

S3

Steps to upload a web application file in s3 bucket.

First of all we have to create a bucket with some requirements as public .

Amazon S3 > Buckets > Create bucket

Create bucket [Info](#)

Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name

Bucket name must be unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#)

AWS Region

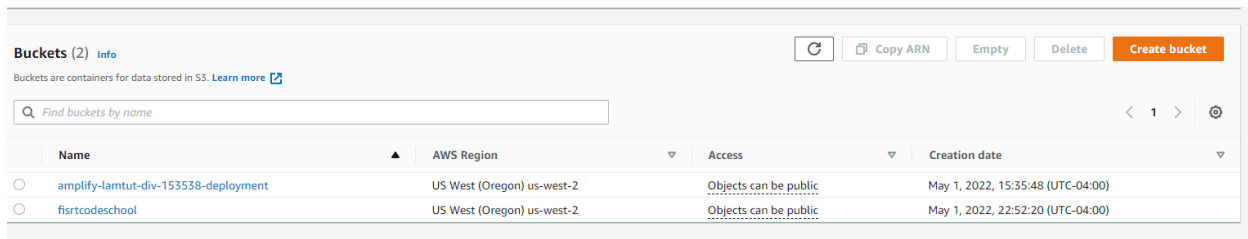
US West (Oregon) us-west-2 ▼

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

Choose bucket

After creating a bucket we can see created buckets in dashboard.

Firstcodeschool is my bucket name for lambda.



The screenshot shows the AWS S3 Buckets dashboard. At the top, there's a header with 'Buckets (2)' and an 'Info' link. Below this, a search bar is labeled 'Find buckets by name'. A table lists the buckets with columns for Name, AWS Region, Access, and Creation date. Two buckets are listed: 'amplify-lamtut-div-153538-deployment' and 'firstcodeschool'. Both are in the 'US West (Oregon) us-west-2' region and have 'Objects can be public' access. The 'firstcodeschool' bucket was created on May 1, 2022, at 22:52:20 (UTC-04:00).

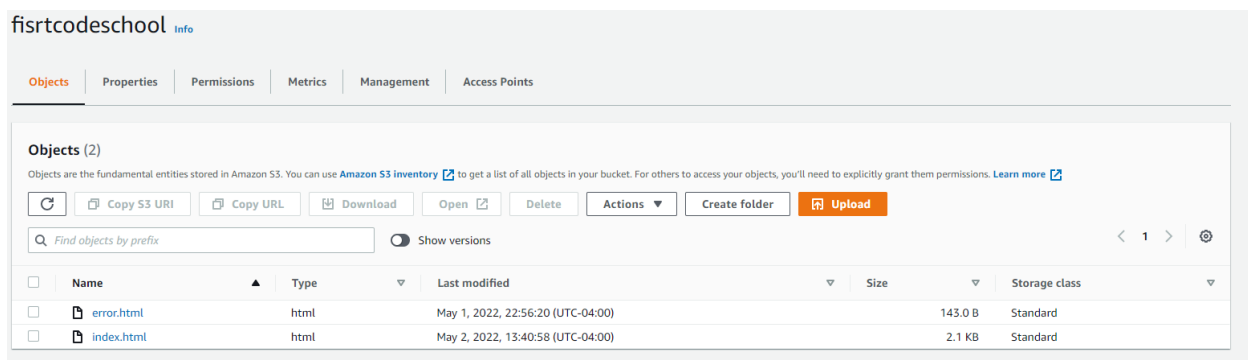
	Name	AWS Region	Access	Creation date
<input type="radio"/>	amplify-lamtut-div-153538-deployment	US West (Oregon) us-west-2	Objects can be public	May 1, 2022, 15:35:48 (UTC-04:00)
<input type="radio"/>	firstcodeschool	US West (Oregon) us-west-2	Objects can be public	May 1, 2022, 22:52:20 (UTC-04:00)

Then I upload index.html and error.html files in my bucket.

After that I have given permission to public access

So that It will create one http link to access our html files.

<https://firstcodeschool.s3.us-west-2.amazonaws.com/index.html>



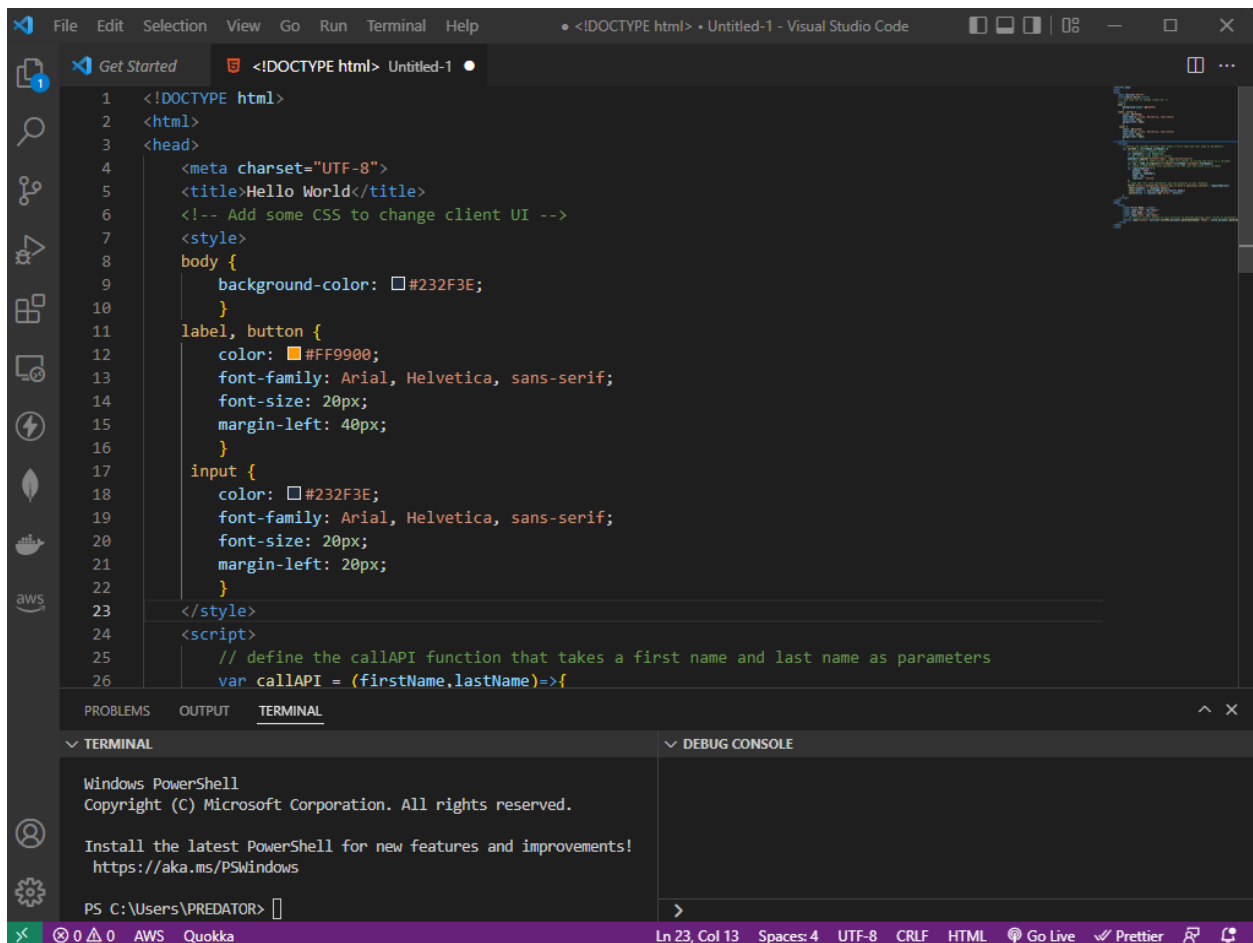
The screenshot shows the AWS S3 Objects page for the 'firstcodeschool' bucket. At the top, there's a header with 'firstcodeschool' and an 'Info' link. Below this, there are tabs for 'Objects', 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'. The 'Objects' tab is selected. Below the tabs, there's a section titled 'Objects (2)' with a description and a 'Learn more' link. A toolbar contains buttons for 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', 'Actions', 'Create folder', and 'Upload'. A search bar is labeled 'Find objects by prefix'. A table lists the objects with columns for Name, Type, Last modified, Size, and Storage class. Two objects are listed: 'error.html' and 'index.html'. Both are 'html' files. 'error.html' was last modified on May 1, 2022, at 22:56:20 (UTC-04:00) and is 143.0 B in size. 'index.html' was last modified on May 2, 2022, at 13:40:58 (UTC-04:00) and is 2.1 KB in size. Both are stored in the 'Standard' storage class.

	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	error.html	html	May 1, 2022, 22:56:20 (UTC-04:00)	143.0 B	Standard
<input type="checkbox"/>	index.html	html	May 2, 2022, 13:40:58 (UTC-04:00)	2.1 KB	Standard

I finished with s3 bucket.

By using VS code

I have created a code and deployed into aws using terminal and extensions in vscode.



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>Hello World</title>
6   <!-- Add some CSS to change client UI -->
7   <style>
8     body {
9       background-color: #232F3E;
10    }
11    label, button {
12      color: #FF9900;
13      font-family: Arial, Helvetica, sans-serif;
14      font-size: 20px;
15      margin-left: 40px;
16    }
17    input {
18      color: #232F3E;
19      font-family: Arial, Helvetica, sans-serif;
20      font-size: 20px;
21      margin-left: 20px;
22    }
23  </style>
24  <script>
25    // define the callAPI function that takes a first name and last name as parameters
26    var callAPI = (firstName,lastName)=>{
```

Lambda

Step to create lambda function

First, we have selected a create function button from dashboard then

We must give Function name and runtime so I have given python 3.8

After that some we have to give some permission setting according to the files.

Lambda > Functions > Create function

Create function [Info](#)

Choose one of the following options to create your function.

Author from scratch ☒

Start with a simple Hello World example.

Use a blueprint ☐

Build a Lambda application from sample code and configuration presets for common use cases.

Container image ☐

Select a container image to deploy for your function.

Basic information

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. Note that the console code editor supports only Node.js, Python, and Ruby.

Architecture [Info](#)
Choose the instruction set architecture you want for your function code.

☒ x86_64
☐ arm64

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](#)

After creating function we can see created function on dashboard

Below is my function for this task Is Helloworldfunction

Lambda > Functions

Functions (5) Last fetched in 0 seconds [Refresh](#) [Actions](#) [Create function](#)

<input type="checkbox"/>	Function name	Description	Package type	Runtime	Code size	Last modified
<input type="checkbox"/>	Helloworldfunction	-	Zip	Python 3.8	439.0 byte	9 hours ago
<input type="checkbox"/>	gfrFunc-div	-	Zip	Node.js 12.x	1.1 kB	1 day ago
<input type="checkbox"/>	hexal	-	Zip	Node.js 14.x	304.0 byte	1 day ago
<input type="checkbox"/>	get	-	Zip	Node.js 14.x	304.0 byte	1 day ago
<input type="checkbox"/>	hexalput	-	Zip	Node.js 12.x	564.0 byte	1 day ago

Then we can see some tabs that are code, Test, Monitor, configuration and etc.

In code tab we must create a code, so I copied a code form GitHub and I pasted in pad.

Code source [Info](#)

File Edit Find View Go Tools Window

TestDeploy

Go to Anything (Ctrl-P)

Environment

Helloworldfunction - /

lambda_function.py

lambda_function x

```
1 import json
2 import boto3
3 from time import gmtime, strftime
4
5 dynamodb = boto3.resource('dynamodb')
6 table = dynamodb.Table('Helloworlddatabase')
7 now = strftime("%a, %d %b %Y %H:%M:%S +0000", gmtime())
8
9 def lambda_handler(event, context):
10     name = event['firstName'] + ' ' + event['lastName']
11     response = table.put_item(
12         Item={
13             'ID': name,
14             'LatestGreetingTime':now
15         })
16     return {
17         'statusCode': 200,
18         'body': json.dumps('Hello from Lambda, ' + name)
19     }
```

In this code we created some import json and some events to get First name and Last name .

With some status code 200.

Configure test event

A test event is a JSON object that mocks the structure of requests emitted by AWS services to invoke a Lambda function. Use it to see the function's invocation result.

To invoke your function without saving an event, modify the event, then choose Test. Lambda uses the modified event to invoke your function, but does not overwrite the original event until you choose Save changes.

Test event action

☐ Create new event

☒ Edit saved event

Event name

Newevent

▼

↺

Delete

Event JSON

Format JSON

```
1 {
2   "firstName": "gannon",
3   "lastName": "university"
4 }
```


API Gateway

Step to create a API Gateway

First I have selected a build rest API to build a api then

We have to some names and region details to work .

Choose the protocol

Select whether you would like to create a REST API or a WebSocket API.

☒ REST ☐ WebSocket

Create new API

In Amazon API Gateway, a REST API refers to a collection of resources and methods that can be invoked through HTTPS endpoints.

☒ New API ☐ Clone from existing API ☐ Import from Swagger or Open API 3 ☐ Example API

Settings

Choose a friendly name and description for your API.

API name*

My API

Description

Endpoint Type

Edge optimized

* Required

After creating a api we can see APIs in dashboard

Helloworld is the api I created for this task

APIs (2)							Find APIs	Actions	Create API
	Name	Description	ID	Protocol	Endpoint type	Created			
<input type="radio"/>	gfrAPI		pyk2gx1r07	REST	Edge	2022-05-01			
<input type="radio"/>	HelloworldAPI		v4458ys6f7	REST	Edge	2022-05-02			

After that I have selected a action as post

Resources

Actions

/ Methods

OPTIONS

POST

Mock Endpoint

Authorization None

API Key Not required

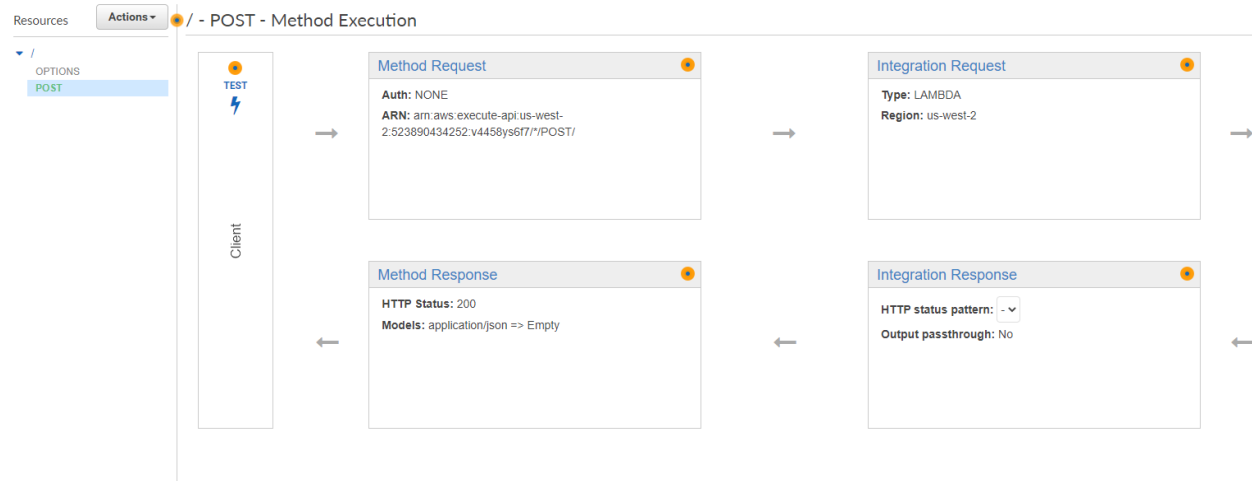
POST

arn:aws:lambda:us-west-2:523890434252:functi...

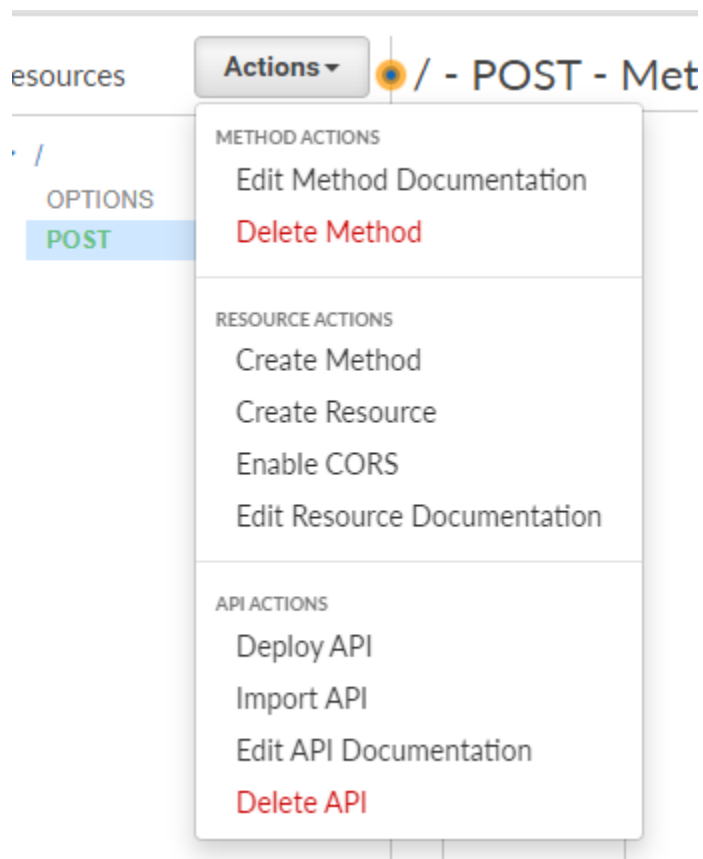
Authorization None

API Key Not required

Then we can see connection api to connect to lambda



After that I edited cors and deploy API s



To check api are working I have copied a link from aws api gateway then pasted POSTMAN

So it showed we good results its working

Overview

POST https://v4458ys6f7.e

No Environment

https://v4458ys6f7.execute-api.us-west-2.amazonaws.com/dev

Save

Send

ParamsAuthorizationHeaders (8)Body●Pre-request ScriptTestsSettingsCookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

BodyCookiesHeaders (12)Test Results

200 OK1097 ms587 BSave Response

PrettyRawPreviewVisualizeJSON

```
1
2  "statusCode": 200,
3  "body": "\"Hello from Lambda, nac greenwood\""
4
```

CookiesCapture requestsBootcampRunnerTrash

Dynamo DB

Step to create a table and link to Lambda function

DynamoDB > Tables > Create table

Create table

Table details

Info

DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

Table name

This will be used to identify your table.

Enter name for table

Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (.).

Partition key

The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.

IDString

1 to 255 characters and case sensitive.

Sort key - optional

You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.

Enter the sort key nameString

1 to 255 characters and case sensitive.

Settings

Default settings

The fastest way to create your table. You can modify these settings now or after your table has been created.

Customize settings

Use these advanced features to make DynamoDB work better for your needs.

First of all I have created a table

By giving a table name, partition key and some default settings .

DynamoDB > Tables

Tables (4) [Info](#)

[Refresh](#) [Actions](#) [Delete](#) [Create table](#)

[<](#) [1](#) [>](#) [⚙️](#)

<input type="checkbox"/>	Name	Status	Partition key	Sort key	Indexes	Read capacity mode	Write capacity mode	Size	Table class
<input type="checkbox"/>	Helloworlddatabase	Active	ID (S)	-	0	Provisioned with auto scaling (1)	Provisioned with auto scaling (1)	506 bytes	DynamoDB Standard
<input type="checkbox"/>	NewTable	Active	ID (S)	-	0	Provisioned with auto scaling (1)	Provisioned with auto scaling (1)	0 bytes	DynamoDB Standard
<input type="checkbox"/>	Products	Active	id (S)	-	0	Provisioned with auto scaling (1)	Provisioned with auto scaling (1)	10 bytes	DynamoDB Standard
<input type="checkbox"/>	Todo-75uf7optpjeanozy7vs2urdc5a-div	Active	id (S)	-	0	On-demand	On-demand	7 bytes	DynamoDB Standard

We have see our tables what we created on dashboard.

For this task Helloworlddatabase is table name .

Helloworlddatabase [Refresh](#) [Actions](#) [Explore table items](#)

[Overview](#) [Indexes](#) [Monitor](#) [Global tables](#) [Backups](#) [Exports and streams](#) [Additional settings](#)

General information

Partition key ID (String)	Sort key -	Capacity mode Provisioned	Table status Active No active alarms
------------------------------	---------------	------------------------------	--

▼ Additional info

Table class DynamoDB Standard	Indexes 0 globals, 0 locals	DynamoDB stream Disabled	Point-in-time recovery (PITR) Disabled
Time to Live (TTL) Info Disabled	Replication Regions 0 Regions	Encryption Owned by Amazon	Date created May 1, 2022, 23:57:02 (UTC-04:00)

Amazon Resource Name (ARN)
 arn:aws:dynamodb:us-west-2:523890434252:table/Helloworlddatabase

These are full details of our table

We have copy the ARN (amazon resource name) from here and we have paste this link in lambda in policies to get link those two functions.

IAM

Steps to create a IAM roles for DynamoDB

Select trusted entity

Trusted entity type

☒ **AWS service**
Allow AWS services like EC2, Lambda, or others to perform actions in this account.

☐ **AWS account**
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

☐ **Web identity**
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

☐ **SAML 2.0 federation**
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

☐ **Custom trust policy**
Create a custom trust policy to enable others to perform actions in this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Common use cases

- ☐ **EC2**
Allows EC2 instances to call AWS services on your behalf.
- ☒ **Lambda**
Allows Lambda functions to call AWS services on your behalf.

Use cases for other AWS services:

Choose a service to view use case

We have select trusted entity to create a role so I selected aws services with lambda function

Then after we have to give a permissions policies for that so I have selected AWSLambdaBasic Execution from using search bar.

Permissions policies (Selected 1/753)

Choose one or more policies to attach to your new role.

4 matches

"basic" X Clear filters

	Policy name	Type	Description
<input type="checkbox"/>	AWSLambdaBasicExec...	Custom...	
<input type="checkbox"/>	AWSLambdaBasicExec...	Custom...	
<input type="checkbox"/>	AWSLambdaBasicExec...	Custom...	
<input checked="" type="checkbox"/>	AWSLambdaBasicEx...	AWS ma...	Provides write permissions to CloudWatch Logs.

After that we have to give policy to run lambda function

so here two option to give policy like visual editor and json format.

So I have selected Json format to create a policy.

Edit HelloworldDyanmoDBPolicy

1 2

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editorJSON

Import managed policy

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "VisualEditor0",
6       "Effect": "Allow",
7       "Action": [
8         "dynamodb:PutItem",
9         "dynamodb>DeleteItem",
10        "dynamodb:GetItem",
11        "dynamodb:Scan",
12        "dynamodb:Query",
13        "dynamodb:UpdateItem"
14      ],
15       "Resource": "arn:aws:dynamodb:us-west-2:523890434252:table/Helloworlddatabase"
16     }
17   ]
18 }
```

Security: 0Errors: 0Warnings: 0Suggestions: 0

In this policy I have given related to code like putitem , Deleteltem, GetItem, scan, Query and UpdatItem.

Introducing the new IAM roles experience

We've redesigned the IAM roles experience to make it easier to use. [Let us know what you think.](#)

IAM > Roles

Roles (13) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Role name

Trusted entities

Last activity

☐

amplify-lamut-div-153538-authRole

None

-

☐

amplify-lamut-div-153538-unauthRole

None

-

☐

AWSServiceRoleForAPIGateway

AWS Service: ops.apigateway (Service-Linked Role)

-

☐

AWSServiceRoleForApplicationAutoScaling_DynamoDBTable

AWS Service: dynamodb.application-autoscaling (Service-Linked Role)

17 minutes ago

☐

AWSServiceRoleForSupport

AWS Service: support (Service-Linked Role)

-

☐

AWSServiceRoleForTrustedAdvisor

AWS Service: trustedadvisor (Service-Linked Role)

-

☐

helloworld-role-zxxvv0ev

AWS Service: lambda

-

☐

Helloworldfunction-role-ismx5gyd

AWS Service: lambda

17 hours ago

☐

hexal-lambda-basic-execution

AWS Service: lambda

Yesterday

After reviewing the policy then we can see roles on dashboard.

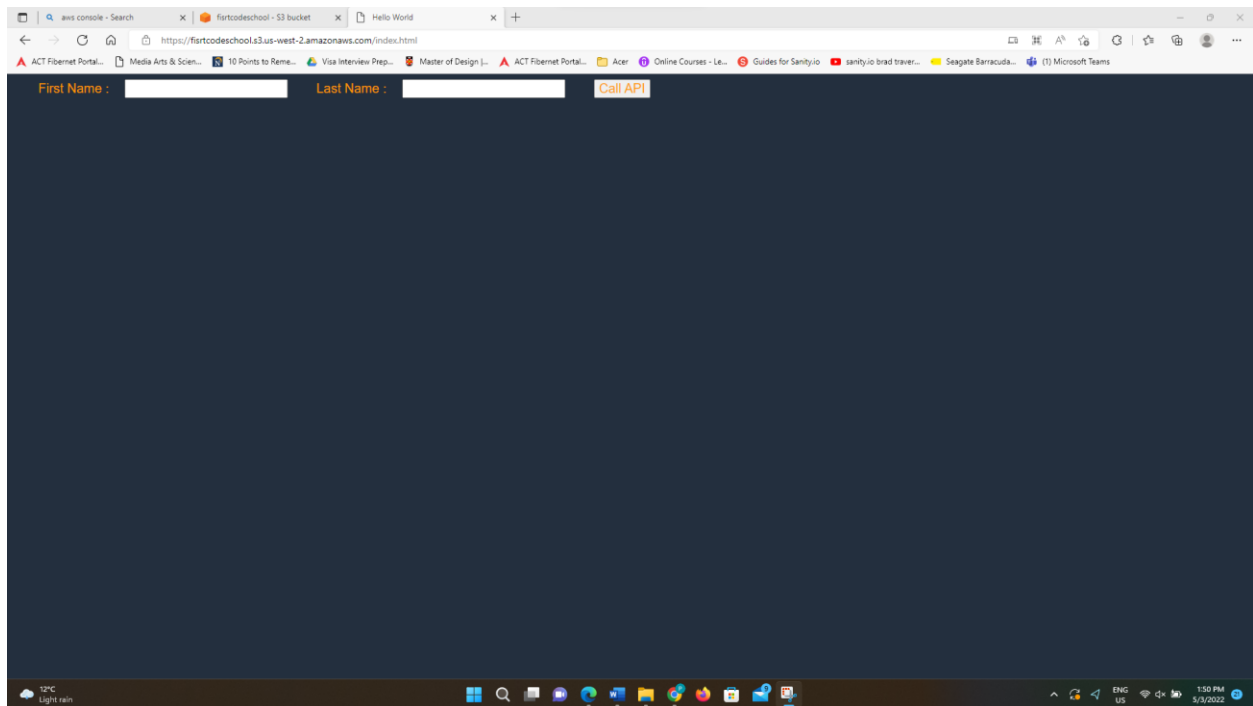
We have finished all task now.

Output of Lambda Function DynamoDB

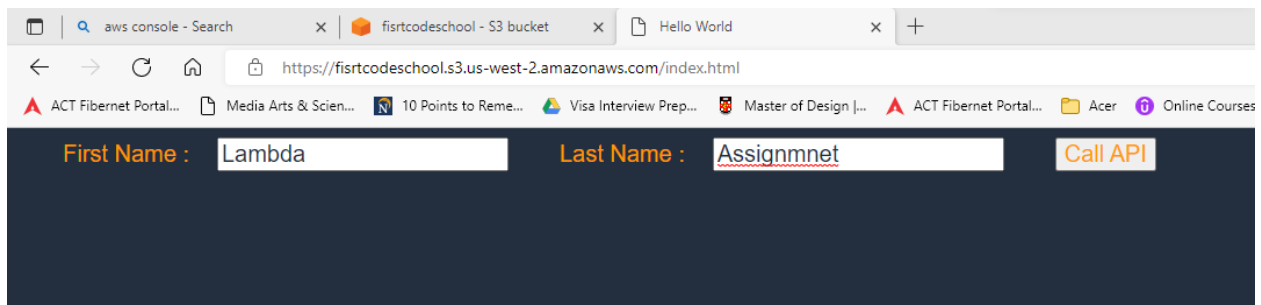
First we have to go back to S3 bucket and select our index.html file to open on browser.

[Hello World \(amazonaws.com\)](https://firtcodeschool.s3.us-west-2.amazonaws.com/index.html)

<https://firtcodeschool.s3.us-west-2.amazonaws.com/index.html>



This is the page we created in vscode by using html, css and python.

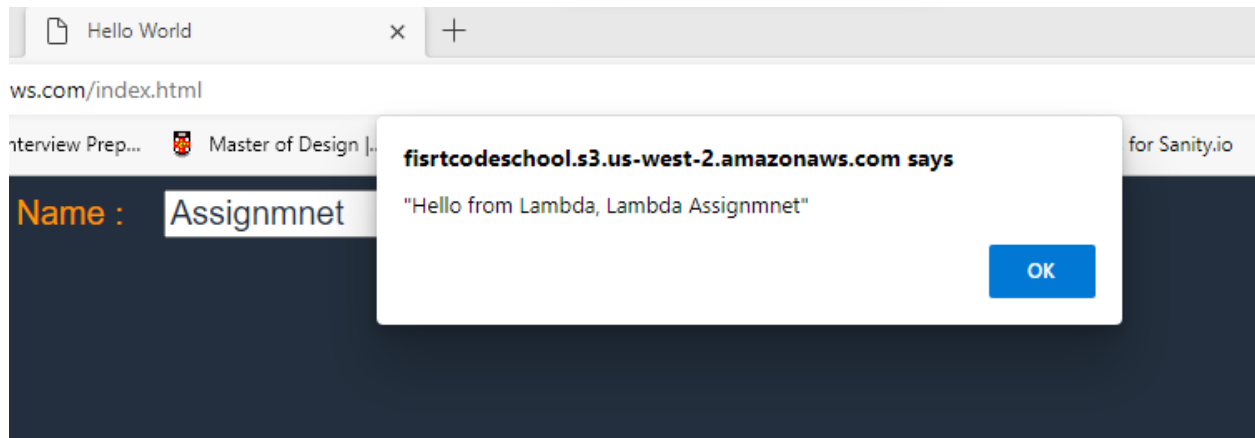


Then we have to type anything on tabs to reflect on DynamoDB by using Lambda.

While clicking on call Api

We can see a pop on browser to show the output

“Hello from Lambda Assignment”



We can see our api what we called on browser it will say on DynamoDB on Items.

Total I have created 7 items and we can create more api and it will directly save on table.

We can also delete items from table.

And we can update table settings in DynamoDB.

The screenshot shows the AWS DynamoDB console interface. On the left, a sidebar lists tables: 'Helloworlddatabase', 'NewTable', 'Products', and 'Todo-75uf7optpjeanozy7vs2urdc5a-div'. The main area displays the 'Helloworlddatabase' table. At the top right of the main area are buttons for 'Autopreview', 'Actions', 'Create item', and 'Update table settings'. Below these is a 'Scan/Query items' section. The 'Items returned (7)' section shows a table with 7 items. Each item has a checkbox, an 'ID' column, and a 'LatestGreetingTime' column.

	ID	LatestGreetingTime
<input type="checkbox"/>	gannon uni...	Tue, 03 May 2022 00:09:57 +0000
<input type="checkbox"/>	exjwe xhjwsx	Mon, 02 May 2022 17:38:53 +0000
<input type="checkbox"/>	nac greenw...	Tue, 03 May 2022 00:23:26 +0000
<input type="checkbox"/>	coffaman g...	Mon, 02 May 2022 17:38:53 +0000
<input type="checkbox"/>	Lambda Ass...	Tue, 03 May 2022 17:52:57 +0000
<input type="checkbox"/>	samantha ss	Mon, 02 May 2022 17:38:53 +0000
<input type="checkbox"/>	pranay reddy	Mon, 02 May 2022 14:56:04 +0000