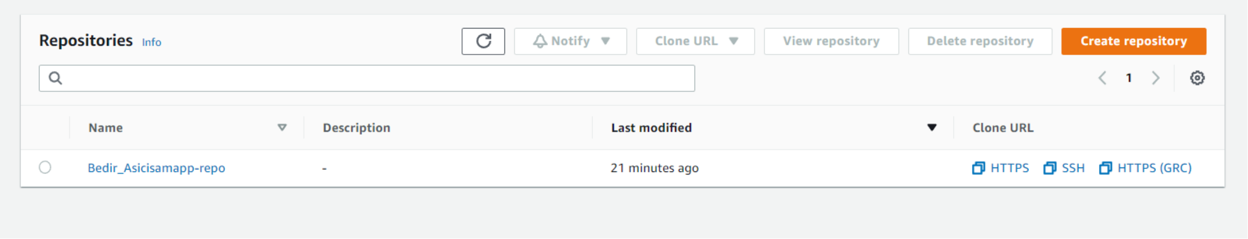
-Figure 34-Using git init to set up git



```
voclabs:~/environment/Bedir_Asici-1539000-sam-app $ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/ec2-user/environment/Bedir_Asici-1539000-sam-app/.git/
voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $ git add .
voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $ git commit m "Initial Commit"
error: pathspec 'm' did not match any file(s) known to git
error: pathspec 'Initial Commit' did not match any file(s) known to git
voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $ git commit -m "Initial Commit"
[master (root-commit) 7fce2cf] Initial Commit
 13 files changed, 420 insertions(+)
 create mode 100644 .aws-sam/build.toml
 create mode 100644 .aws-sam/build/HelloWorldFunction/helloworld/App.class
 create mode 100644 .aws-sam/build/HelloWorldFunction/lib/com.amazonaws.aws-lambda-java-core-1.2.1.jar
 create mode 100644 .aws-sam/build/HelloWorldFunction/lib/com.amazonaws.aws-lambda-java-events-3.11.0.jar
 create mode 100644 .aws-sam/build/HelloWorldFunction/lib/joda-time.joda-time-2.6.jar
 create mode 100644 .aws-sam/build/template.yaml
 create mode 100644 .gitignore
 create mode 100644 HelloWorldFunction/pom.xml
 create mode 100644 HelloWorldFunction/src/main/java/helloworld/App.java
 create mode 100644 HelloWorldFunction/src/test/java/helloworld/AppTest.java
 create mode 100644 README.md
 create mode 100644 events/event.json
 create mode 100644 template.yaml
voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $
```

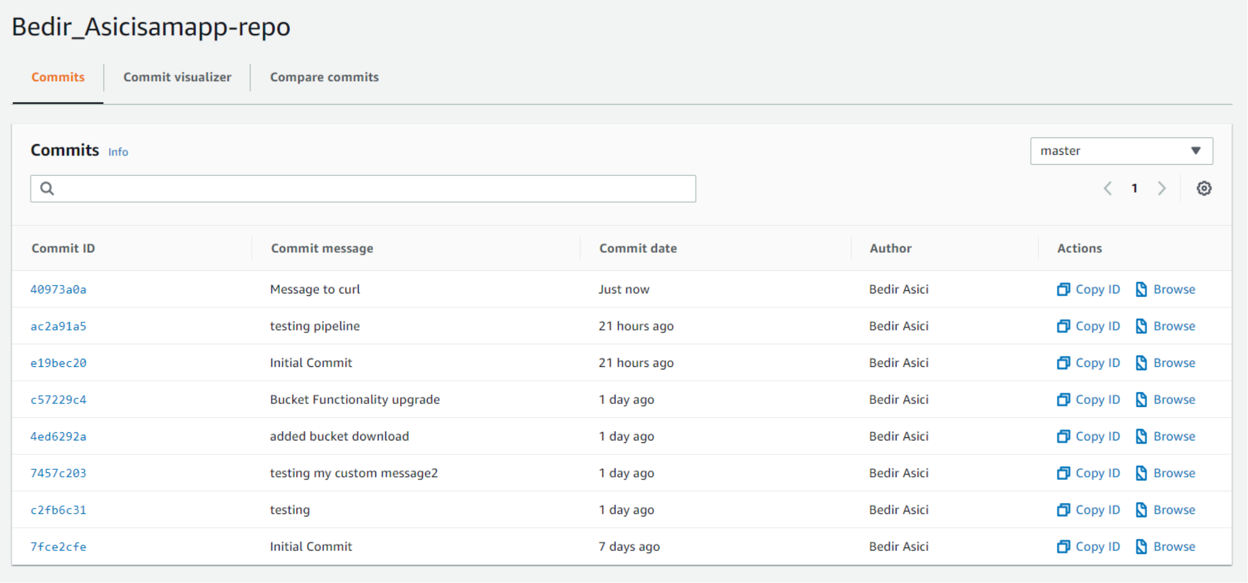
The commit history shown in Figure 36 has a duplicate message in “initial commit”. This is due to an error in which the commit history disappeared, however after adding a further commit brought it back.

-Figure 35-Showing existence of repository in code-commit



Name	Description	Last modified	Clone URL
Bedir_Asicisamapp-repo	-	21 minutes ago	HTTPS SSH HTTPS (GRC)

-Figure 36-Showing commit history

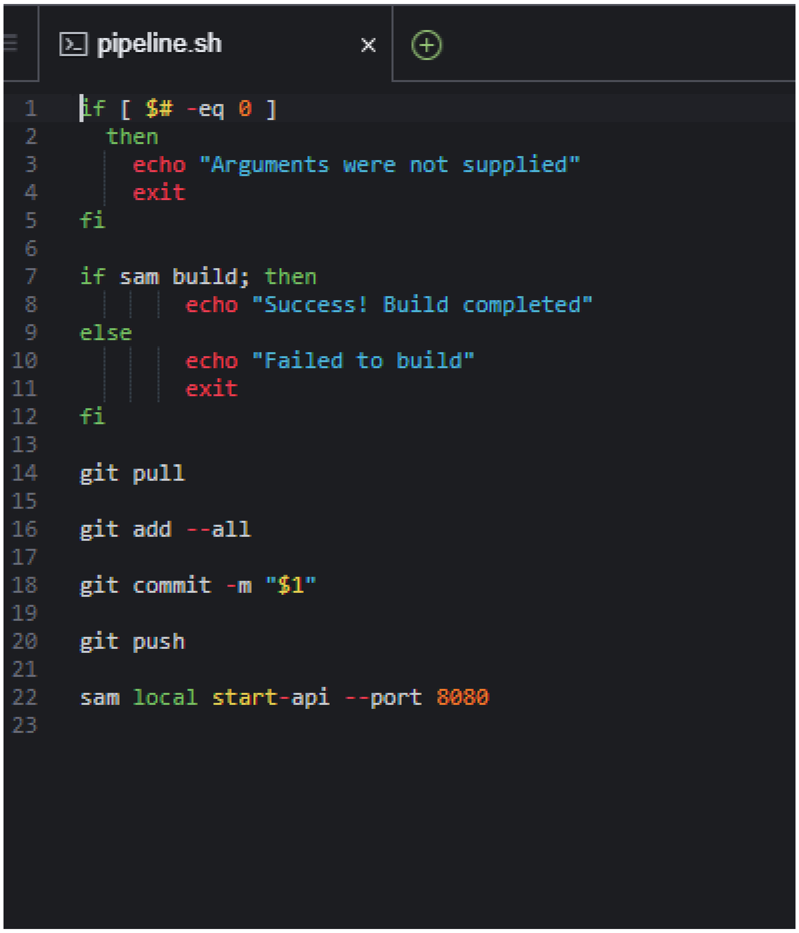


Commit ID	Commit message	Commit date	Author	Actions
40973a0a	Message to curl	Just now	Bedir Asici	Copy ID Browse
ac2a91a5	testing pipeline	21 hours ago	Bedir Asici	Copy ID Browse
e19bec20	Initial Commit	21 hours ago	Bedir Asici	Copy ID Browse
c57229c4	Bucket Functionality upgrade	1 day ago	Bedir Asici	Copy ID Browse
4ed6292a	added bucket download	1 day ago	Bedir Asici	Copy ID Browse
7457c203	testing my custom message2	1 day ago	Bedir Asici	Copy ID Browse
c2fb6c31	testing	1 day ago	Bedir Asici	Copy ID Browse
7fce2cfe	Initial Commit	7 days ago	Bedir Asici	Copy ID Browse

Task Five: Implementing a CI/CD pipeline

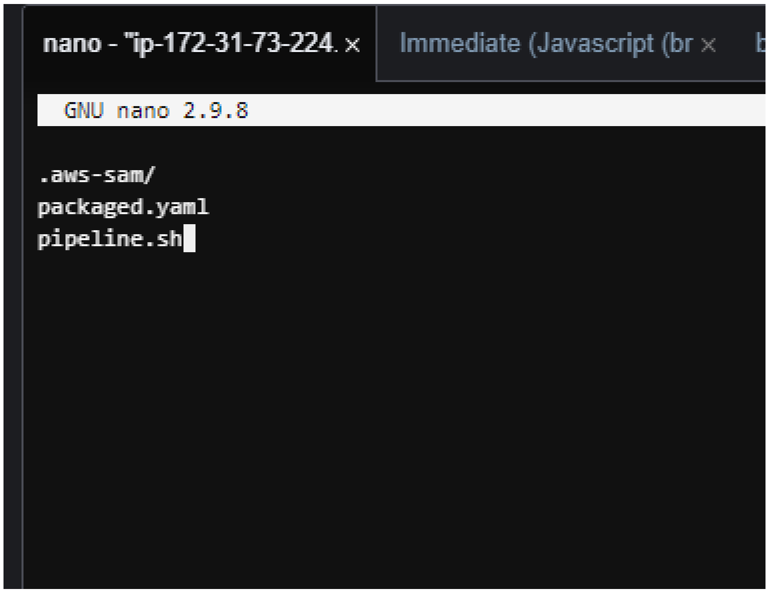
A CI/CD pipeline was used to pull , add . commit and push changes before launching the application. This pipeline required some further interaction with the user, requiring a commit message as an argument and then a further interaction via console asking for deployment permission.

-Figure 37-Showing pipeline.sh



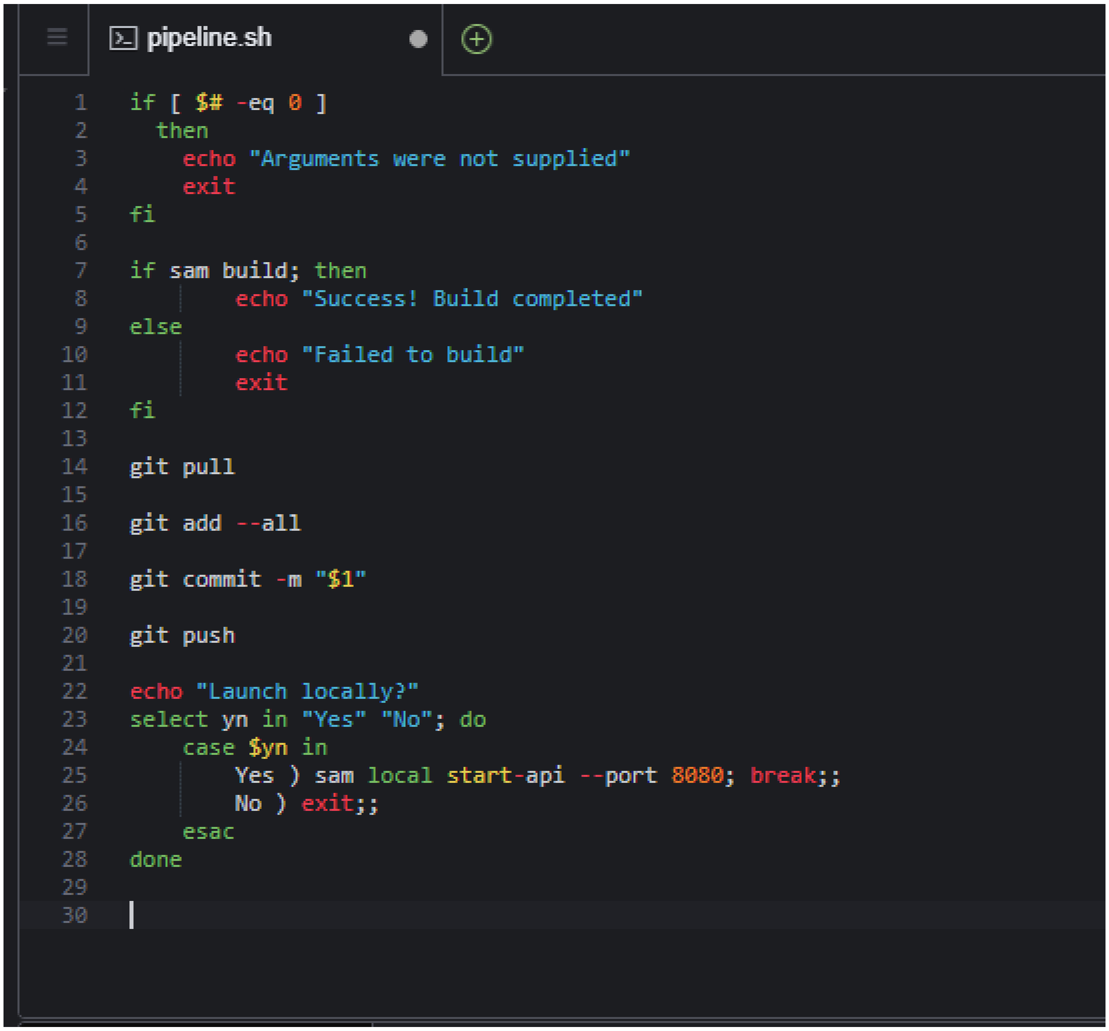
```
if [ $# -eq 0 ]
  then
    echo "Arguments were not supplied"
    exit
fi

if sam build; then
        echo "Success! Build completed"
else
        echo "Failed to build"
        exit
fi

git pull

git add --all

git commit -m "$1"

git push

sam local start-api --port 8080

```

-Figure 38-Updating .gitignore to include the pipeline.sh



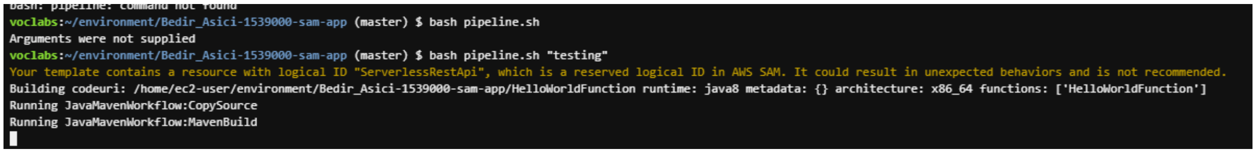
```
.aws-sam/
packaged.yaml
pipeline.sh
```

-Figure 39-Updating pipeline to include local deployability interaction



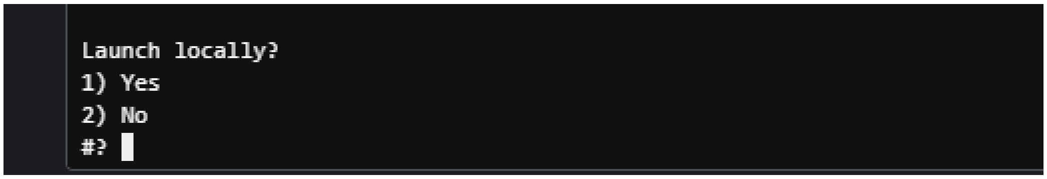
```
if [ $# -eq 0 ]
  then
    echo "Arguments were not supplied"
    exit
fi

if sam build; then
        echo "Success! Build completed"
else
        echo "Failed to build"
        exit
fi

git pull

git add --all

git commit -m "$1"

git push

echo "Launch locally?"
select yn in "Yes" "No"; do
    case $yn in
        Yes ) sam local start-api --port 8080; break;;
        No ) exit;;
    esac
done

```

-Figure 40-Showing the pipeline working pt.1



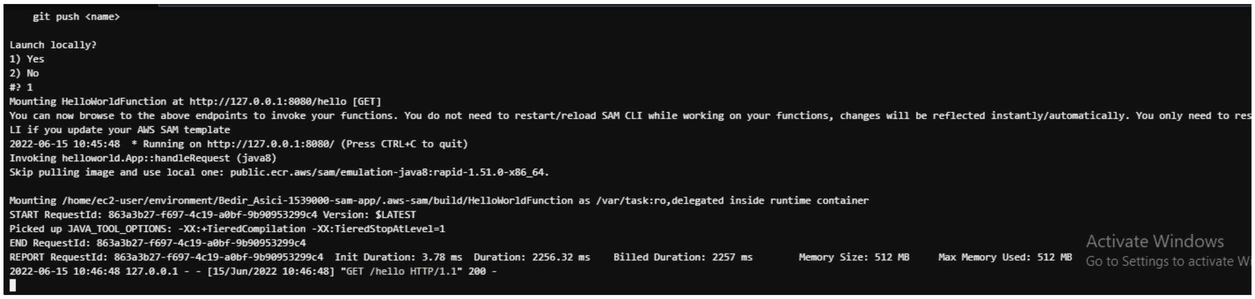
```
bash: pipeline: command not found
voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $ bash pipeline.sh
Arguments were not supplied
voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $ bash pipeline.sh "testing"
Your template contains a resource with logical ID "ServerlessRestApi", which is a reserved logical ID in AWS SAM. It could result in unexpected behaviors and is not recommended.
Building codeuri: /home/ec2-user/environment/Bedir_Asici-1539000-sam-app/HelloWorldFunction runtime: java8 metadata: {} architecture: x86_64 functions: ['HelloWorldFunction']
Running JavaMavenWorkflow:CopySource
Running JavaMavenWorkflow:MavenBuild
```

-Figure 41-Showing the pipeline working pt.2



```
Launch locally?
1) Yes
2) No
#?
```

-Figure 42-Showing the pipeline working pt.3

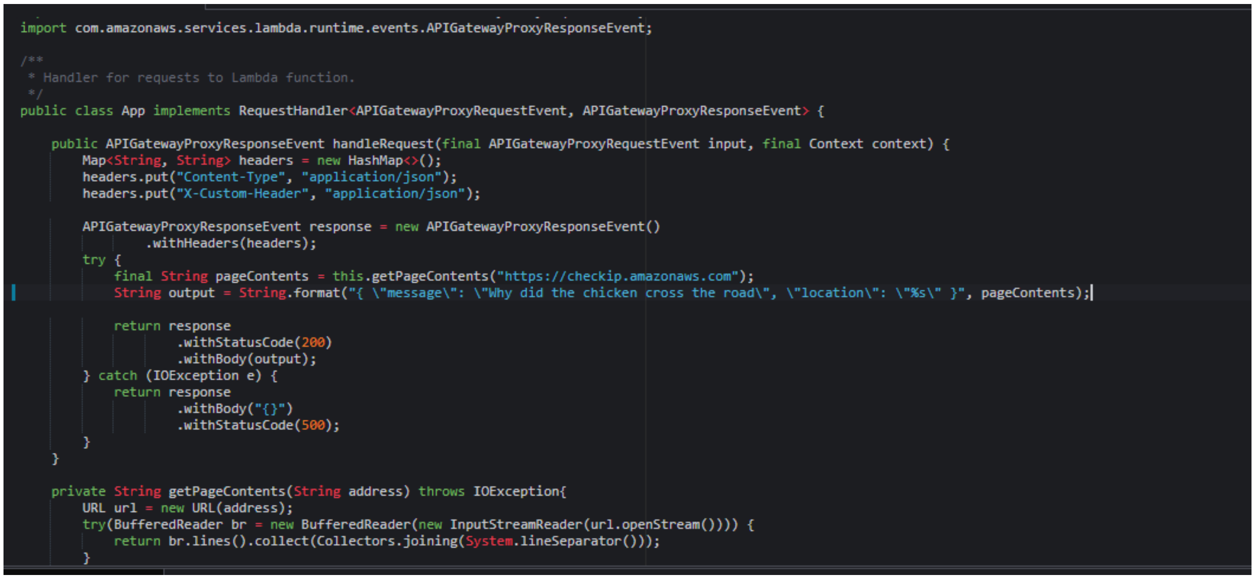


```
    git push <name>

Launch locally?
1) Yes
2) No
#? 1
Mounting HelloWorldFunction at http://127.0.0.1:8080/hello [GET]
You can now browse to the above endpoints to invoke your functions. You do not need to restart/reload SAM CLI while working on your functions, changes will be reflected instantly/automatically. You only need to res
LI if you update your AWS SAM template
2022-06-15 10:45:48  * Running on http://127.0.0.1:8080/ (Press CTRL+C to quit)
Invoking helloworld.App::handleRequest (java8)
Skip pulling image and use local one: public.ecr.aws/sam/emulation-java8:rapid-1.51.0-x86_64.

Mounting /home/ec2-user/environment/Bedir_Asici-1539000-sam-app/.aws-sam/build/HelloWorldFunction as /var/task:ro,delegated inside runtime container
START RequestId: 863a3b27-f697-4c19-a0bf-9b90953299c4 Version: $LATEST
Picked up JAVA_TOOL_OPTIONS: -XX:+TieredCompilation -XX:TieredStopAtLevel=1
END RequestId: 863a3b27-f697-4c19-a0bf-9b90953299c4
REPORT RequestId: 863a3b27-f697-4c19-a0bf-9b90953299c4  Init Duration: 3.78 ms  Duration: 2256.32 ms    Billed Duration: 2257 ms        Memory Size: 512 MB     Max Memory Used: 512 MB
2022-06-15 10:46:48 127.0.0.1 - - [15/Jun/2022 10:46:48] "GET /hello HTTP/1.1" 200 -
```

Task Six: Implementing Functionality and Testing

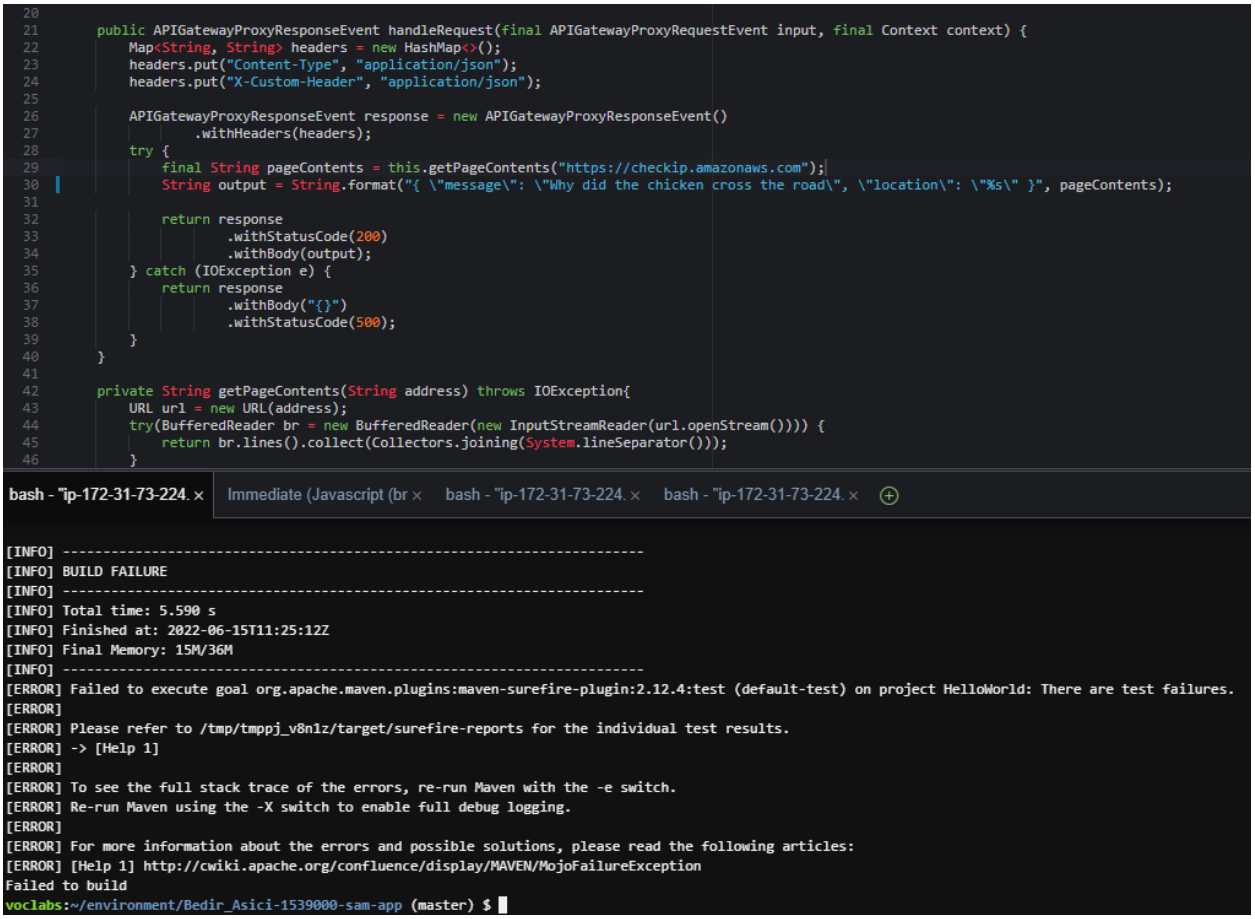
First a unique message was tested. Both the app.java line to pass the message as well as the test.java was edited to allow the message to go through.

-Figure 43-Code for custom message



```java
import com.amazonaws.services.lambda.runtime.events.APIGatewayProxyResponseEvent;

/**
 * Handler for requests to Lambda function.
 */
public class App implements RequestHandler<APIGatewayProxyRequestEvent, APIGatewayProxyResponseEvent> {

    public APIGatewayProxyResponseEvent handleRequest(final APIGatewayProxyRequestEvent input, final Context context) {
        Map<String, String> headers = new HashMap<>();
        headers.put("Content-Type", "application/json");
        headers.put("X-Custom-Header", "application/json");

        APIGatewayProxyResponseEvent response = new APIGatewayProxyResponseEvent()
                .withHeaders(headers);
        try {
            final String pageContents = this.getPageContents("https://checkip.amazonaws.com");
            String output = String.format("{ \"message\": \"Why did the chicken cross the road\", \"location\": \"%s\" }", pageContents);

            return response
                    .withStatusCode(200)
                    .withBody(output);
        } catch (IOException e) {
            return response
                    .withBody("{}")
                    .withStatusCode(500);
        }
    }

    private String getPageContents(String address) throws IOException{
        URL url = new URL(address);
        try(BufferedReader br = new BufferedReader(new InputStreamReader(url.openStream()))) {
            return br.lines().collect(Collectors.joining(System.lineSeparator()));
        }
```

-Figure 44-Code fails without changing apptest.java

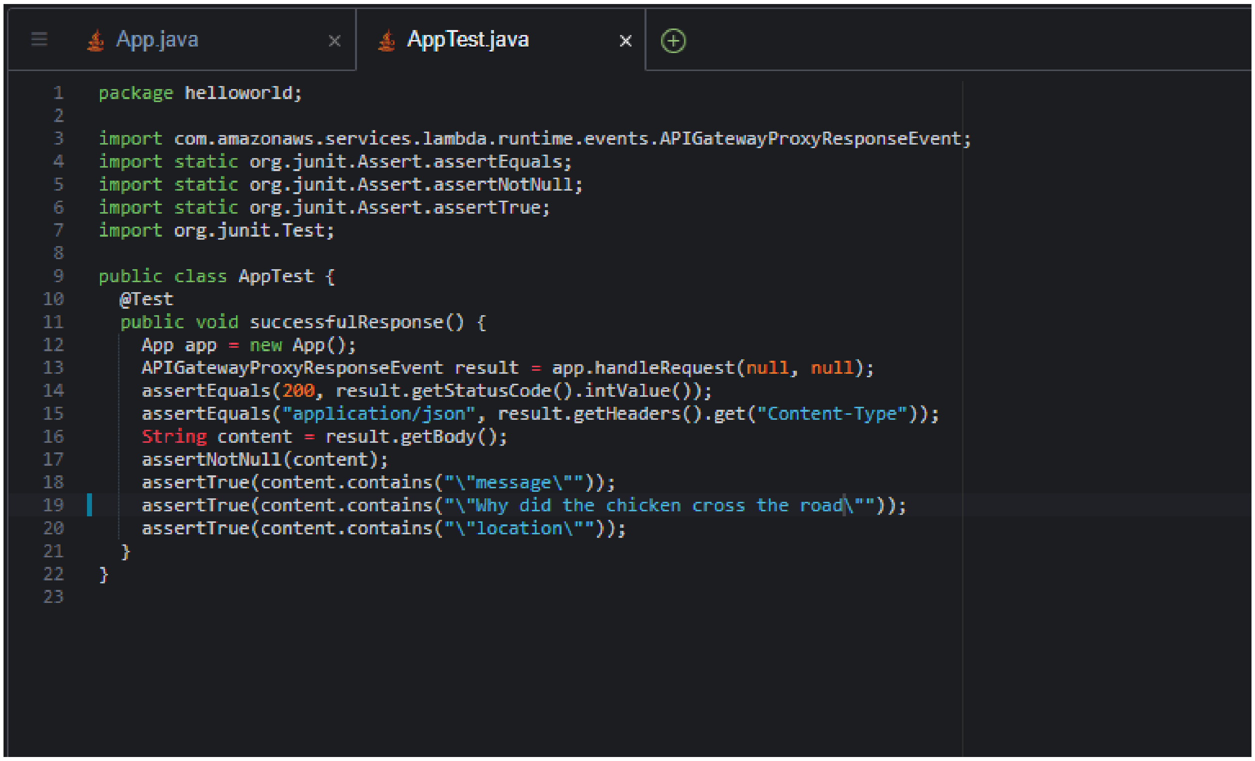


```java

    public APIGatewayProxyResponseEvent handleRequest(final APIGatewayProxyRequestEvent input, final Context context) {
        Map<String, String> headers = new HashMap<>();
        headers.put("Content-Type", "application/json");
        headers.put("X-Custom-Header", "application/json");

        APIGatewayProxyResponseEvent response = new APIGatewayProxyResponseEvent()
                .withHeaders(headers);
        try {
            final String pageContents = this.getPageContents("https://checkip.amazonaws.com");
            String output = String.format("{ \"message\": \"Why did the chicken cross the road\", \"location\": \"%s\" }", pageContents);

            return response
                    .withStatusCode(200)
                    .withBody(output);
        } catch (IOException e) {
            return response
                    .withBody("{}")
                    .withStatusCode(500);
        }
    }

    private String getPageContents(String address) throws IOException{
        URL url = new URL(address);
        try(BufferedReader br = new BufferedReader(new InputStreamReader(url.openStream()))) {
            return br.lines().collect(Collectors.joining(System.lineSeparator()));
        }
```

bash - "ip-172-31-73-224. ×
Immediate (Javascript (br ×
bash - "ip-172-31-73-224. ×
bash - "ip-172-31-73-224. ×

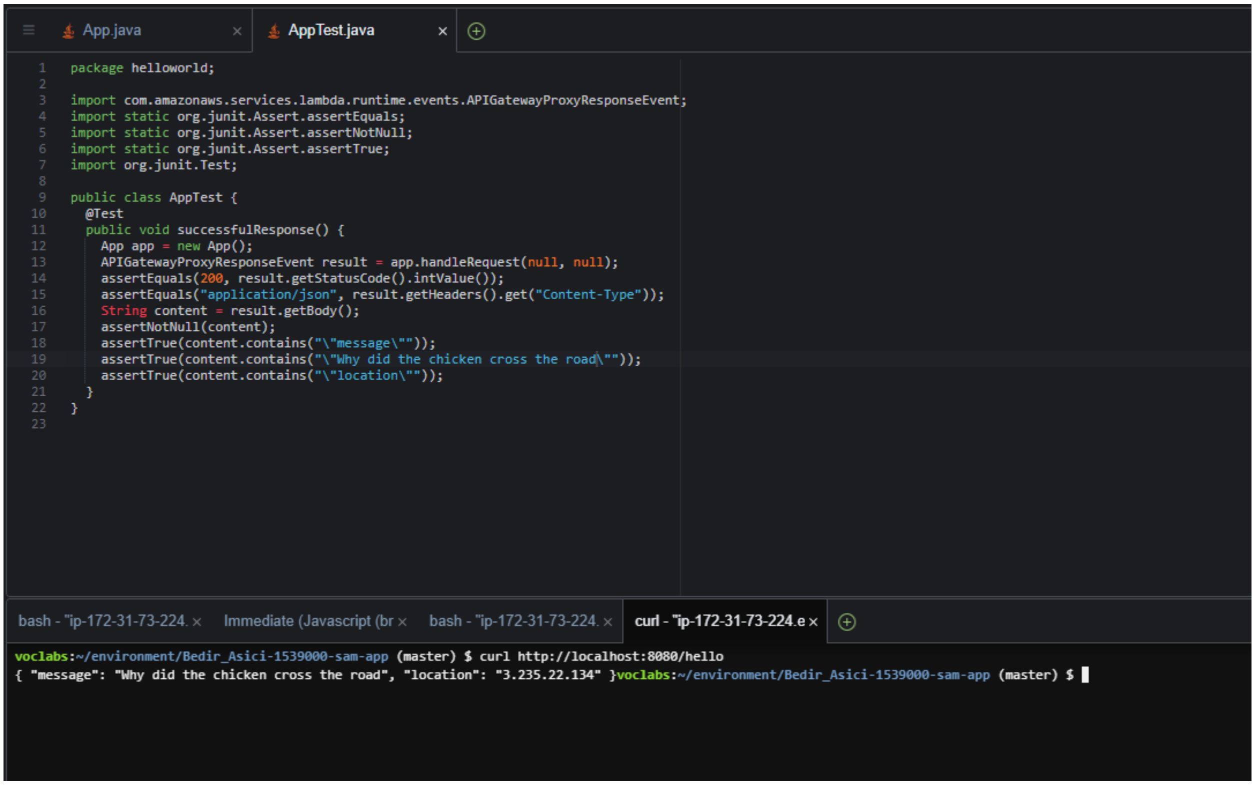
```
[INFO] ------------------------------------------------------------------------
[INFO] BUILD FAILURE
[INFO] ------------------------------------------------------------------------
[INFO] Total time: 5.590 s
[INFO] Finished at: 2022-06-15T11:25:12Z
[INFO] Final Memory: 15M/36M
[INFO] ------------------------------------------------------------------------
[ERROR] Failed to execute goal org.apache.maven.plugins:maven-surefire-plugin:2.12.4:test (default-test) on project HelloWorld: There are test failures.
[ERROR]
[ERROR] Please refer to /tmp/tmppj_v8n1z/target/surefire-reports for the individual test results.
[ERROR] -> [Help 1]
[ERROR]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.
[ERROR]
[ERROR] For more information about the errors and possible solutions, please read the following articles:
[ERROR] [Help 1] http://cwiki.apache.org/confluence/display/MAVEN/MojoFailureException
Failed to build
voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $
```

-Figure 45-Changing AppTest.java



```java
package helloworld;

import com.amazonaws.services.lambda.runtime.events.APIGatewayProxyResponseEvent;
import static org.junit.Assert.assertEquals;
import static org.junit.Assert.assertNotNull;
import static org.junit.Assert.assertTrue;
import org.junit.Test;

public class AppTest {
  @Test
  public void successfulResponse() {
    App app = new App();
    APIGatewayProxyResponseEvent result = app.handleRequest(null, null);
    assertEquals(200, result.getStatusCode().intValue());
    assertEquals("application/json", result.getHeaders().get("Content-Type"));
    String content = result.getBody();
    assertNotNull(content);
    assertTrue(content.contains("\"message\""));
    assertTrue(content.contains("\"Why did the chicken cross the road\""));
    assertTrue(content.contains("\"location\""));
  }
}
```

-Figure 46-Code works after test is updated

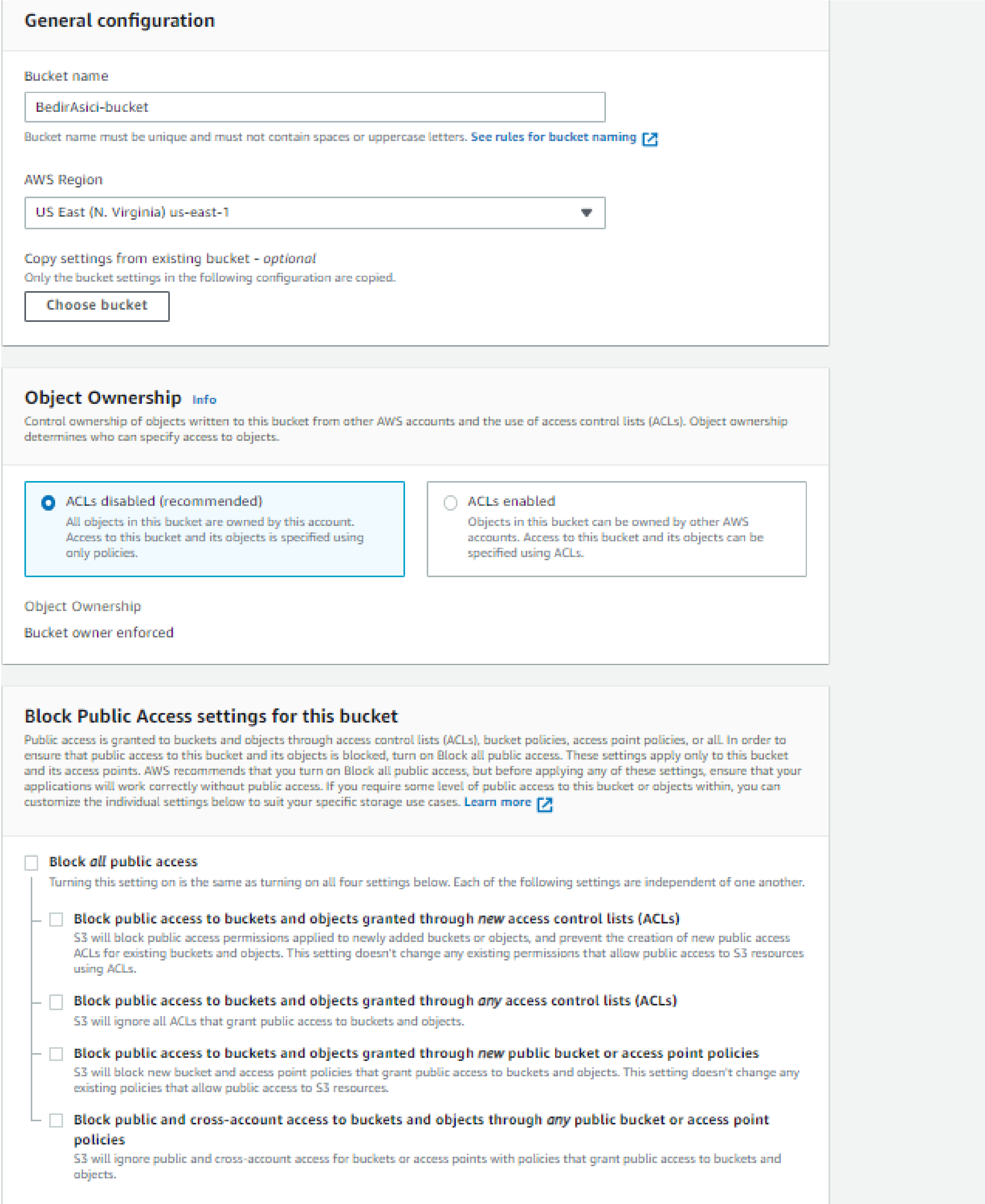

App.java
AppTest.java
bash - "ip-172-31-73-224.
Immediate (Javascript (br
bash - "ip-172-31-73-224.
curl - "ip-172-31-73-224.e

```java
package helloworld;

import com.amazonaws.services.lambda.runtime.events.APIGatewayProxyResponseEvent;
import static org.junit.Assert.assertEquals;
import static org.junit.Assert.assertNotNull;
import static org.junit.Assert.assertTrue;
import org.junit.Test;

public class AppTest {
  @Test
  public void successfulResponse() {
    App app = new App();
    APIGatewayProxyResponseEvent result = app.handleRequest(null, null);
    assertEquals(200, result.getStatusCode().intValue());
    assertEquals("application/json", result.getHeaders().get("Content-Type"));
    String content = result.getBody();
    assertNotNull(content);
    assertTrue(content.contains("\"message\""));
    assertTrue(content.contains("\"Why did the chicken cross the road\""));
    assertTrue(content.contains("\"location\""));
  }
}
```

```
voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $ curl http://localhost:8080/hello
{ "message": "Why did the chicken cross the road", "location": "3.235.22.134" }voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $
```

Setting up bucket usage.

-Figure 47-Setting bucket name and location



General configuration

Bucket name

BedirAsici-bucket

Bucket name must be unique and must not contain spaces or uppercase letters. See rules for bucket naming

AWS Region

US East (N. Virginia) us-east-1

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Choose bucket

Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

● ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

○ ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership

Bucket owner enforced

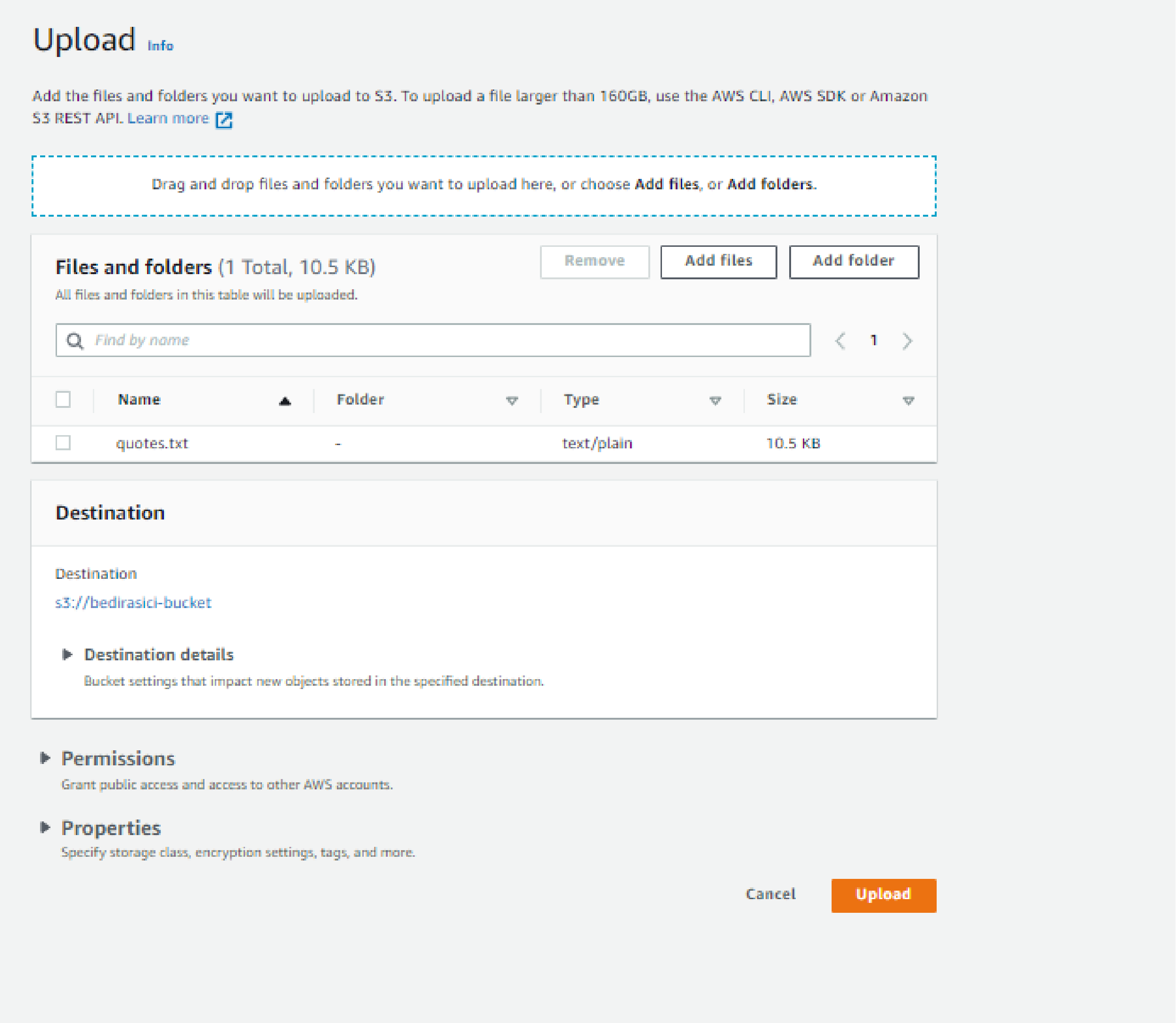
Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

□ Block *all* public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- □ Block public access to buckets and objects granted through *new* access control lists (ACLs)
  S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- □ Block public access to buckets and objects granted through *any* access control lists (ACLs)
  S3 will ignore all ACLs that grant public access to buckets and objects.
- □ Block public access to buckets and objects granted through *new* public bucket or access point policies
  S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- □ Block public and cross-account access to buckets and objects through *any* public bucket or access point policies
  S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

-Figure 48-Adding necessary files



Upload

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more

Drag and drop files and folders you want to upload here, or choose **Add files**, or **Add folders**.

Files and folders (1 Total, 10.5 KB)

All files and folders in this table will be uploaded.

□	Name ▲	Folder ▽	Type ▽	Size ▽
□	quotes.txt	-	text/plain	10.5 KB

Destination

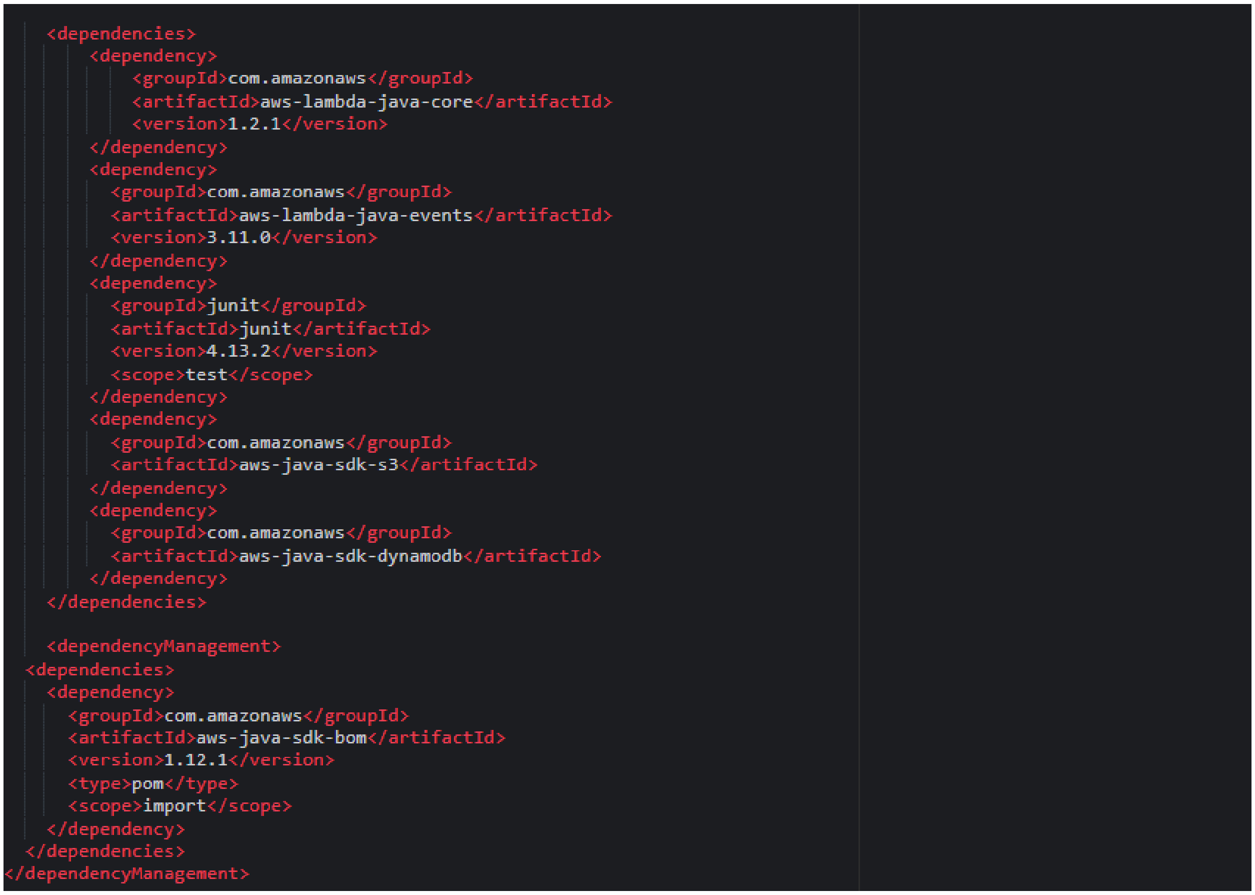
Destination
s3://bedirasici-bucket

▶ **Destination details**
Bucket settings that impact new objects stored in the specified destination.

▶ **Permissions**
Grant public access and access to other AWS accounts.

▶ **Properties**
Specify storage class, encryption settings, tags, and more.

-Figure 49-Pom.xml updates

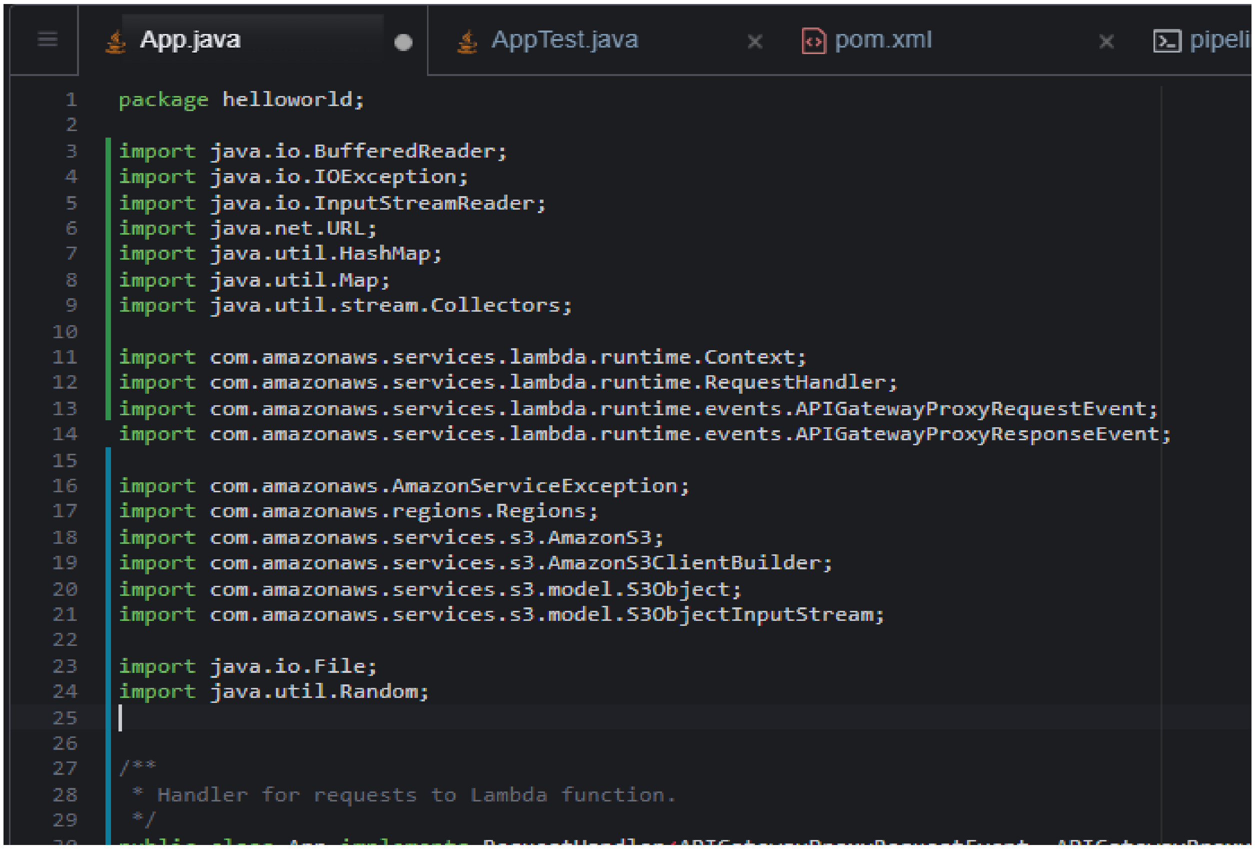


```
    <dependencies>
        <dependency>
            <groupId>com.amazonaws</groupId>
            <artifactId>aws-lambda-java-core</artifactId>
            <version>1.2.1</version>
        </dependency>
        <dependency>
          <groupId>com.amazonaws</groupId>
          <artifactId>aws-lambda-java-events</artifactId>
          <version>3.11.0</version>
        </dependency>
        <dependency>
          <groupId>junit</groupId>
          <artifactId>junit</artifactId>
          <version>4.13.2</version>
          <scope>test</scope>
        </dependency>
        <dependency>
          <groupId>com.amazonaws</groupId>
          <artifactId>aws-java-sdk-s3</artifactId>
        </dependency>
        <dependency>
          <groupId>com.amazonaws</groupId>
          <artifactId>aws-java-sdk-dynamodb</artifactId>
        </dependency>
    </dependencies>

    <dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>com.amazonaws</groupId>
      <artifactId>aws-java-sdk-bom</artifactId>
      <version>1.12.1</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
```

Pom.xml edit

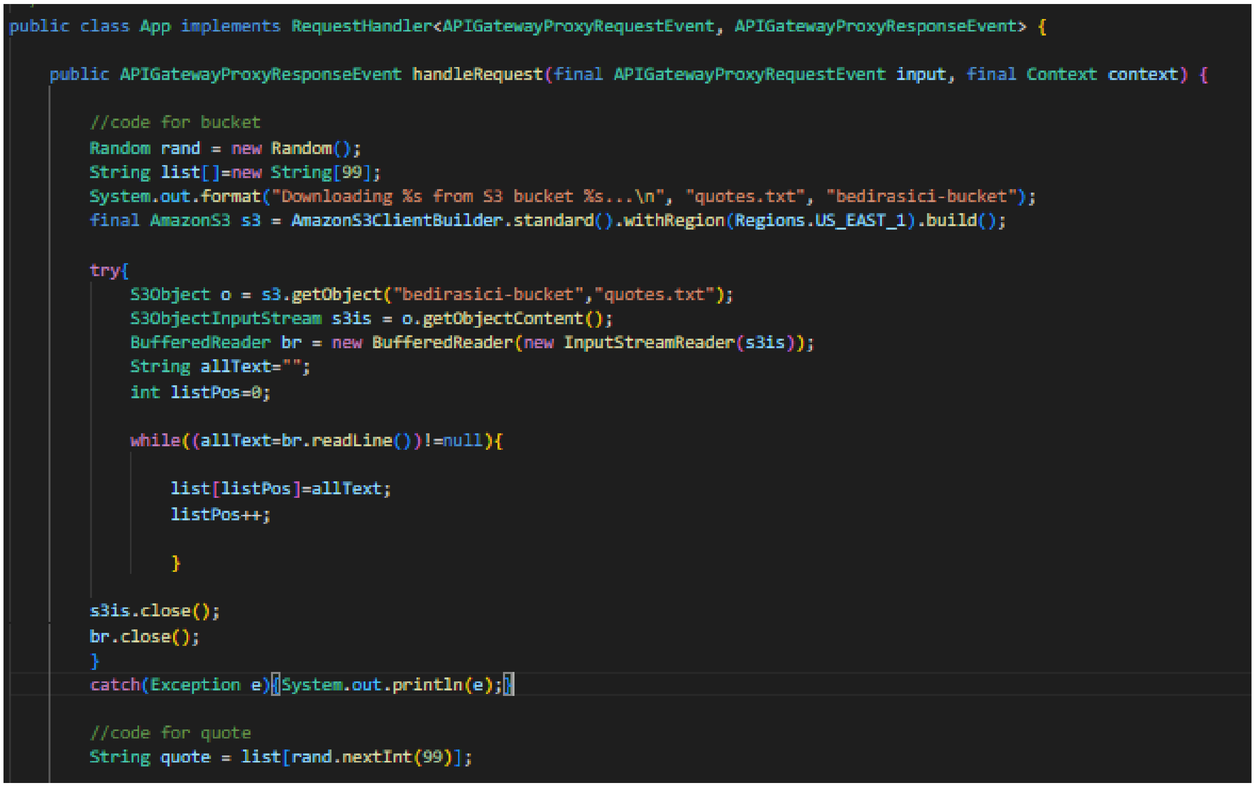
The edits in Figure 49 were the dependency manager specifying the version of aws java sdk to use. And the dependency of the java sdk3. The version 1.12.1 was used simply to show that different versions can be used. The minimum that can be used is a subset of 1.11.

-Figure 50-Needed imports for processing S3Bucket



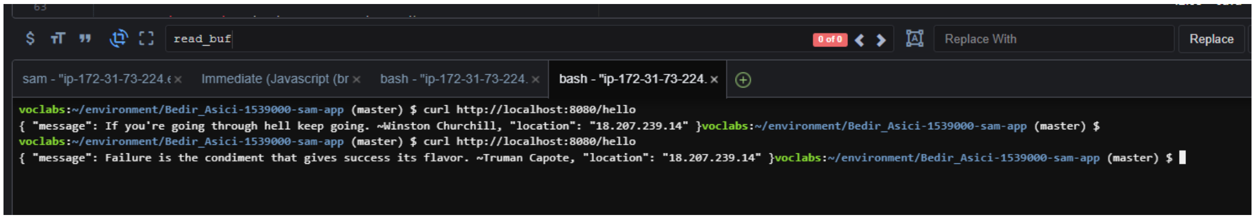
```java
package helloworld;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.net.URL;
import java.util.HashMap;
import java.util.Map;
import java.util.stream.Collectors;

import com.amazonaws.services.lambda.runtime.Context;
import com.amazonaws.services.lambda.runtime.RequestHandler;
import com.amazonaws.services.lambda.runtime.events.APIGatewayProxyRequestEvent;
import com.amazonaws.services.lambda.runtime.events.APIGatewayProxyResponseEvent;

import com.amazonaws.AmazonServiceException;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.S3Object;
import com.amazonaws.services.s3.model.S3ObjectInputStream;

import java.io.File;
import java.util.Random;



/**
 * Handler for requests to Lambda function.
 */
```

-Figure 51-Code for processing bucket



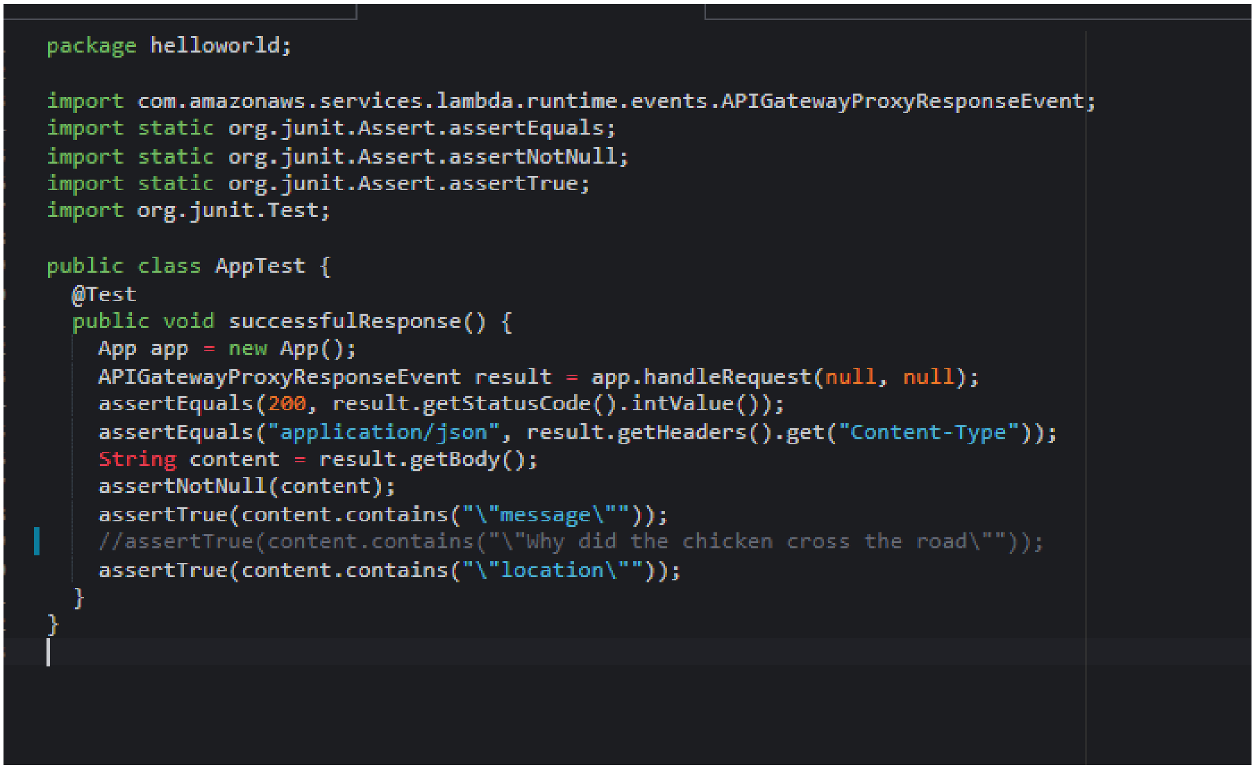
```java
public class App implements RequestHandler<APIGatewayProxyRequestEvent, APIGatewayProxyResponseEvent> {

    public APIGatewayProxyResponseEvent handleRequest(final APIGatewayProxyRequestEvent input, final Context context) {

        //code for bucket
        Random rand = new Random();
        String list[]=new String[99];
        System.out.format("Downloading %s from S3 bucket %s...\n", "quotes.txt", "bedirasici-bucket");
        final AmazonS3 s3 = AmazonS3ClientBuilder.standard().withRegion(Regions.US_EAST_1).build();

        try{
            S3Object o = s3.getObject("bedirasici-bucket","quotes.txt");
            S3ObjectInputStream s3is = o.getObjectContent();
            BufferedReader br = new BufferedReader(new InputStreamReader(s3is));
            String allText="";
            int listPos=0;

            while((allText=br.readLine())!=null){

                list[listPos]=allText;
                listPos++;

                }

        s3is.close();
        br.close();
        }
        catch(Exception e){System.out.println(e);}

        //code for quote
        String quote = list[rand.nextInt(99)];
```

-Figure 52-Showing the random quote query working


read_buf
0 of 0
Replace With
Replace
sam - "ip-172-31-73-224.
Immediate (Javascript (br
bash - "ip-172-31-73-224.
bash - "ip-172-31-73-224.

```
voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $ curl http://localhost:8080/hello
{ "message": If you're going through hell keep going. ~Winston Churchill, "location": "18.207.239.14" }voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $
voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $ curl http://localhost:8080/hello
{ "message": Failure is the condiment that gives success its flavor. ~Truman Capote, "location": "18.207.239.14" }voclabs:~/environment/Bedir_Asici-1539000-sam-app (master) $
```

-Figure 53-Showing removed unit test



```java
package helloworld;

import com.amazonaws.services.lambda.runtime.events.APIGatewayProxyResponseEvent;
import static org.junit.Assert.assertEquals;
import static org.junit.Assert.assertNotNull;
import static org.junit.Assert.assertTrue;
import org.junit.Test;

public class AppTest {
  @Test
  public void successfulResponse() {
    App app = new App();
    APIGatewayProxyResponseEvent result = app.handleRequest(null, null);
    assertEquals(200, result.getStatusCode().intValue());
    assertEquals("application/json", result.getHeaders().get("Content-Type"));
    String content = result.getBody();
    assertNotNull(content);
    assertTrue(content.contains("\"message\""));
    //assertTrue(content.contains("\"Why did the chicken cross the road\""));
    assertTrue(content.contains("\"location\""));
  }
}
```

Discussion:

Building of the pipeline script was made to be as simple as possible. First arguments were counted, the number of arguments was equal to 0, the script would exit with a prompt for user input. Next it would check if the application can be built, if it can then the next stage is started, however if there are build errors the application is not pushed, as we don't want erroneous code pushed to master and it is the pipeline's job to filter out these errors. Passing this it would pull first, this is more important when working in groups as merging locally is better than merging in the deployed and active file that the team is working from. Next all current changes were added, committed using the user input and pushed. As a result of not having the ability to deploy globally, the user is asked within the script interaction if they wish to deploy locally. Inputting the numbers 1 or 2 in this scenario will let the user decide to launch or not. On selection the pipeline is finished and application starts or is ended based on user choice.

To read in the bucket the location, bucket name and stored document name were given to create the correct object. This object was then used to create an object input stream. This stream only reads bytes, therefore this stream was used as a parameter to initialize a buffered reader which can read into string from the object. The file was read line by line, after which it was added to an array. It could have been inserted into an arraylist to avoid declaring a size on initialization; however in an effort to not import extra to conserve memory, the array was used instead. After being added to the list. A random number generator was to select a string. This was then put into the message ready to be queried.

For the testing section of exercise 6 there were a few options to consider. The first option was having a quotes.txt local file. This file would be read in using a buffered reader then split into an array. Then the random quote could be tested to see if it was in the array read in by the test. This was not implemented as during the development of this project, aws warnings concerning lack of memory were expressed during cloud 9 ide use. Therefore even though the current quotes .txt is decently small, the act of reading it all in especially if it was a non static file increasing in size during application use seemed like an inefficient method.

The second option considered was a regex. A regex-like operation could be applied to the quote seeing if it had the correct structure of a quote : a full sentence followed by ~ author. This could be further utilized to only allow quotes from non anonymous authors or other limiting factors to make the end result more moldable. A test is given below, the regex is represented in a simplified sql syntax where % is a string of any length.

It reads : any capital letter, followed by any string followed by . ~ followed by any string.

assertTrue(content.contains("[A-Z]%'. ~'%");