

WELCOME TO BLACK MEN CODE

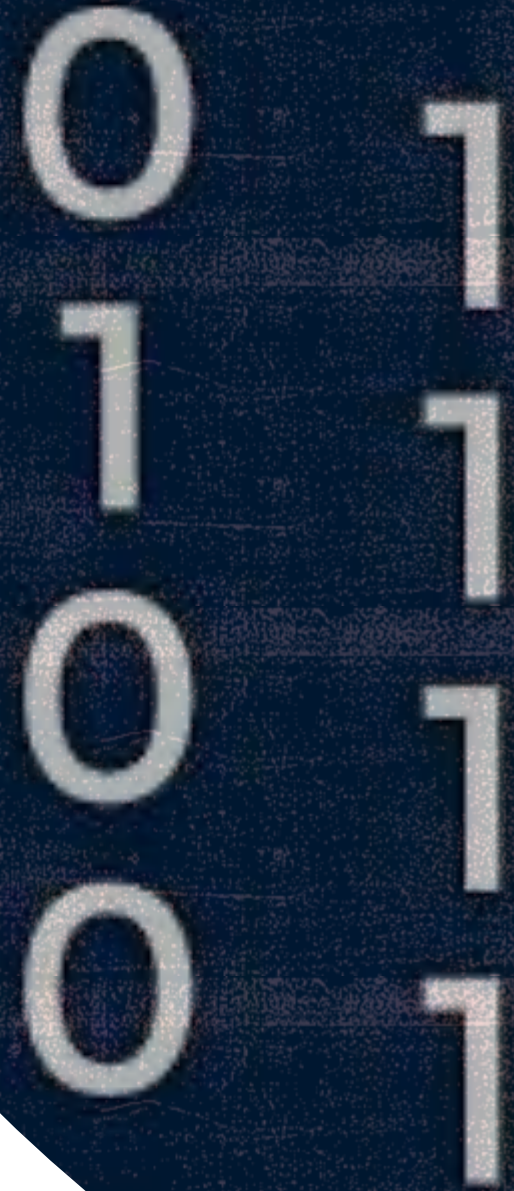
AWS LAMBDA TRAINING

**PRESENTED BY:
CALVIN BROWN (ADJO)
AWS CERTIFIED ARCHITECT**



OUR MISSION

Black Men Code is a fiscally sponsored, non-profit, social benefit organization. **Our mission is to provide free and low-cost training in computer programming that inspires and motivates Black men** to: (1) pursue education in STEM fields and (2) participate in the innovation economy through entrepreneurship.



WHAT IS AWS

Amazon Web Services (AWS) is a secure cloud services platform, offering compute power, database storage, content delivery and other functionality to help businesses scale and grow.

- ❖ Running web and application servers in the cloud to host dynamic websites.
- ❖ Securely store all your files on the cloud so you can access them from anywhere.
- ❖ Using managed databases like MySQL, PostgreSQL, Oracle or SQL Server to store information.
- ❖ Deliver static and dynamic files quickly around the world using a Content Delivery Network (CDN).
- ❖ Send bulk email to your customers



WHAT IS **AWS** **LAMBDA**

AWS Lambda is an event-driven, serverless computing platform provided by Amazon as a part of Amazon Web Services. It is a computing service that runs code in response to events and automatically manages the computing resources required by that code. It was introduced in November 2014.

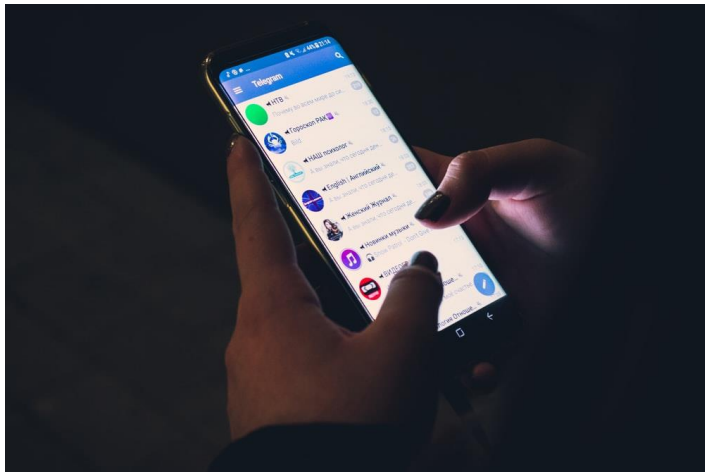


Amazon
Lambda



HOW CAN I USE LAMBDA WITH MY MOBILE OR WEB APP

- ❖ You can call a Lambda function from a mobile application. Put business logic in functions to separate its development lifecycle from that of front-end clients, making mobile applications less complex to develop and maintain. With the Mobile SDK for Android, you use Amazon Cognito to authenticate users and authorize requests.
- ❖ When you invoke a function from a mobile application, you choose the event structure, invocation type, and permission model. You can use aliases to enable seamless updates to your function code, but otherwise the function and application are tightly coupled. As you add more functions, you can create an API layer to decouple your function code from your front-end clients and improve performance.
- ❖ To create a fully-featured web API for your mobile and web applications, use Amazon API Gateway. With API Gateway, you can add custom authorizers, throttle requests, and cache results for all of your functions.

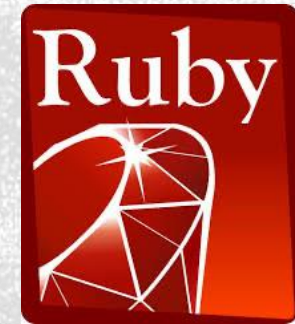


WHAT LANGUAGES CAN YOU USE IN **AWS** **LAMBDA?**

- ❖ Node. Js
- ❖ Python
- ❖ Java
- ❖ Ruby
- ❖ C#
- ❖ Go
- ❖ PowerShell



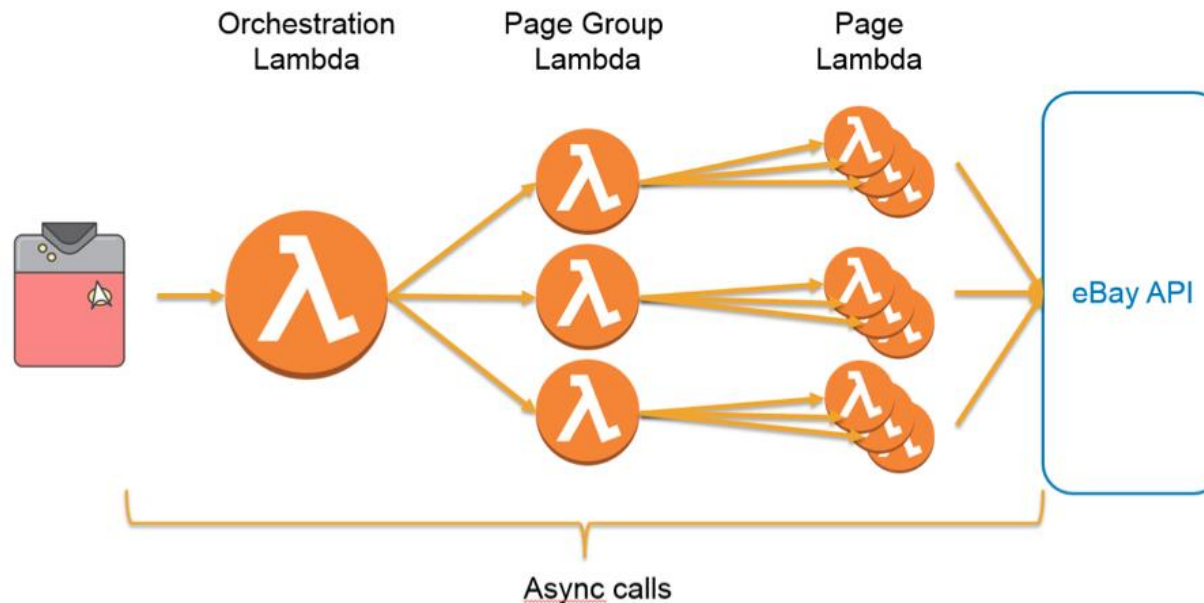
PowerShell



SCALING IN AWS LAMBDA

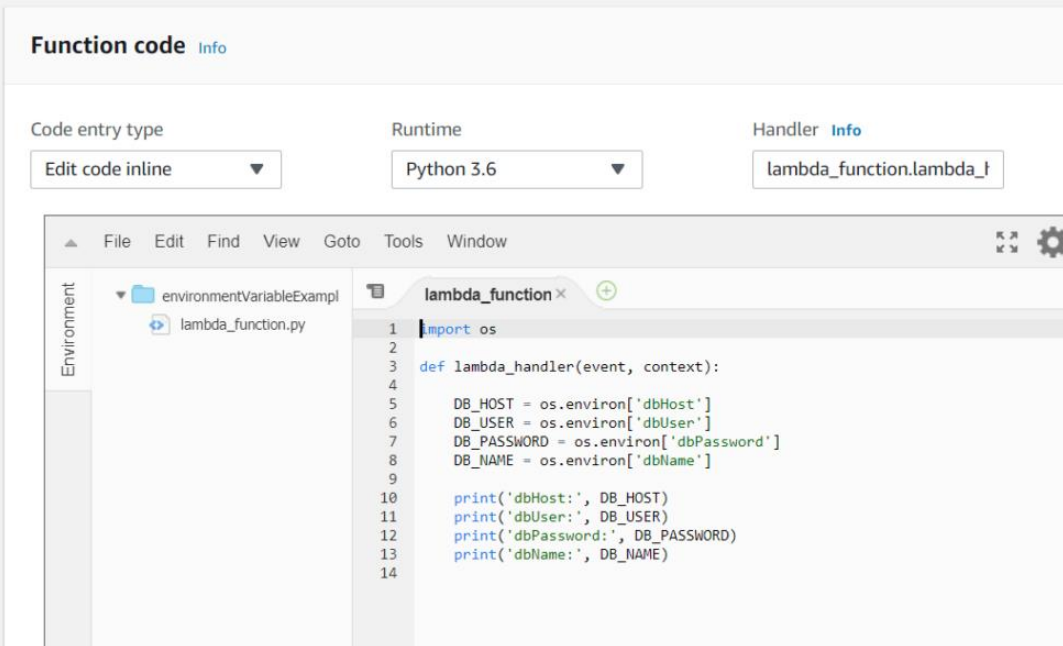
You do not have to scale your Lambda functions – AWS Lambda scales them automatically on your behalf.

Every time an event notification is received for your function, AWS Lambda quickly locates free capacity within its compute fleet and runs your code.



MODIFYING VARIABLES INSIDE LAMBDA

- ❖ A lambda expression can capture variables like local and anonymous classes.
- ❖ Inside lambda expression, we can't assign any value to some local variable declared outside the lambda expression.
- ❖ Because the local variables declared outside the lambda expression can be final or effectively final.



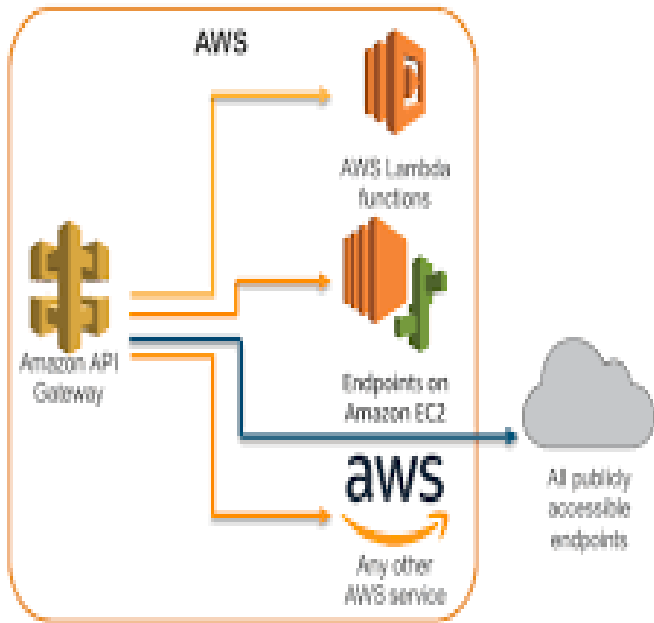
Function code [Info](#)

Code entry type: Runtime: Handler: [Info](#)

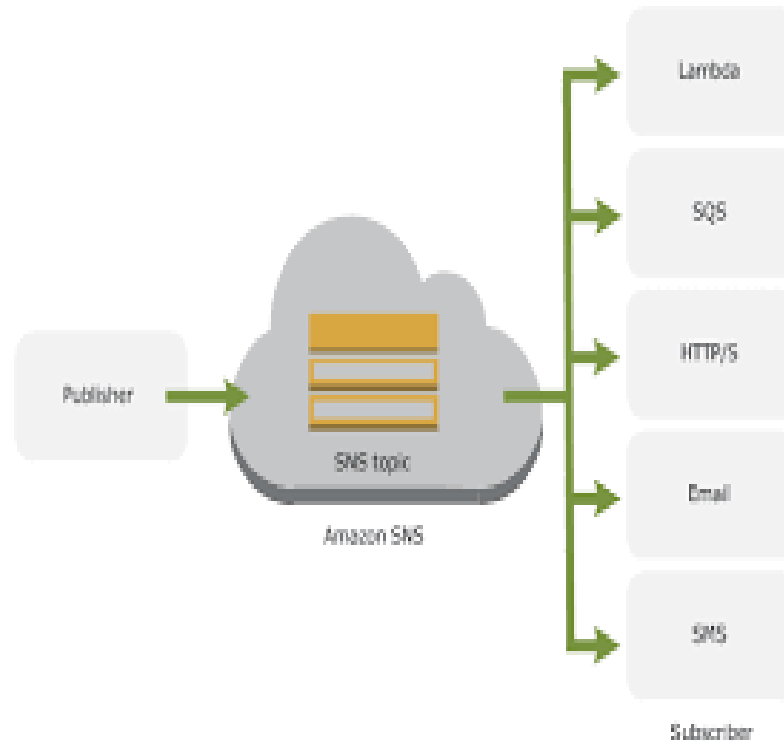
```
File Edit Find View Goto Tools Window
Environment
  environmentVariableExamp
    lambda_function.py
lambda_function
1 import os
2
3 def lambda_handler(event, context):
4
5     DB_HOST = os.environ['dbHost']
6     DB_USER = os.environ['dbUser']
7     DB_PASSWORD = os.environ['dbPassword']
8     DB_NAME = os.environ['dbName']
9
10    print('dbHost:', DB_HOST)
11    print('dbUser:', DB_USER)
12    print('dbPassword:', DB_PASSWORD)
13    print('dbName:', DB_NAME)
14
```



AWS ADVANCED TOPICS



- ❖ Amazon **API Gateway** is an **AWS** service for creating, publishing, maintaining, monitoring, and securing REST, HTTP, and WebSocket **APIs** at any scale.



- ❖ Amazon **Simple Notification Service** (Amazon SNS) is a web service that coordinates and manages the delivery or sending of messages to subscribing endpoints or clients



AWS Step Functions

The screenshot shows the AWS Step Functions console for a specific execution. The execution ID is d3b71093-335a-327c-7647-898e4bf1d4a0, and it is marked as successful with a green checkmark. The console displays a state machine graph with the following steps: Start (yellow circle), Hello (green square), a parallel split (dashed box containing p1 and p2, both green squares), Goodbye (green square), and End (yellow circle). The right-hand panel shows 'Execution Details' with the following information:

- Execution Status:** Succeeded
- State Machine Arn:** arn:aws:states.eu-west-1:stateMachine:MyStateMachine
- Execution ID:** arn:aws:states.eu-west-1:execution:MyStateMachine:d3b71093-335a-327c-7647-898e4bf1d4a0
- Started:** Nov 23, 2016 3:07:06 PM
- Closed:** Nov 23, 2016 3:07:07 PM

- ❖ **AWS Step Functions** lets you coordinate multiple **AWS** services into serverless workflows so you can build and update apps quickly.

Q&A

