

Lab Assignment – 05

Introduction to AWS Step Function

Serverless Microservice Orchestration

Task – 1: Create Role for AWS Lambda Function.

The screenshot shows the AWS IAM Management Console in a web browser. The URL is [https://console.aws.amazon.com/iam/home?region=ap-south-1#/roles\\$new?step=review&commonUseCase=Lambda%2BLambda...](https://console.aws.amazon.com/iam/home?region=ap-south-1#/roles$new?step=review&commonUseCase=Lambda%2BLambda...). The page is titled 'Review' and instructs the user to provide required information before creating the role.

Role name*

Use alphanumeric and '+', '=', '@', '_', '-' characters. Maximum 64 characters.

Role description

Maximum 1000 characters. Use alphanumeric and '+', '=', '@', '_', '-' characters.

Trusted entities AWS service: [lambda.amazonaws.com](#)

Policies

- [AWSLambdaBasicExecutionRole](#)
- [CloudWatchFullAccess](#)

Permissions boundary Permissions boundary is not set

No tags were added.

* Required

[Cancel](#) [Previous](#) [Create role](#)

Feedback English (US) © 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Task – 2: Create role for AWS Step Function.

The screenshot shows the AWS IAM Management Console in a web browser. The URL is [https://console.aws.amazon.com/iam/home?region=ap-south-1#/roles\\$new?step=review&selectedService=StepFunctions&select...](https://console.aws.amazon.com/iam/home?region=ap-south-1#/roles$new?step=review&selectedService=StepFunctions&select...). The page is titled 'Create role' and 'Review'. There are four numbered steps in the top right, with step 4 being the current one.

Role name*

Use alphanumeric and '+', '=', '@', '_', '-' characters. Maximum 64 characters.

Role description

Maximum 1000 characters. Use alphanumeric and '+', '=', '@', '_', '-' characters.

Trusted entities AWS service: [states.amazonaws.com](#)

Policies

- [AWSLambdaRole](#)

Permissions boundary Permissions boundary is not set

No tags were added.

* Required

[Cancel](#) [Previous](#) [Create role](#)

Feedback English (US) © 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Task – 3: Create 2 lambda functions and by using AWS Step Functions make them work together by writing some code.

Function name
Enter a name that describes the purpose of your function.

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime [Info](#)
Choose the language to use to write your function.

Permissions [Info](#)
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

▼ **Change default execution role**

Execution role
Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#).

- ☐ Create a new role with basic Lambda permissions
- ☒ Use an existing role

▼ **Change default execution role**

Execution role
Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#).

- ☐ Create a new role with basic Lambda permissions
- ☒ Use an existing role
- ☐ Create a new role from AWS policy templates

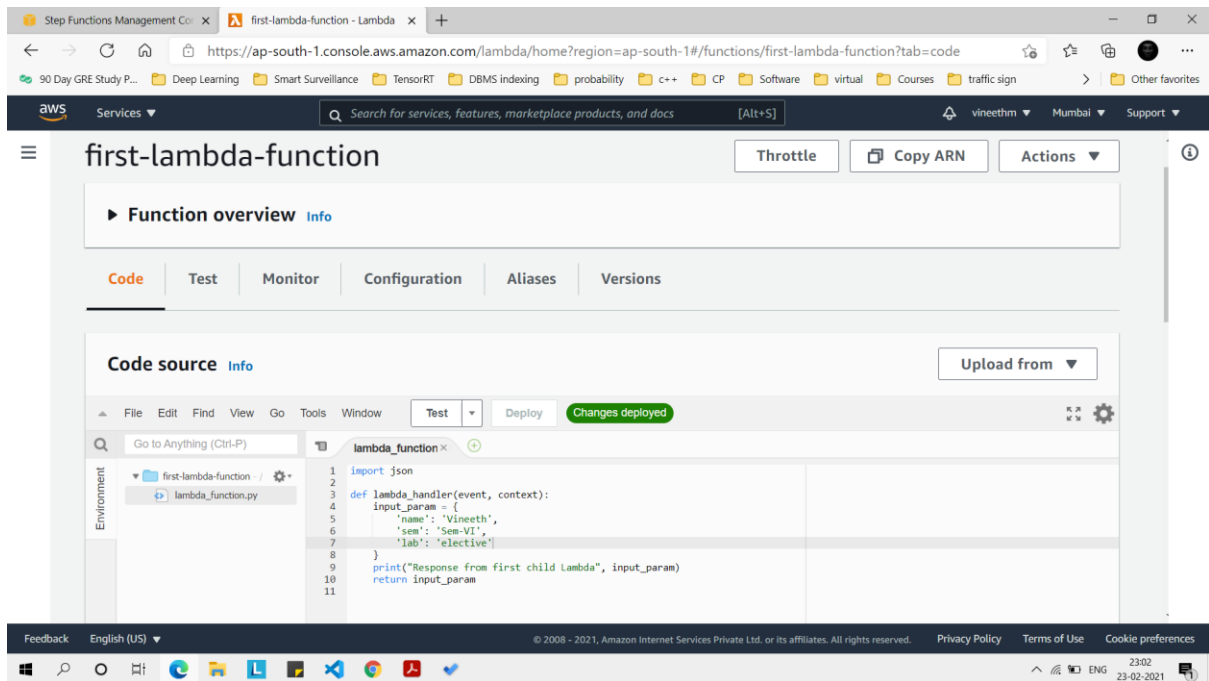
Existing role
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.

[View the LambdaStepFunction role on the IAM console.](#)

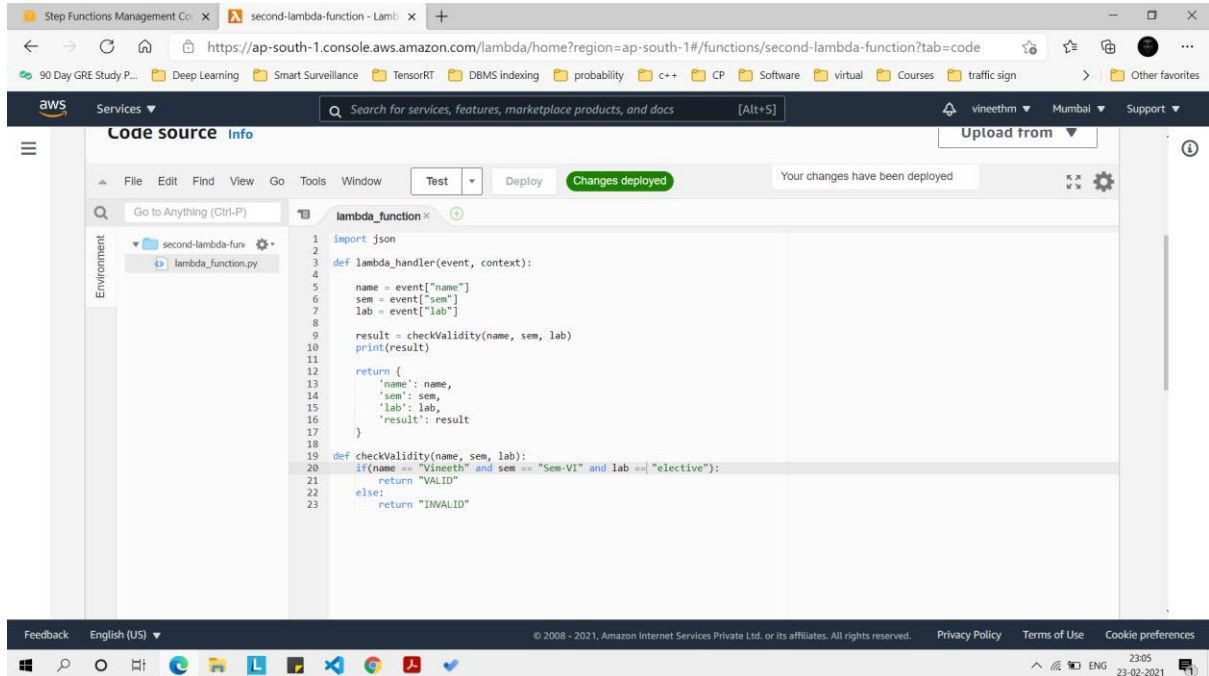
► **Advanced settings**

[Cancel](#) [Create function](#)

Lambda Function-1: Code

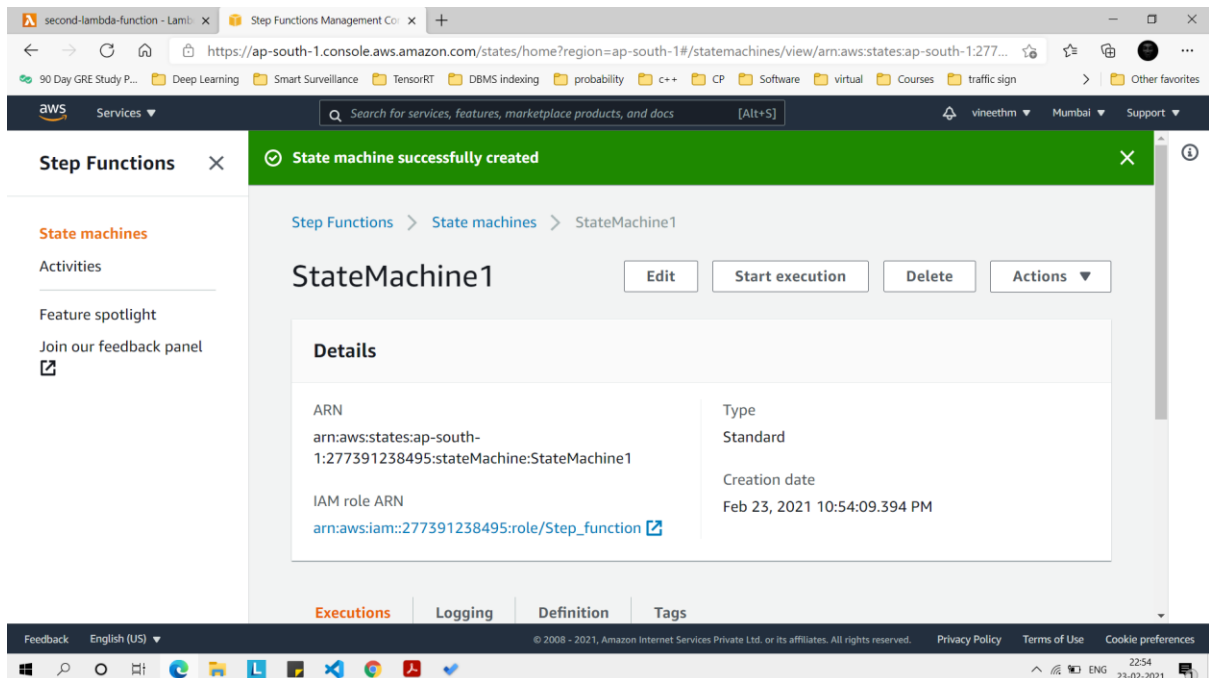


Lambda Function-2: Code



Task – 4: Create AWS Step Function:

Create a State Machine.

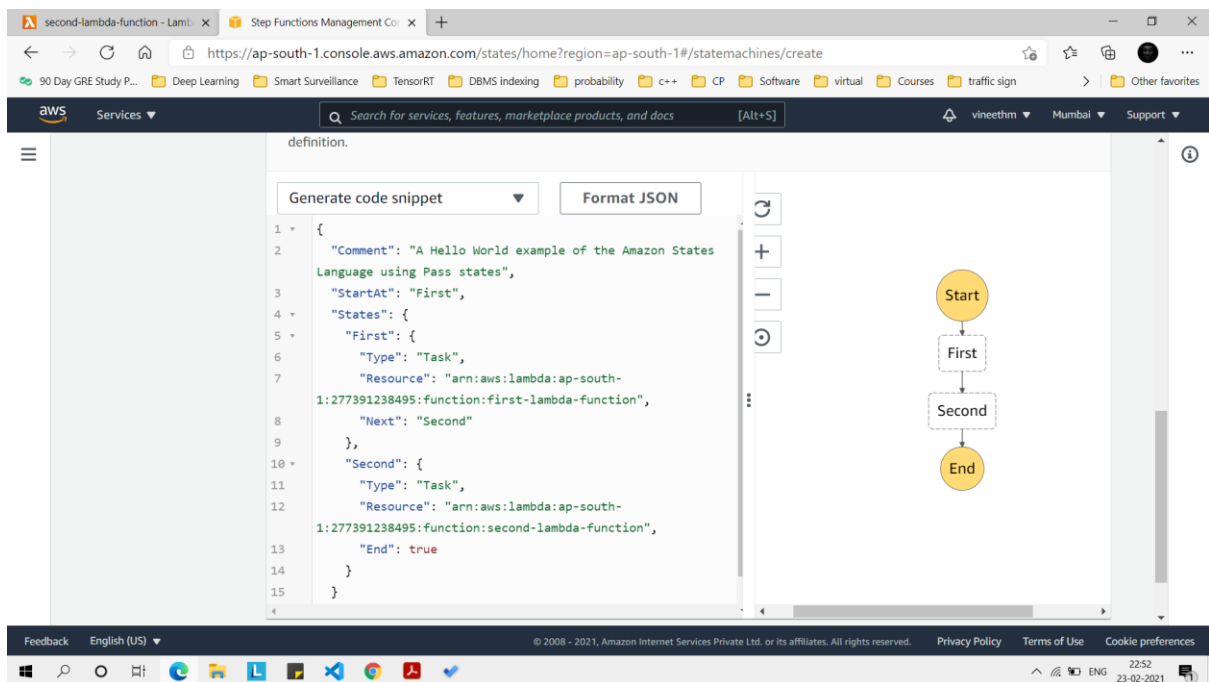


The screenshot shows the AWS Step Functions console interface. A green banner at the top indicates "State machine successfully created". The main content area displays "StateMachine1" with buttons for "Edit", "Start execution", "Delete", and "Actions". Below this, the "Details" tab is active, showing the following information:

| Property | Value |
|---------------|---|
| ARN | arn:aws:states:ap-south-1:277391238495:stateMachine:StateMachine1 |
| Type | Standard |
| Creation date | Feb 23, 2021 10:54:09.394 PM |
| IAM role ARN | arn:aws:iam::277391238495:role/Step_function |

The left sidebar shows navigation options: "State machines", "Activities", "Feature spotlight", and "Join our feedback panel". The bottom of the console shows the "Executions", "Logging", "Definition", and "Tags" tabs.

Step Function code:



The screenshot shows the AWS Step Functions console interface for creating a new state machine. The "definition" tab is active, displaying a JSON code snippet and a visual state machine diagram.

Generate code snippet **Format JSON**

```
1 {
2   "Comment": "A Hello World example of the Amazon States
3   Language using Pass states",
4   "StartAt": "First",
5   "States": {
6     "First": {
7       "Type": "Task",
8       "Resource": "arn:aws:lambda:ap-south-
9       1:277391238495:function:first-lambda-function",
10      "Next": "Second"
11    },
12    "Second": {
13      "Type": "Task",
14      "Resource": "arn:aws:lambda:ap-south-
15      1:277391238495:function:second-lambda-function",
16      "End": true
17    }
18  }
19 }
```

The visual diagram on the right shows a flow starting from a yellow circle labeled "Start", passing through a dashed box labeled "First", then a dashed box labeled "Second", and finally ending at a yellow circle labeled "End".

Task – 5: Test the AWS Step function by reviewing the output and the code in the design panel.

Click on the start execution button.

The screenshot shows the AWS Step Functions console for a state machine named `8e0b6dbb-9a31-d5cc-515b-bd35a15981ca`. The execution status is **Succeeded**. The execution ARN is `arn:aws:states:ap-south-1:277391238495:execution:StateMachine1:8e0b6dbb-9a31-d5cc-515b-bd35a15981ca`. The execution started on Feb 23, 2021 at 11:05:36.283 PM and ended at 11:05:36.643 PM.

| Details | Execution input | Execution output | Definition |
|---|-----------------|------------------|---|
| Execution Status ✔ Succeeded | | | Started Feb 23, 2021 11:05:36.283 PM |
| Execution ARN arn:aws:states:ap-south-1:277391238495:execution:StateMachine1:8e0b6dbb-9a31-d5cc-515b-bd35a15981ca | | | End Time Feb 23, 2021 11:05:36.643 PM |

Output from the first Lambda Function:

The screenshot shows the AWS Step Functions console with the **Graph inspector** view. The state machine diagram shows a linear flow: **Start** (yellow circle) → **First** (green rectangle) → **Second** (green rectangle) → **End** (yellow circle). The **Step output** for the **First** step is displayed as a JSON object:

```

1 {
2   "name": "Vineeth",
3   "sem": "Sem-VI",
4   "lab": "elective"
5 }

```

This is provided as the input to the second Lambda Function:

The screenshot shows the AWS Step Functions console. The 'Graph inspector' is open, displaying a state machine graph with four states: Start (yellow circle), First (green rectangle), Second (green rectangle), and End (yellow circle). The 'Step input' tab is selected, showing the input JSON for the 'Second' state:

```
1 {  
2   "name": "Vineeth",  
3   "sem": "Sem-VI",  
4   "lab": "elective"  
5 }
```

Output of the second Lambda Function:

The screenshot shows the AWS Step Functions console. The 'Graph inspector' is open, displaying the same state machine graph. The 'Step output' tab is selected, showing the output JSON for the 'Second' state:

```
1 {  
2   "name": "Vineeth",  
3   "sem": "Sem-VI",  
4   "lab": "elective",  
5   "result": "VALID"  
6 }
```

As we can notice, the result comes out to be “VALID” indicating the successful communication between the 2 Lambda function.