

APPLICATION DEVELOPMENT ON CLOUD

NAME: S SAI SRAVAN

ID-NO:2000031350

LAB-9: CLOUD FORMATION

→Cloud Formation to create EC2 instance

Resources:

MyInstance:

Type: AWS::EC2::Instance

Properties:

AvailabilityZone: us-east-2a

ImageId: ami-02d1e544b84bf7502

InstanceType: t2.micro

The screenshot shows the AWS Management Console CloudFormation wizard. The current step is 'Prerequisite - Prepare template'. It displays three options: 'Template is ready' (selected), 'Use a sample template', and 'Create template in Designer'. Below this, the 'Specify template' step is shown, which allows uploading a template file. A file named 'ec2template.yaml' is selected. The S3 URL for the uploaded file is provided as <https://s3.us-east-2.amazonaws.com/cf-templates-8rlesg6i7cu0-us-east-2/20221761kq-ec2template.yaml>. The bottom of the screen shows the standard AWS navigation bar.

Week 1 Interview Questions | HackerRank AWS Management Console CloudFormation - Stack

Search for services, features, blogs, docs, and more [Alt+S]

CloudFormation > Stacks > Create stack

Step 1 Specify template

Step 2 Specify stack details

Step 3 Configure stack options

Step 4 Review

Specify stack details

Stack name

Stack name: EC2Instance-June25

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

No parameters

There are no parameters defined in your template

Cancel Previous Next

This screenshot shows the 'Specify stack details' step of a CloudFormation stack creation. It includes fields for the stack name ('EC2Instance-June25') and parameters, which are currently empty. Navigation buttons for 'Cancel', 'Previous', and 'Next' are at the bottom.

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences

ENG IN 09:53 25-06-2022

Week 1 Interview Questions | HackerRank AWS Management Console CloudFormation - Stack

Search for services, features, blogs, docs, and more [Alt+S]

CloudFormation - Stack

Stack failure options

Behavior on provisioning failure

Specify the roll back behavior for a stack failure. Learn more

Roll back all stack resources

Roll back the stack to the last known stable state.

Preserve successfully provisioned resources

Preserves the state of successfully provisioned resources, while rolling back failed resources to the last known stable state. Resources without a last known stable state will be deleted upon the next stack operation.

Advanced options

You can set additional options for your stack, like notification options and a stack policy. Learn more

► **Stack policy**

Defines the resources that you want to protect from unintentional updates during a stack update.

► **Rollback configuration**

Specify alarms for CloudFormation to monitor when creating and updating the stack. If the operation breaches an alarm threshold, CloudFormation rolls it back. Learn more

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences

ENG IN 09:54 25-06-2022

This screenshot displays the 'Stack failure options' and 'Advanced options' steps of the CloudFormation stack creation process. It includes sections for specifying behavior on provisioning failure (with 'Roll back all stack resources' selected) and setting advanced options like stack policies and rollback configurations. The interface is consistent with the first screenshot, showing the same navigation and status bar.

The screenshot shows the AWS CloudFormation console with the stack 'EC2Instance-June25' selected. The 'Events' tab is active, displaying a single event: 'CREATE_IN_PROGRESS' for the logical ID 'EC2Instance-June25' at 2022-06-25 09:55:13 UTC+0530, with the status reason 'User Initiated'. The browser's address bar shows the URL: us-east-2.console.aws.amazon.com/cloudformation/home?region=us-east-2#/stacks/events?stackId=arn%3aws%3Acloudformation%3Aus-east-2... .

Stack got created and successfully created EC2 instance through cloud formation

→Cloud formation to create lambda function

Template code:

```
{
  "Resources": {
    "HelloLambdaRole": {
      "Type": "AWS::IAM::Role",
      "Properties": {
        "RoleName": "HelloLambdaRole",
        "AssumeRolePolicyDocument": {
          "Statement": [
            {
              "Effect": "Allow",
              "Principal": {
                "Service": "lambda.amazonaws.com"
              },
              "Action": "sts:AssumeRole"
            }
          ]
        }
      }
    }
  }
}
```

```
    }
}

}

"HelloLambdaFunction": {

  "Type": "AWS::Lambda::Function",

  "Properties": {

    "FunctionName": "HelloLambdaFunction",

    "Role": {

      "Fn::GetAtt": ["HelloLambdaRole", "Arn"]

    },

    "Runtime": "python3.7",

    "Handler": "index.my_handler",

    "Code": {

      "ZipFile": "def my_handler(event, context):\n  message = \"Hello Lambda World!\"\n  return\n  message\n"

    }

  }

}

}
```

The screenshot shows the AWS CloudFormation console with the 'Events' tab selected for the stack 'cloudformlambda-june29'. The event details are as follows:

| Timestamp | Logical ID | Status | Status reason |
|------------------------------|------------------------|--------------------|----------------|
| 2022-06-29 09:46:18 UTC+0530 | cloudformlambda-june29 | CREATE_IN_PROGRESS | User Initiated |

The screenshot shows the AWS Lambda console with the 'Functions' tab selected. The function details are as follows:

| Function name | Description | Package type | Runtime | Last modified |
|---------------------|-------------|--------------|------------|----------------|
| HelloLambdaFunction | - | Zip | Python 3.7 | 39 seconds ago |

