1. Serverless Architecture & Workflow

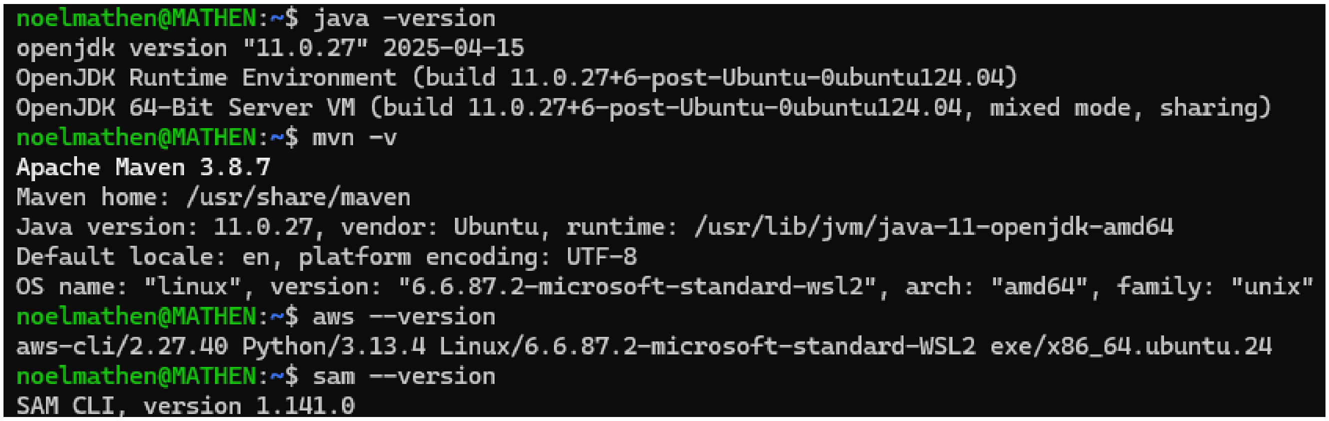
1. **Definition**
   - **Serverless** (FaaS) means you write individual “functions,” and a cloud provider automatically:
     - Provisions compute behind the scenes
     - Autos-scales to zero (no functions running → no cost)
     - Bills you per-invocation + compute-time
   - You never manage servers or VMs directly.
2. **Key Components**
   - **Event Source**: e.g. API Gateway, S3 event, message queue
   - **Function**: your code packaged as a handler (NodeJS, Java, Python, or container)
   - **Compute Fabric**: opaque to you—spins up containers or VMs on demand
   - **Trigger & Execution**: event → cloud event router → function container → runs your code → returns result
   - **Monitoring & Logging**: e.g. CloudWatch captures logs/metrics automatically
3. **Typical Workflow**

   Developer writes function handler → Packages code (+deps) as ZIP or container image → Deploys via CLI/console/IaC (SAM, Serverless Framework) → Cloud provider registers event source (API, queue, cron…) → Event occurs → provider fetches code → runs in ephemeral container → returns response → Logs emitted to monitoring service → container destroyed (or frozen until next call)

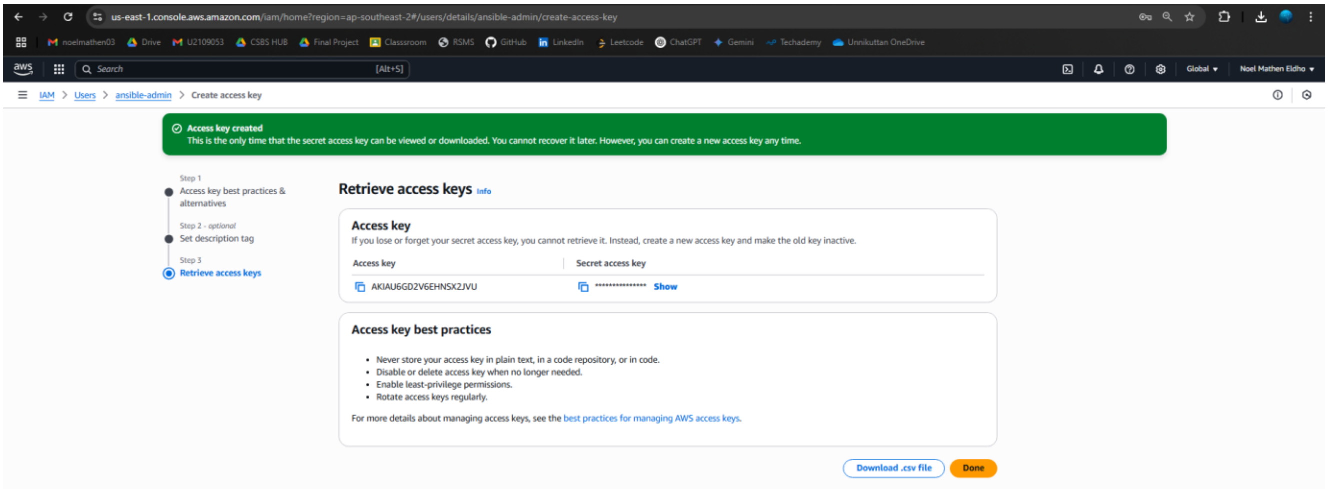
2. “Hello World” Serverless in Java

Commands:

1. sudo apt update && sudo apt install -y unzip curl git
2. sudo apt install -y openjdk-11-jdk maven
3. curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
4. unzip awscliv2.zip
5. sudo ./aws/install
6. curl -Lo sam.zip https://github.com/aws/aws-sam-cli/releases/latest/download/aws-sam-cli-linux-x86_64.zip
7. unzip sam.zip -d sam-installation
8. sudo ./sam-installation/install
9. java -version
10. mvn -v
11. aws --version
12. sam --version



```
noelmathen@MATHEN:~$ java -version
openjdk version "11.0.27" 2025-04-15
OpenJDK Runtime Environment (build 11.0.27+6-post-Ubuntu-0ubuntu124.04)
OpenJDK 64-Bit Server VM (build 11.0.27+6-post-Ubuntu-0ubuntu124.04, mixed mode, sharing)
noelmathen@MATHEN:~$ mvn -v
Apache Maven 3.8.7
Maven home: /usr/share/maven
Java version: 11.0.27, vendor: Ubuntu, runtime: /usr/lib/jvm/java-11-openjdk-amd64
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "6.6.87.2-microsoft-standard-wsl2", arch: "amd64", family: "unix"
noelmathen@MATHEN:~$ aws --version
aws-cli/2.27.40 Python/3.13.4 Linux/6.6.87.2-microsoft-standard-WSL2 exe/x86_64.ubuntu.24
noelmathen@MATHEN:~$ sam --version
SAM CLI, version 1.141.0
```



Access key created
This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time.

Step 1
● Access key best practices & alternatives

Step 2 - *optional*
● Set description tag

Step 3
◉ Retrieve access keys

Retrieve access keys Info

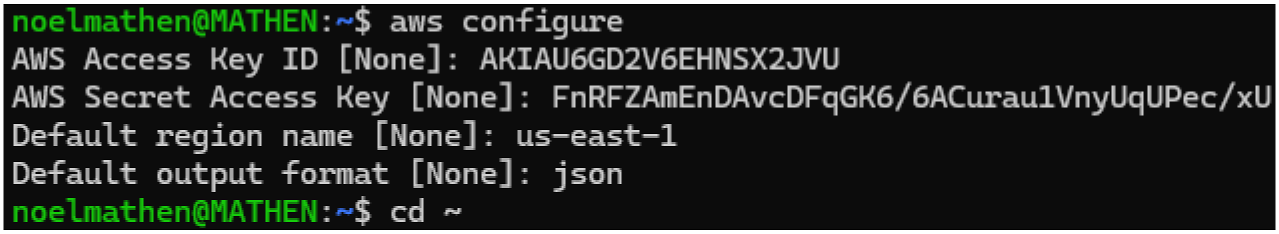
Access key
If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key	Secret access key
AKIAU6GD2V6EHNSX2JVU	*************** Show

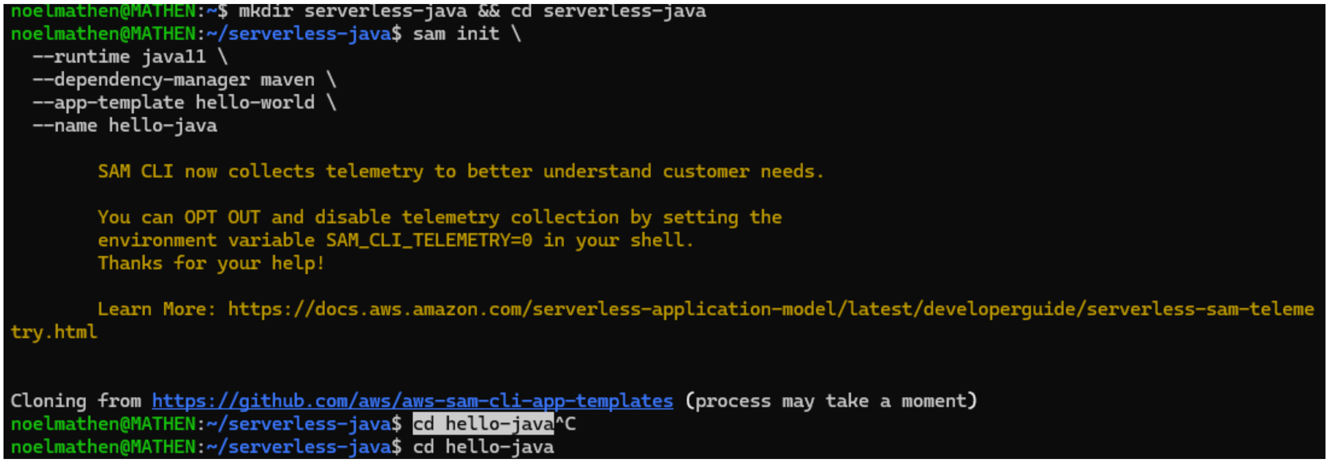
Access key best practices

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

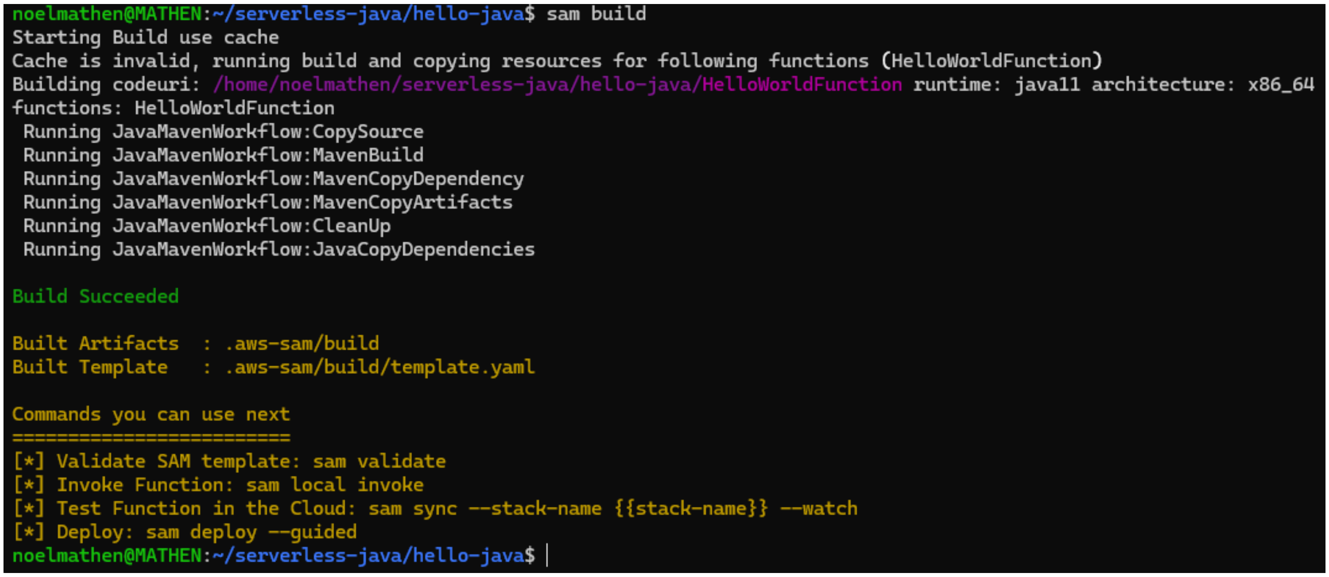
For more details about managing access keys, see the best practices for managing AWS access keys.



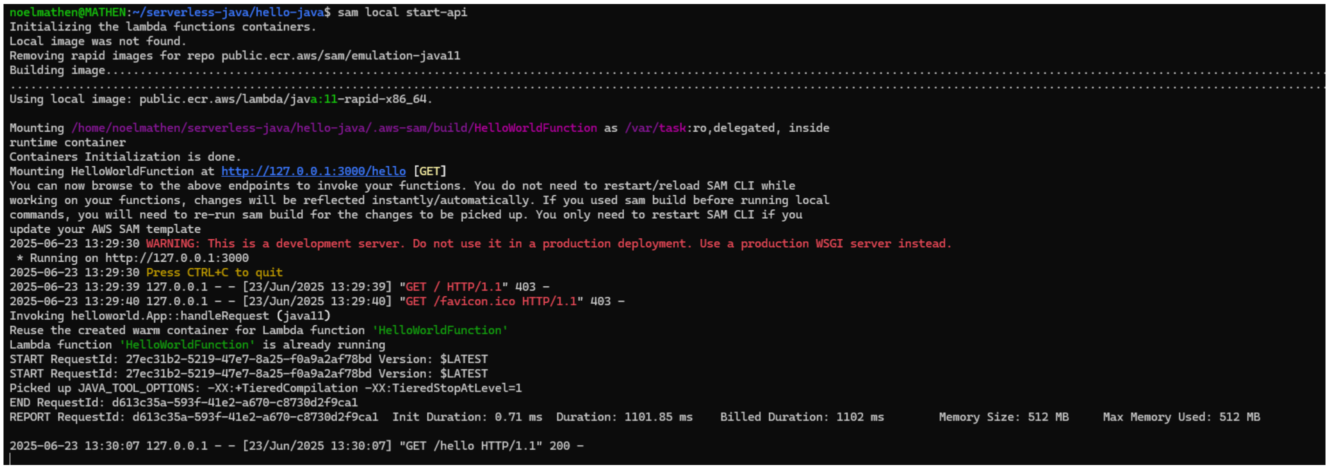
```
noelmathen@MATHEN:~$ aws configure
AWS Access Key ID [None]: AKIAU6GD2V6EHNSX2JVU
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]: json
noelmathen@MATHEN:~$ cd ~
```



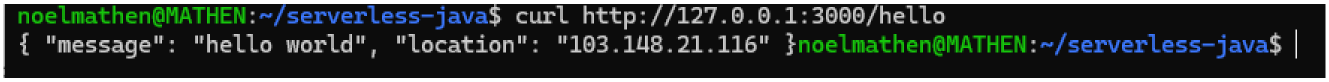
```
noelmathen@MATHEN:~$ mkdir serverless-java && cd serverless-java
noelmathen@MATHEN:~/serverless-java$ sam init \
  --runtime java11 \
  --dependency-manager maven \
  --app-template hello-world \
  --name hello-java

        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-teleme
try.html


Cloning from https://github.com/aws/aws-sam-cli-app-templates (process may take a moment)
noelmathen@MATHEN:~/serverless-java$ cd hello-java^C
noelmathen@MATHEN:~/serverless-java$ cd hello-java
```



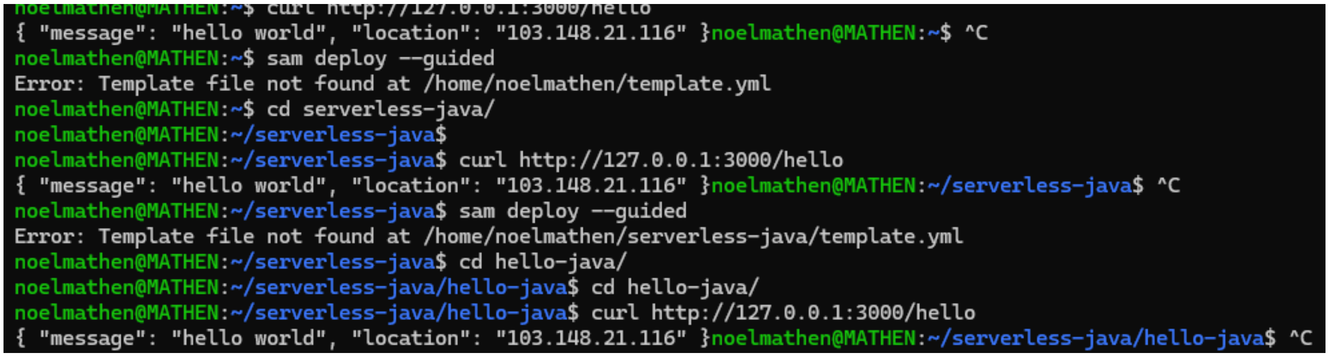
```
noelmathen@MATHEN:~/serverless-java/hello-java$ sam build
Starting Build use cache
Cache is invalid, running build and copying resources for following functions (HelloWorldFunction)
Building codeuri: /home/noelmathen/serverless-java/hello-java/HelloWorldFunction runtime: java11 architecture: x86_64
functions: HelloWorldFunction
 Running JavaMavenWorkflow:CopySource
 Running JavaMavenWorkflow:MavenBuild
 Running JavaMavenWorkflow:MavenCopyDependency
 Running JavaMavenWorkflow:MavenCopyArtifacts
 Running JavaMavenWorkflow:CleanUp
 Running JavaMavenWorkflow:JavaCopyDependencies

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
noelmathen@MATHEN:~/serverless-java/hello-java$
```



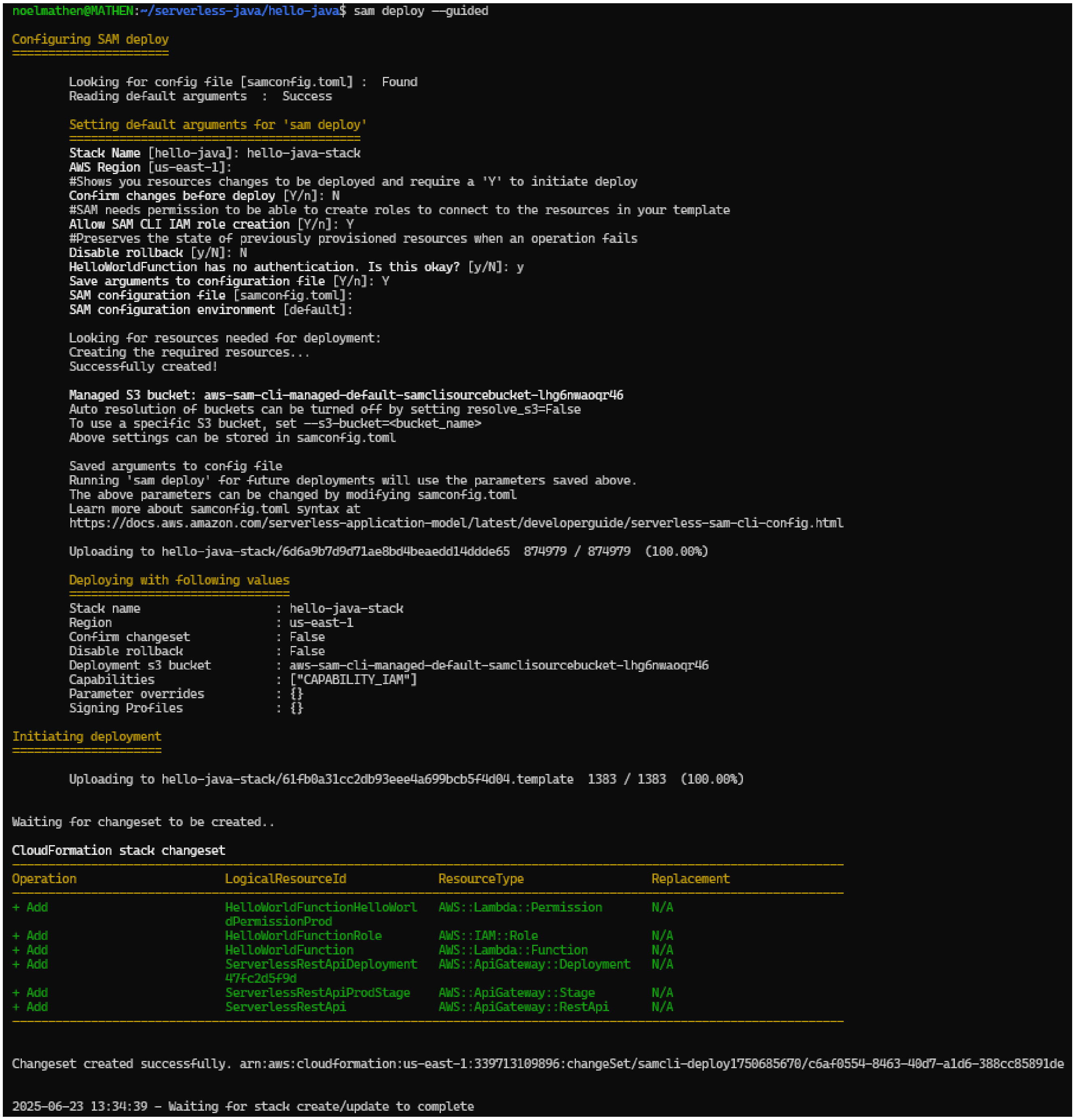
```
noelmathen@MATHEN:~/serverless-java/hello-java$ sam local start-api
Initializing the lambda functions containers.
Local image was not found.
Removing rapid images for repo public.ecr.aws/sam/emulation-java11
Building image...........................................................................................................................................................................
.................................................................................................................................................................................................
Using local image: public.ecr.aws/lambda/java:11-rapid-x86_64.

Mounting /home/noelmathen/serverless-java/hello-java/.aws-sam/build/HelloWorldFunction as /var/task:ro,delegated, inside
runtime container
Containers Initialization is done.
Mounting HelloWorldFunction at http://127.0.0.1:3000/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. If you used sam build before running local
commands, you will need to re-run sam build for the changes to be picked up. You only need to restart SAM CLI if you
update your AWS SAM template
2025-06-23 13:29:30 WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:3000
2025-06-23 13:29:30 Press CTRL+C to quit
2025-06-23 13:29:39 127.0.0.1 - - [23/Jun/2025 13:29:39] "GET / HTTP/1.1" 403 -
2025-06-23 13:29:40 127.0.0.1 - - [23/Jun/2025 13:29:40] "GET /favicon.ico HTTP/1.1" 403 -
Invoking helloworld.App::handleRequest (java11)
Reuse the created warm container for Lambda function 'HelloWorldFunction'
Lambda function 'HelloWorldFunction' is already running
START RequestId: 27ec31b2-5219-47e7-8a25-f0a9a2af78bd Version: $LATEST
START RequestId: 27ec31b2-5219-47e7-8a25-f0a9a2af78bd Version: $LATEST
Picked up JAVA_TOOL_OPTIONS: -XX:+TieredCompilation -XX:TieredStopAtLevel=1
END RequestId: d613c35a-593f-41e2-a670-c8730d2f9ca1
REPORT RequestId: d613c35a-593f-41e2-a670-c8730d2f9ca1  Init Duration: 0.71 ms  Duration: 1101.85 ms    Billed Duration: 1102 ms        Memory Size: 512 MB     Max Memory Used: 512 MB

2025-06-23 13:30:07 127.0.0.1 - - [23/Jun/2025 13:30:07] "GET /hello HTTP/1.1" 200 -
```



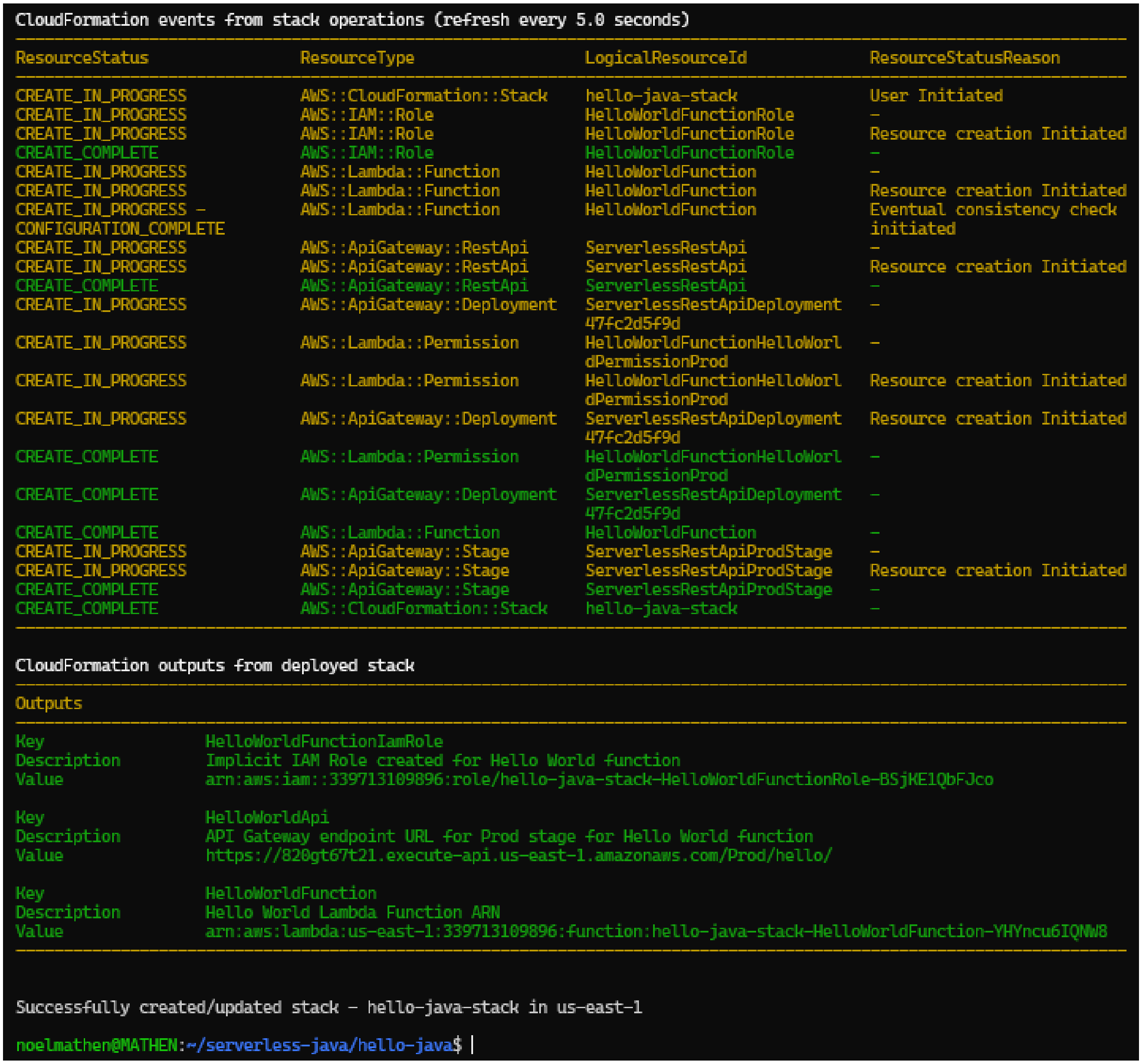
```
noelmathen@MATHEN:~/serverless-java$ curl http://127.0.0.1:3000/hello
{ "message": "hello world", "location": "103.148.21.116" }noelmathen@MATHEN:~/serverless-java$
```



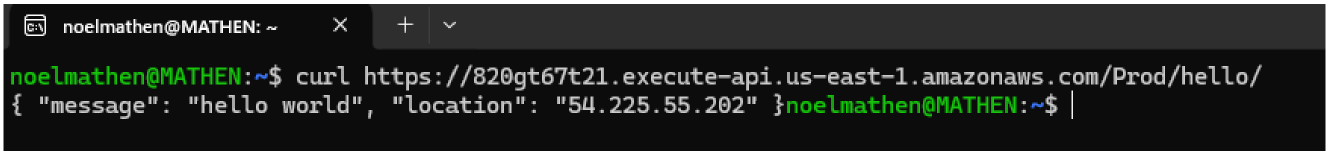
```
noelmathen@MATHEN:~$ curl http://127.0.0.1:3000/hello
{ "message": "hello world", "location": "103.148.21.116" }noelmathen@MATHEN:~$ ^C
noelmathen@MATHEN:~$ sam deploy --guided
Error: Template file not found at /home/noelmathen/template.yml
noelmathen@MATHEN:~$ cd serverless-java/
noelmathen@MATHEN:~/serverless-java$
noelmathen@MATHEN:~/serverless-java$ curl http://127.0.0.1:3000/hello
{ "message": "hello world", "location": "103.148.21.116" }noelmathen@MATHEN:~/serverless-java$ ^C
noelmathen@MATHEN:~/serverless-java$ sam deploy --guided
Error: Template file not found at /home/noelmathen/serverless-java/template.yml
noelmathen@MATHEN:~/serverless-java$ cd hello-java/
noelmathen@MATHEN:~/serverless-java/hello-java$ cd hello-java/
noelmathen@MATHEN:~/serverless-java/hello-java$ curl http://127.0.0.1:3000/hello
{ "message": "hello world", "location": "103.148.21.116" }noelmathen@MATHEN:~/serverless-java/hello-java$ ^C
```



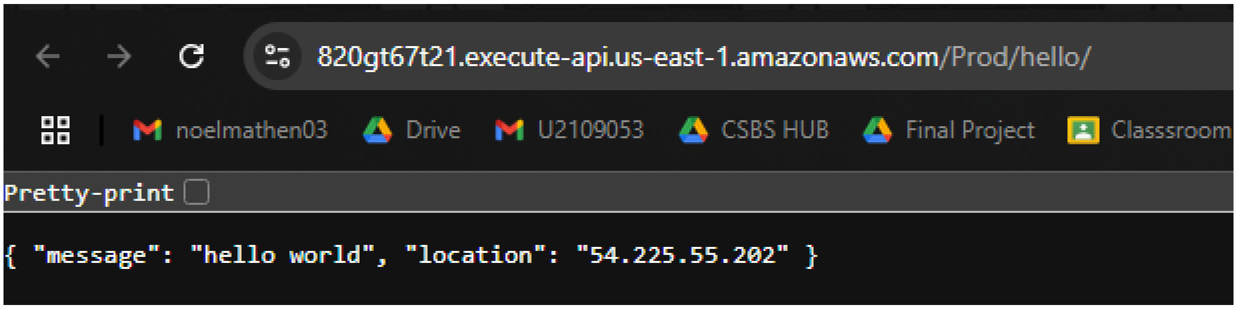
```
noelmathen@MATHEN:~/serverless-java/hello-java$ sam deploy --guided

Configuring SAM deploy
======================

        Looking for config file [samconfig.toml] :  Found
        Reading default arguments  :  Success

        Setting default arguments for 'sam deploy'
        ==========================================
        Stack Name [hello-java]: hello-java-stack
        AWS Region [us-east-1]:
        #Shows you resources changes to be deployed and require a 'Y' to initiate deploy
        Confirm changes before deploy [Y/n]: N
        #SAM needs permission to be able to create roles to connect to the resources in your template
        Allow SAM CLI IAM role creation [Y/n]: Y
        #Preserves the state of previously provisioned resources when an operation fails
        Disable rollback [y/N]: N
        HelloWorldFunction has no authentication. Is this okay? [y/N]: y
        Save arguments to configuration file [Y/n]: Y
        SAM configuration file [samconfig.toml]:
        SAM configuration environment [default]:

        Looking for resources needed for deployment:
        Creating the required resources...
        Successfully created!

        Managed S3 bucket: aws-sam-cli-managed-default-samclisourcebucket-lhg6nwaoqr46
        Auto resolution of buckets can be turned off by setting resolve_s3=False
        To use a specific S3 bucket, set --s3-bucket=<bucket_name>
        Above settings can be stored in samconfig.toml

        Saved arguments to config file
        Running 'sam deploy' for future deployments will use the parameters saved above.
        The above parameters can be changed by modifying samconfig.toml
        Learn more about samconfig.toml syntax at
        https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-cli-config.html

        Uploading to hello-java-stack/6d6a9b7d9d71ae8bd4beaedd14ddde65  874979 / 874979  (100.00%)

        Deploying with following values
        ===============================
        Stack name                   : hello-java-stack
        Region                       : us-east-1
        Confirm changeset            : False
        Disable rollback             : False
        Deployment s3 bucket         : aws-sam-cli-managed-default-samclisourcebucket-lhg6nwaoqr46
        Capabilities                 : ["CAPABILITY_IAM"]
        Parameter overrides          : {}
        Signing Profiles             : {}

Initiating deployment
=====================

        Uploading to hello-java-stack/61fb0a31cc2db93eee4a699bcb5f4d04.template  1383 / 1383  (100.00%)


Waiting for changeset to be created..

CloudFormation stack changeset
--------------------------------------------------------------------------------------------------------------------
Operation                     LogicalResourceId             ResourceType                  Replacement
--------------------------------------------------------------------------------------------------------------------
+ Add                         HelloWorldFunctionHelloWorl   AWS::Lambda::Permission       N/A
                              dPermissionProd
+ Add                         HelloWorldFunctionRole        AWS::IAM::Role                N/A
+ Add                         HelloWorldFunction            AWS::Lambda::Function         N/A
+ Add                         ServerlessRestApiDeployment   AWS::ApiGateway::Deployment   N/A
                              47fc2d5f9d
+ Add                         ServerlessRestApiProdStage    AWS::ApiGateway::Stage        N/A
+ Add                         ServerlessRestApi             AWS::ApiGateway::RestApi      N/A
--------------------------------------------------------------------------------------------------------------------


Changeset created successfully. arn:aws:cloudformation:us-east-1:339713109896:changeSet/samcli-deploy1750685670/c6af0554-8463-40d7-a1d6-388cc85891de


2025-06-23 13:34:39 - Waiting for stack create/update to complete
```



```
CloudFormation events from stack operations (refresh every 5.0 seconds)
-------------------------------------------------------------------------------------------------------------------------
ResourceStatus                ResourceType                  LogicalResourceId             ResourceStatusReason
-------------------------------------------------------------------------------------------------------------------------
CREATE_IN_PROGRESS            AWS::CloudFormation::Stack    hello-java-stack              User Initiated
CREATE_IN_PROGRESS            AWS::IAM::Role                HelloWorldFunctionRole        -
CREATE_IN_PROGRESS            AWS::IAM::Role                HelloWorldFunctionRole        Resource creation Initiated
CREATE_COMPLETE               AWS::IAM::Role                HelloWorldFunctionRole        -
CREATE_IN_PROGRESS            AWS::Lambda::Function         HelloWorldFunction            -
CREATE_IN_PROGRESS            AWS::Lambda::Function         HelloWorldFunction            Resource creation Initiated
CREATE_IN_PROGRESS -          AWS::Lambda::Function         HelloWorldFunction            Eventual consistency check
CONFIGURATION_COMPLETE                                                                    initiated
CREATE_IN_PROGRESS            AWS::ApiGateway::RestApi      ServerlessRestApi             -
CREATE_IN_PROGRESS            AWS::ApiGateway::RestApi      ServerlessRestApi             Resource creation Initiated
CREATE_COMPLETE               AWS::ApiGateway::RestApi      ServerlessRestApi             -
CREATE_IN_PROGRESS            AWS::ApiGateway::Deployment   ServerlessRestApiDeployment   -
                                                            47fc2d5f9d
CREATE_IN_PROGRESS            AWS::Lambda::Permission       HelloWorldFunctionHelloWorl   -
                                                            dPermissionProd
CREATE_IN_PROGRESS            AWS::Lambda::Permission       HelloWorldFunctionHelloWorl   Resource creation Initiated
                                                            dPermissionProd
CREATE_IN_PROGRESS            AWS::ApiGateway::Deployment   ServerlessRestApiDeployment   Resource creation Initiated
                                                            47fc2d5f9d
CREATE_COMPLETE               AWS::Lambda::Permission       HelloWorldFunctionHelloWorl   -
                                                            dPermissionProd
CREATE_COMPLETE               AWS::ApiGateway::Deployment   ServerlessRestApiDeployment   -
                                                            47fc2d5f9d
CREATE_COMPLETE               AWS::Lambda::Function         HelloWorldFunction            -
CREATE_IN_PROGRESS            AWS::ApiGateway::Stage        ServerlessRestApiProdStage    -
CREATE_IN_PROGRESS            AWS::ApiGateway::Stage        ServerlessRestApiProdStage    Resource creation Initiated
CREATE_COMPLETE               AWS::ApiGateway::Stage        ServerlessRestApiProdStage    -
CREATE_COMPLETE               AWS::CloudFormation::Stack    hello-java-stack              -
-------------------------------------------------------------------------------------------------------------------------

CloudFormation outputs from deployed stack
-------------------------------------------------------------------------------------------------------------------------
Outputs
-------------------------------------------------------------------------------------------------------------------------
Key                 HelloWorldFunctionIamRole
Description         Implicit IAM Role created for Hello World function
Value               arn:aws:iam::339713109896:role/hello-java-stack-HelloWorldFunctionRole-BSjKE1QbFJco

Key                 HelloWorldApi
Description         API Gateway endpoint URL for Prod stage for Hello World function
Value               https://820gt67t21.execute-api.us-east-1.amazonaws.com/Prod/hello/

Key                 HelloWorldFunction
Description         Hello World Lambda Function ARN
Value               arn:aws:lambda:us-east-1:339713109896:function:hello-java-stack-HelloWorldFunction-YHYncu6IQNW8
-------------------------------------------------------------------------------------------------------------------------


Successfully created/updated stack - hello-java-stack in us-east-1

noelmathen@MATHEN:~/serverless-java/hello-java$
```

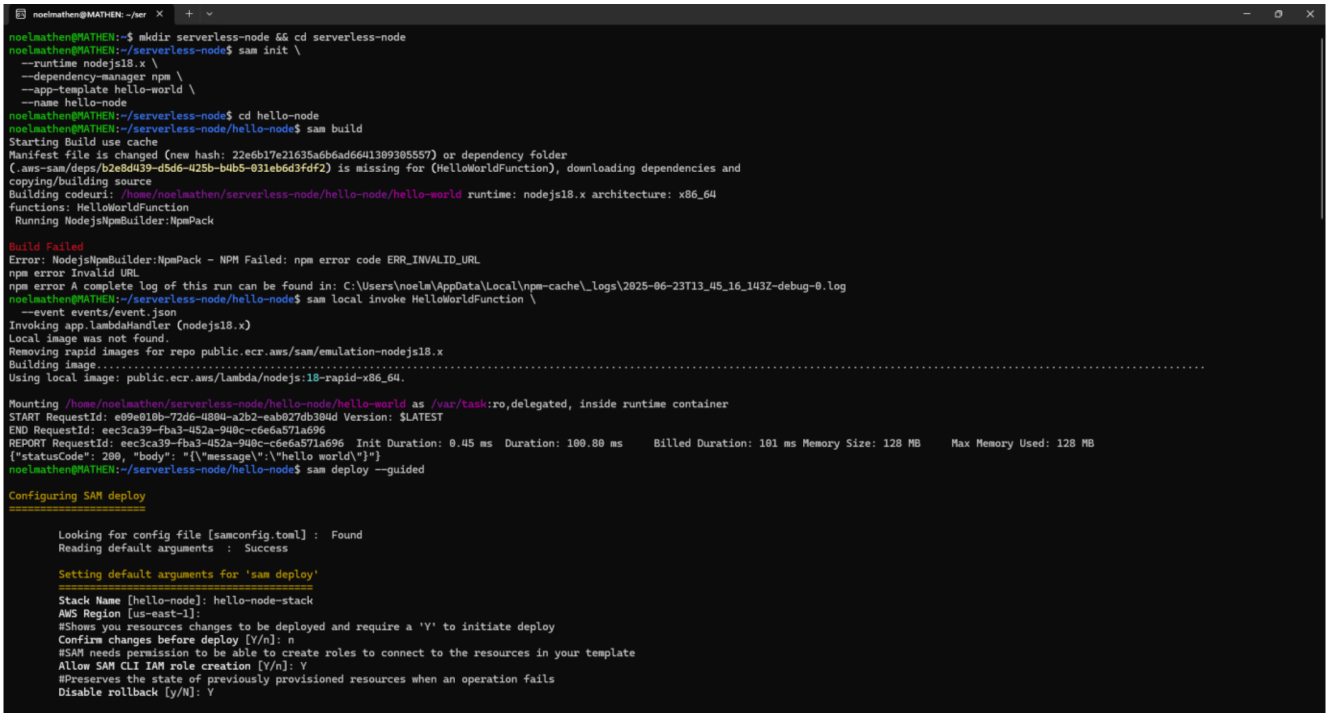


```
noelmathen@MATHEN:~$ curl https://820gt67t21.execute-api.us-east-1.amazonaws.com/Prod/hello/
{ "message": "hello world", "location": "54.225.55.202" }noelmathen@MATHEN:~$
```

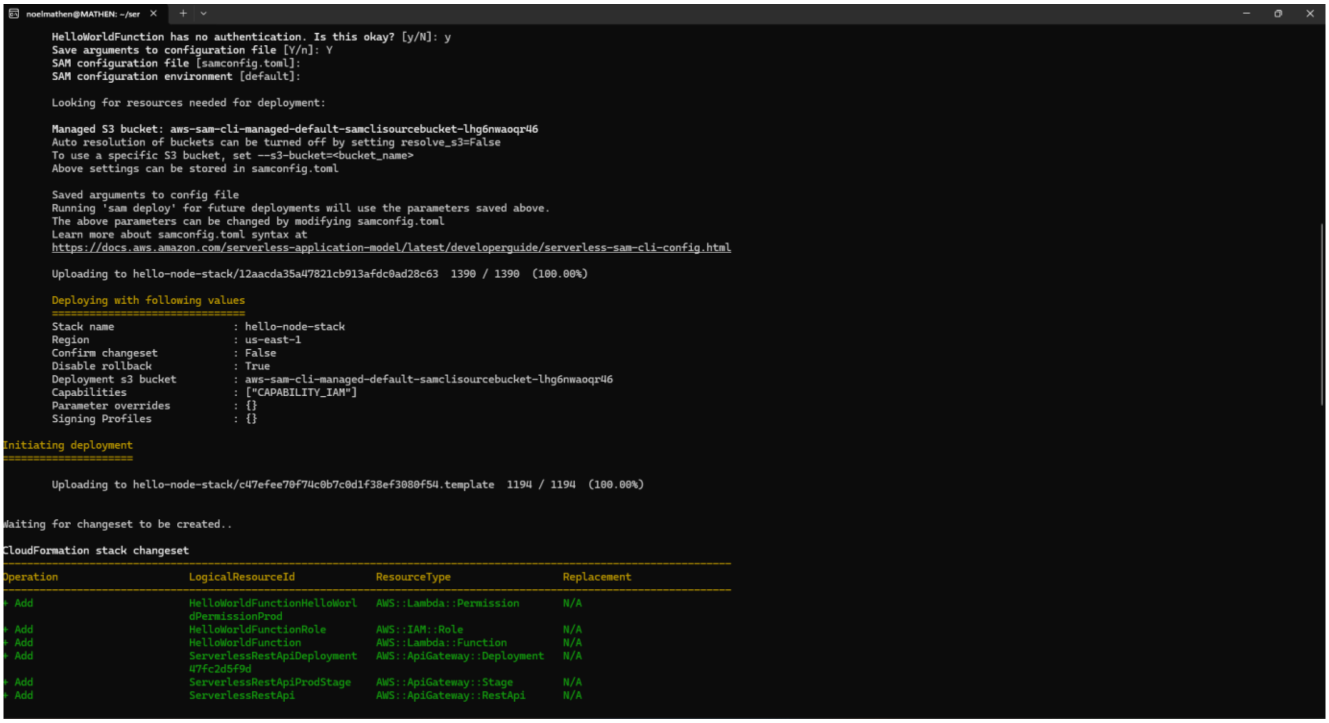


```
{ "message": "hello world", "location": "54.225.55.202" }
```

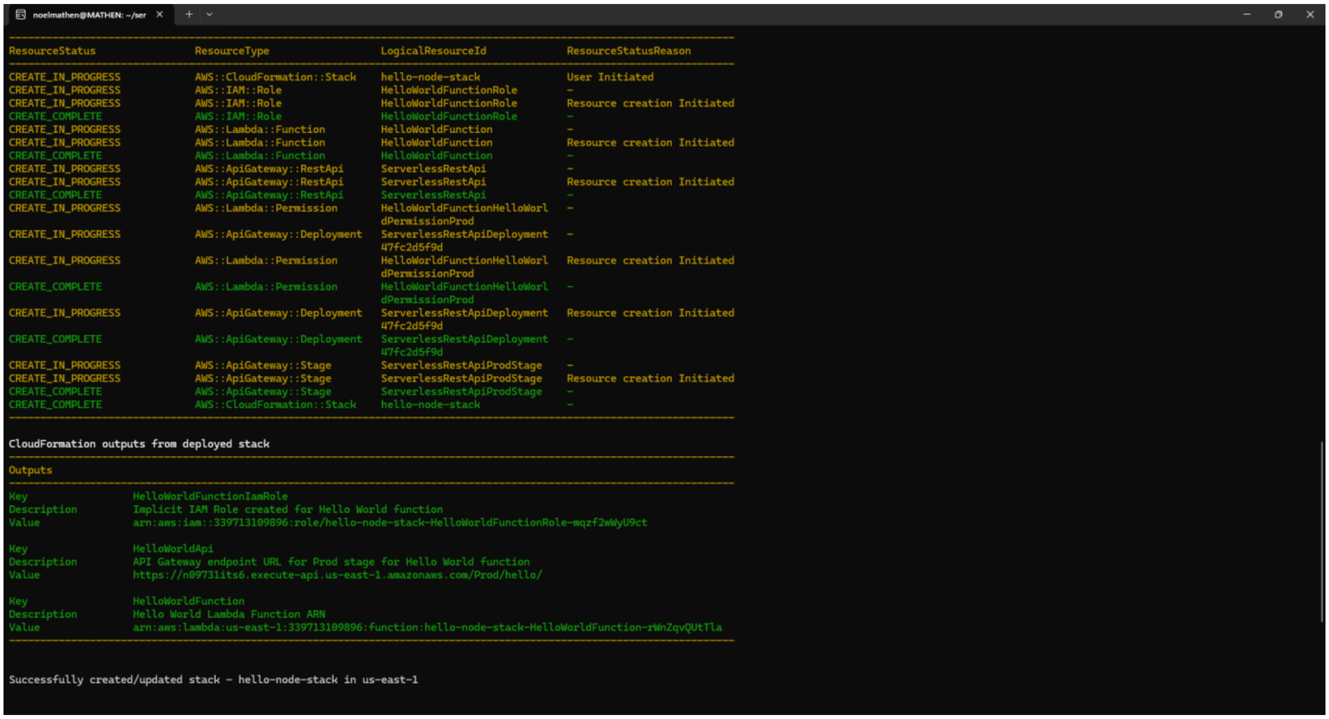
3. “Hello World” Serverless in NodeJS



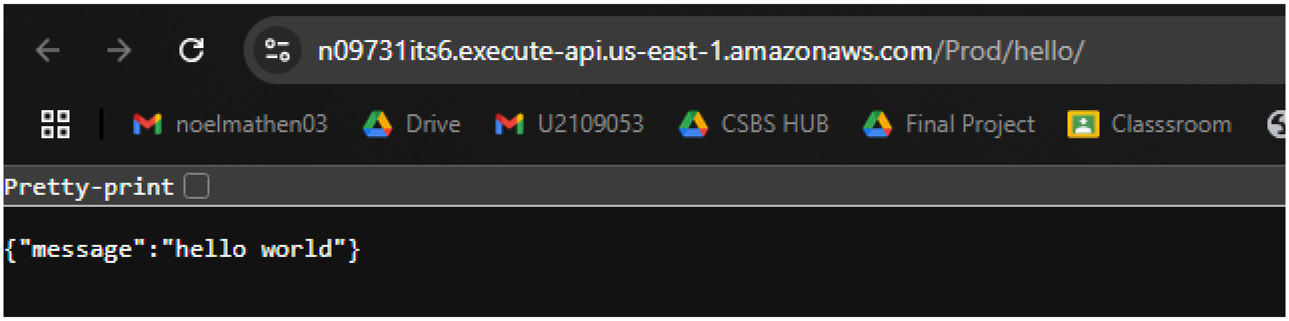
```
noelmathen@MATHEN:~$ mkdir serverless-node && cd serverless-node
noelmathen@MATHEN:~/serverless-node$ sam init \
  --runtime nodejs18.x \
  --dependency-manager npm \
  --app-template hello-world \
  --name hello-node
noelmathen@MATHEN:~/serverless-node$ cd hello-node
noelmathen@MATHEN:~/serverless-node/hello-node$ sam build
Starting Build use cache
Manifest file is changed (new hash: 22e6b17e21635a6b6ad6641309305557) or dependency folder
(.aws-sam/deps/b2e8d439-d5d6-425b-b4b5-031eb6d3fdf2) is missing for (HelloWorldFunction), downloading dependencies and
copying/building source
Building codeuri: /home/noelmathen/serverless-node/hello-node/hello-world runtime: nodejs18.x architecture: x86_64
functions: HelloWorldFunction
 Running NodejsNpmBuilder:NpmPack

Build Failed
Error: NodejsNpmBuilder:NpmPack - NPM Failed: npm error code ERR_INVALID_URL
npm error Invalid URL
npm error A complete log of this run can be found in: C:\Users\noelm\AppData\Local\npm-cache\_logs\2025-06-23T13_45_16_143Z-debug-0.log
noelmathen@MATHEN:~/serverless-node/hello-node$ sam local invoke HelloWorldFunction \
  --event events/event.json
Invoking app.lambdaHandler (nodejs18.x)
Local image was not found.
Removing rapid images for repo public.ecr.aws/sam/emulation-nodejs18.x
Building image.................................................................................................................................
Using local image: public.ecr.aws/lambda/nodejs:18-rapid-x86_64.

Mounting /home/noelmathen/serverless-node/hello-node/hello-world as /var/task:ro,delegated, inside runtime container
START RequestId: e09e010b-72d6-4804-a2b2-eab027db304d Version: $LATEST
END RequestId: eec3ca39-fba3-452a-940c-c6e6a571a696
REPORT RequestId: eec3ca39-fba3-452a-940c-c6e6a571a696  Init Duration: 0.45 ms  Duration: 100.80 ms     Billed Duration: 101 ms Memory Size: 128 MB     Max Memory Used: 128 MB
{"statusCode": 200, "body": "{\"message\":\"hello world\"}"}
noelmathen@MATHEN:~/serverless-node/hello-node$ sam deploy --guided

Configuring SAM deploy
======================

        Looking for config file [samconfig.toml] :  Found
        Reading default arguments  :  Success

        Setting default arguments for 'sam deploy'
        =========================================
        Stack Name [hello-node]: hello-node-stack
        AWS Region [us-east-1]:
        #Shows you resources changes to be deployed and require a 'Y' to initiate deploy
        Confirm changes before deploy [Y/n]: n
        #SAM needs permission to be able to create roles to connect to the resources in your template
        Allow SAM CLI IAM role creation [Y/n]: Y
        #Preserves the state of previously provisioned resources when an operation fails
        Disable rollback [y/N]: Y
```



```
        HelloWorldFunction has no authentication. Is this okay? [y/N]: y
        Save arguments to configuration file [Y/n]: Y
        SAM configuration file [samconfig.toml]:
        SAM configuration environment [default]:

        Looking for resources needed for deployment:

        Managed S3 bucket: aws-sam-cli-managed-default-samclisourcebucket-lhg6nwaoqr46
        Auto resolution of buckets can be turned off by setting resolve_s3=False
        To use a specific S3 bucket, set --s3-bucket=<bucket_name>
        Above settings can be stored in samconfig.toml

        Saved arguments to config file
        Running 'sam deploy' for future deployments will use the parameters saved above.
        The above parameters can be changed by modifying samconfig.toml
        Learn more about samconfig.toml syntax at
        https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-cli-config.html

        Uploading to hello-node-stack/12aacda35a47821cb913afdc0ad28c63  1390 / 1390  (100.00%)

        Deploying with following values
        ===============================
        Stack name                   : hello-node-stack
        Region                       : us-east-1
        Confirm changeset            : False
        Disable rollback             : True
        Deployment s3 bucket         : aws-sam-cli-managed-default-samclisourcebucket-lhg6nwaoqr46
        Capabilities                 : ["CAPABILITY_IAM"]
        Parameter overrides          : {}
        Signing Profiles             : {}

Initiating deployment
=====================

        Uploading to hello-node-stack/c47efee70f74c0b7c0d1f38ef3080f54.template  1194 / 1194  (100.00%)


Waiting for changeset to be created..

CloudFormation stack changeset
-------------------------------------------------------------------------------------------------------------------------
Operation                    LogicalResourceId            ResourceType                 Replacement
-------------------------------------------------------------------------------------------------------------------------
+ Add                        HelloWorldFunctionHelloWorl  AWS::Lambda::Permission      N/A
                             dPermissionProd
+ Add                        HelloWorldFunctionRole       AWS::IAM::Role               N/A
+ Add                        HelloWorldFunction           AWS::Lambda::Function        N/A
+ Add                        ServerlessRestApiDeployment  AWS::ApiGateway::Deployment  N/A
                             47fc2d5f9d
+ Add                        ServerlessRestApiProdStage   AWS::ApiGateway::Stage       N/A
+ Add                        ServerlessRestApi            AWS::ApiGateway::RestApi     N/A
```



```
-------------------------------------------------------------------------------------------------
ResourceStatus           ResourceType             LogicalResourceId        ResourceStatusReason
-------------------------------------------------------------------------------------------------
CREATE_IN_PROGRESS       AWS::CloudFormation::Stack hello-node-stack       User Initiated
CREATE_IN_PROGRESS       AWS::IAM::Role           HelloWorldFunctionRole   -
CREATE_IN_PROGRESS       AWS::IAM::Role           HelloWorldFunctionRole   Resource creation Initiated
CREATE_COMPLETE          AWS::IAM::Role           HelloWorldFunctionRole   -
CREATE_IN_PROGRESS       AWS::Lambda::Function    HelloWorldFunction       -
CREATE_IN_PROGRESS       AWS::Lambda::Function    HelloWorldFunction       Resource creation Initiated
CREATE_COMPLETE          AWS::Lambda::Function    HelloWorldFunction       -
CREATE_IN_PROGRESS       AWS::ApiGateway::RestApi ServerlessRestApi        -
CREATE_IN_PROGRESS       AWS::ApiGateway::RestApi ServerlessRestApi        Resource creation Initiated
CREATE_COMPLETE          AWS::ApiGateway::RestApi ServerlessRestApi        -
CREATE_IN_PROGRESS       AWS::Lambda::Permission  HelloWorldFunctionHelloWorl -
                                                  dPermissionProd
CREATE_IN_PROGRESS       AWS::ApiGateway::Deployment ServerlessRestApiDeployment -
                                                  47fc2d5f9d
CREATE_IN_PROGRESS       AWS::Lambda::Permission  HelloWorldFunctionHelloWorl Resource creation Initiated
                                                  dPermissionProd
CREATE_COMPLETE          AWS::Lambda::Permission  HelloWorldFunctionHelloWorl -
                                                  dPermissionProd
CREATE_IN_PROGRESS       AWS::ApiGateway::Deployment ServerlessRestApiDeployment Resource creation Initiated
                                                  47fc2d5f9d
CREATE_COMPLETE          AWS::ApiGateway::Deployment ServerlessRestApiDeployment -
                                                  47fc2d5f9d
CREATE_IN_PROGRESS       AWS::ApiGateway::Stage   ServerlessRestApiProdStage -
CREATE_IN_PROGRESS       AWS::ApiGateway::Stage   ServerlessRestApiProdStage Resource creation Initiated
CREATE_COMPLETE          AWS::ApiGateway::Stage   ServerlessRestApiProdStage -
CREATE_COMPLETE          AWS::CloudFormation::Stack hello-node-stack       -
-------------------------------------------------------------------------------------------------

CloudFormation outputs from deployed stack
-------------------------------------------------------------------------------------------------
Outputs
-------------------------------------------------------------------------------------------------
Key                 HelloWorldFunctionIamRole
Description         Implicit IAM Role created for Hello World function
Value               arn:aws:iam::339713109896:role/hello-node-stack-HelloWorldFunctionRole-mqzf2wWyU9ct

Key                 HelloWorldApi
Description         API Gateway endpoint URL for Prod stage for Hello World function
Value               https://n09731its6.execute-api.us-east-1.amazonaws.com/Prod/hello/

Key                 HelloWorldFunction
Description         Hello World Lambda Function ARN
Value               arn:aws:lambda:us-east-1:339713109896:function:hello-node-stack-HelloWorldFunction-rWnZqvQUtTla
-------------------------------------------------------------------------------------------------


Successfully created/updated stack - hello-node-stack in us-east-1
```

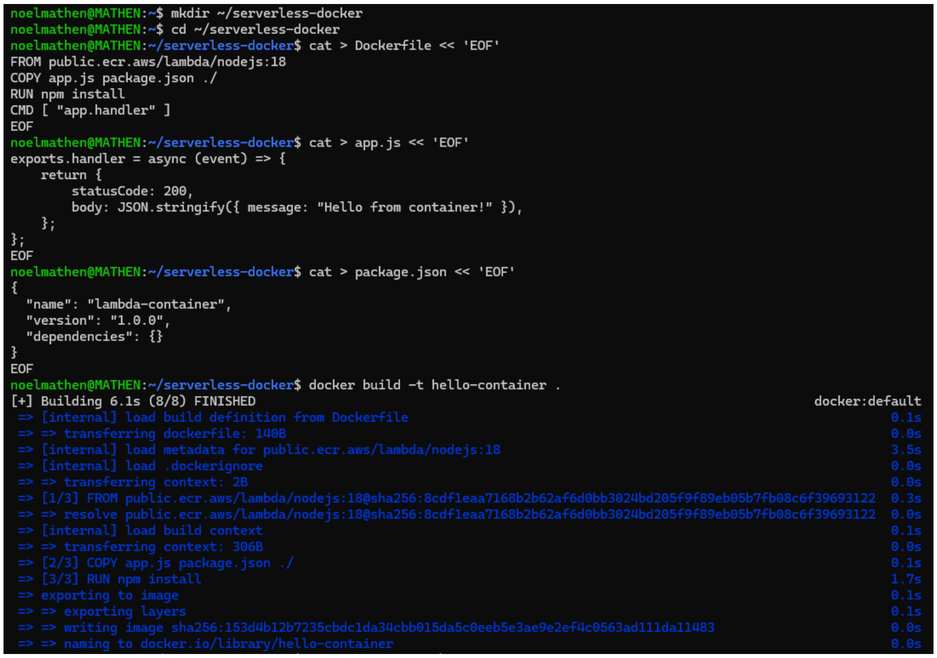


n09731its6.execute-api.us-east-1.amazonaws.com/Prod/hello/

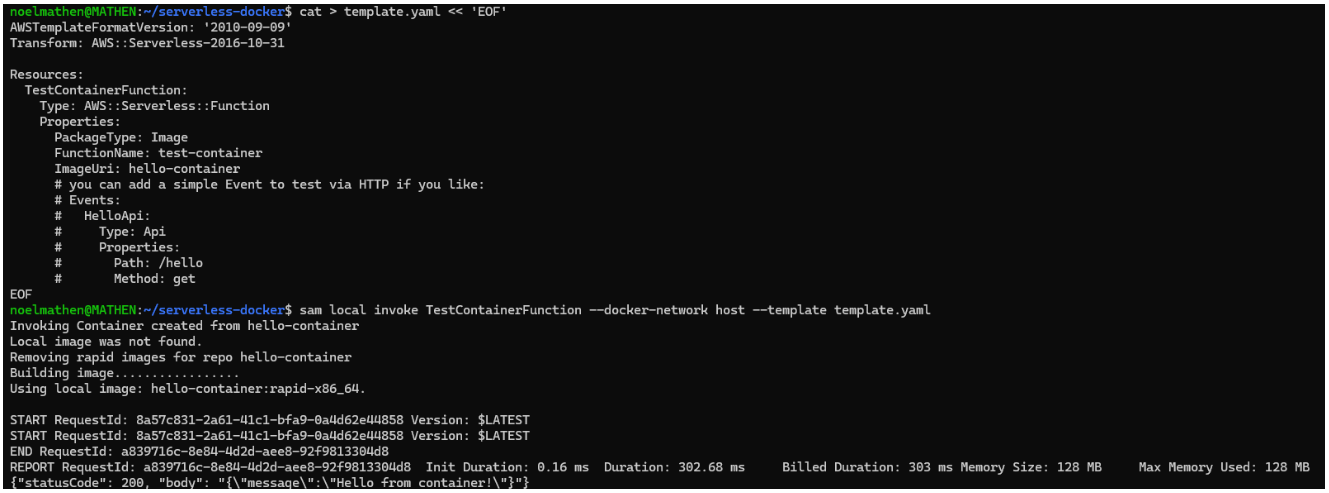
Pretty-print □

```
{"message":"hello world"}
```

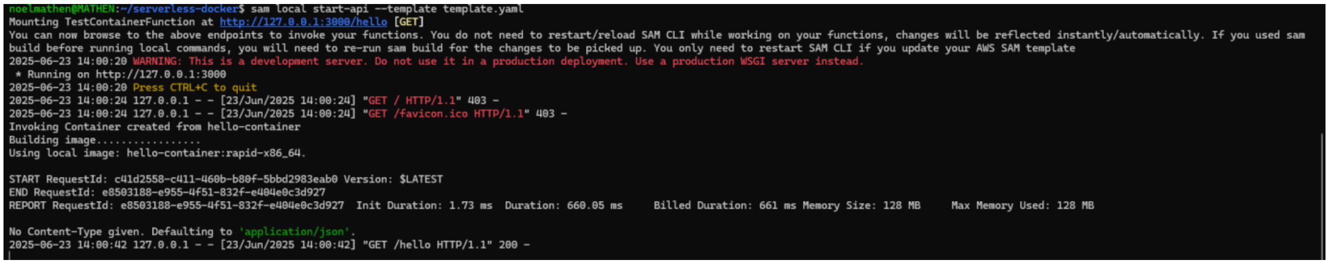
4. Serverless via Docker Container Images



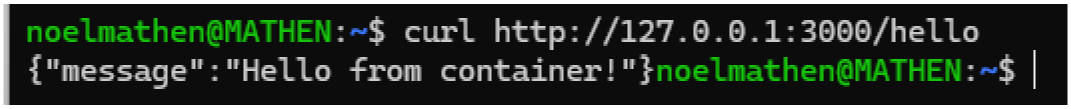
```
noelmathen@MATHEN:~$ mkdir ~/serverless-docker
noelmathen@MATHEN:~$ cd ~/serverless-docker
noelmathen@MATHEN:~/serverless-docker$ cat > Dockerfile << 'EOF'
FROM public.ecr.aws/lambda/nodejs:18
COPY app.js package.json ./
RUN npm install
CMD [ "app.handler" ]
EOF
noelmathen@MATHEN:~/serverless-docker$ cat > app.js << 'EOF'
exports.handler = async (event) => {
    return {
        statusCode: 200,
        body: JSON.stringify({ message: "Hello from container!" }),
    };
};
EOF
noelmathen@MATHEN:~/serverless-docker$ cat > package.json << 'EOF'
{
  "name": "lambda-container",
  "version": "1.0.0",
  "dependencies": {}
}
EOF
noelmathen@MATHEN:~/serverless-docker$ docker build -t hello-container .
[+] Building 6.1s (8/8) FINISHED                                                              docker:default
 => [internal] load build definition from Dockerfile                                                    0.1s
 => => transferring dockerfile: 140B                                                                    0.0s
 => [internal] load metadata for public.ecr.aws/lambda/nodejs:18                                        3.5s
 => [internal] load .dockerignore                                                                       0.0s
 => => transferring context: 2B                                                                         0.0s
 => [1/3] FROM public.ecr.aws/lambda/nodejs:18@sha256:8cdf1eaa7168b2b62af6d0bb3024bd205f9f89eb05b7fb08c6f39693122  0.3s
 => => resolve public.ecr.aws/lambda/nodejs:18@sha256:8cdf1eaa7168b2b62af6d0bb3024bd205f9f89eb05b7fb08c6f39693122  0.0s
 => [internal] load build context                                                                       0.1s
 => => transferring context: 306B                                                                       0.0s
 => [2/3] COPY app.js package.json ./                                                                   0.1s
 => [3/3] RUN npm install                                                                               1.7s
 => exporting to image                                                                                  0.1s
 => => exporting layers                                                                                 0.1s
 => => writing image sha256:153d4b12b7235cbdc1da34cbb015da5c0eeb5e3ae9e2ef4c0563ad111da11483            0.0s
 => => naming to docker.io/library/hello-container                                                      0.0s
```



```
noelmathen@MATHEN:~/serverless-docker$ cat > template.yaml << 'EOF'
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31

Resources:
  TestContainerFunction:
    Type: AWS::Serverless::Function
    Properties:
      PackageType: Image
      FunctionName: test-container
      ImageUri: hello-container
      # you can add a simple Event to test via HTTP if you like:
      # Events:
      #   HelloApi:
      #     Type: Api
      #     Properties:
      #       Path: /hello
      #       Method: get
EOF
noelmathen@MATHEN:~/serverless-docker$ sam local invoke TestContainerFunction --docker-network host --template template.yaml
Invoking Container created from hello-container
Local image was not found.
Removing rapid images for repo hello-container
Building image.................
Using local image: hello-container:rapid-x86_64.

START RequestId: 8a57c831-2a61-41c1-bfa9-0a4d62e44858 Version: $LATEST
START RequestId: 8a57c831-2a61-41c1-bfa9-0a4d62e44858 Version: $LATEST
END RequestId: a839716c-8e84-4d2d-aee8-92f9813304d8
REPORT RequestId: a839716c-8e84-4d2d-aee8-92f9813304d8  Init Duration: 0.16 ms  Duration: 302.68 ms     Billed Duration: 303 ms Memory Size: 128 MB     Max Memory Used: 128 MB
{"statusCode": 200, "body": "{\"message\":\"Hello from container!\"}"}
```



```
noelmathen@MATHEN:~/serverless-docker$ sam local start-api --template template.yaml
Mounting TestContainerFunction at http://127.0.0.1:3000/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. If you used sam
build before running local commands, you will need to re-run sam build for the changes to be picked up. You only need to restart SAM CLI if you update your AWS SAM template
2025-06-23 14:00:20 WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:3000
2025-06-23 14:00:20 Press CTRL+C to quit
2025-06-23 14:00:24 127.0.0.1 - - [23/Jun/2025 14:00:24] "GET / HTTP/1.1" 403 -
2025-06-23 14:00:24 127.0.0.1 - - [23/Jun/2025 14:00:24] "GET /favicon.ico HTTP/1.1" 403 -
Invoking Container created from hello-container
Building image.................
Using local image: hello-container:rapid-x86_64.

START RequestId: c41d2558-c411-460b-b80f-5bbd2983eab0 Version: $LATEST
END RequestId: e8503188-e955-4f51-832f-e404e0c3d927
REPORT RequestId: e8503188-e955-4f51-832f-e404e0c3d927  Init Duration: 1.73 ms  Duration: 660.05 ms     Billed Duration: 661 ms Memory Size: 128 MB     Max Memory Used: 128 MB

No Content-Type given. Defaulting to 'application/json'.
2025-06-23 14:00:42 127.0.0.1 - - [23/Jun/2025 14:00:42] "GET /hello HTTP/1.1" 200 -
```



```
noelmathen@MATHEN:~$ curl http://127.0.0.1:3000/hello
{"message":"Hello from container!"}noelmathen@MATHEN:~$
```