Lambda

16. Lambda

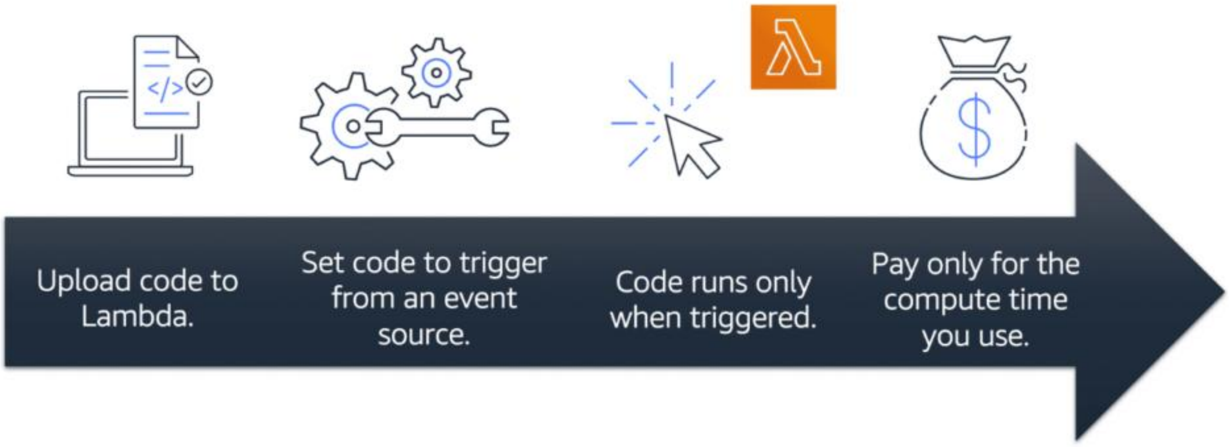
AWS Lambda is a service that lets you run code without needing to provision or manage servers.

1. AWS Lambda is compute service that let’s you run code without provisioning or managing servers.
2. With AWS lambda, you can run code for virtually any type of application or back and service all with zero administration.

- AWS Lambda manages all the administration it manages
- Provisioning and capacity of the computer split that offers a balance of memory.
- Server and O S maintenance.
- High availability and automatic scaling.
- Monitoring fleet health
- Applying Security patches
- Deploying your Code.
- Monitoring and logging your Lambda Functions
- AWS Lambda Runs your Code on a high-availability Compute Infrastructure.

While using AWS Lambda, you pay only for the compute time that you consume. Charges apply only when your code is running. You can also run code for virtually any type of application or backend service, all with zero administration.

For example, a simple Lambda function might involve automatically resizing uploaded images to the AWS Cloud. In this case, the function triggers when uploading a new image.


Upload code to Lambda.
Set code to trigger from an event source.
Code runs only when triggered.
Pay only for the compute time you use.

- AWS Lambda runs your code on a high availability compute infrastructure.
- AWS lambda executes your code only when needed and scales automatically from a few request per day to thousands per second.

- You pay only for the compute time you consume – No charge when you are code is not running.
- All you need to do is supply your code in the form of one or more Lambda functions to a w s Lambda in one of the languages that AWS Lambda supports (currently Note JS, Java, power shell, Ruby C#, Python and Go) and the service can run the code on your behalf.
- Typically the life cycle for an AWS Labra based application includes authoring code deploying code to AWS Lambda and then monitoring and trouble shooting..
- This is in exchange for flexibility which means you cannot log into computer instances or customized the operating system or language run time.
- If you do want to manage your own compute, you can use EC2 or elastic Beanstalk.
- AWS Lambda runs you are code on a high availability compute infrastructure.
- AWS lambda executes you are code only when needed and scales automatically from a few request per day to thousands per second.

- You pay only for the compute time you consume – No charge when you are code is not running.
- All you need to do is supply your code in the form of one or more Lambda functions to a w s Lambda in one of the languages that AWS Lambda supports (currently Note JS, Java, power shell, Ruby C#, Python and Go) and the service can run the code on your behalf.
- Typically the life cycle for an AWS Labra based application includes authoring code deploying code to AWS Lambda and then monitoring and trouble shooting..
- This is in exchange for flexibility which means you cannot log into computer instances or customized the operating system or language run time.
- If you do want to manage your own compute, you can use EC2 or elastic Beanstalk.

How AWS Lambda works

1. You upload your code to Lambda.
2. You set your code to trigger from an event source, such as AWS services, mobile applications, or HTTP endpoints.
3. Lambda runs your code only when triggered.
4. You pay only for the compute time that you use. In the previous example of resizing images, you would pay only for the compute time that you use when uploading new images. Uploading the images triggers Lambda to run code for the image resizing function.

When Lambda Triggers-

- You can use AWS to run your Code in response to-
- Events such as changes to data in an Amazon S3 bucket or and Amazon dynamo DB table.
- To run your code in response to http request using Amazon API gateway.
- With this capabilities you can use lambda to easily build data processing triggers for AWS services like Amazon S3 and Amazon dynamo DB, process streaming data stored in Kiness or create your own bacon that separates at AWS scale performance and security.

Important Terms used in Lambda-

1. **Function-** A function is a resource that you can in walk to run your code in a w s Lambda a functions has code that processes events and a run time that passes request and responses between Lambda and the function code.
2. **Runtime-** Lambda allows functions in different languages do run in the same best executioner environment the runtime sits in between the lambda service and you are function Code relaying in location even context information and response between the two.
3. **Event-** Is a JSON in formatted document that contains data for a function to process.

4. **Event Source/ Trigger-** An AWS service such as Amazon SNS are a custom service that triggers your functions and executes its logic.
5. **Downstream Resource-** An AWS service, such as dynamo DB tables are S3 buckets, that you are Lambda function calls once it is triggered.
6. **Concurrency-** No. Of Request your function is serving in any given time.