**DOCUMENTATION ABOUT HOW TO UPLOAD OUR ECLIPSE PROJECT INTO AWS ENVIRONMENT**

To deploy our project into AWS environment we need to follow some steps to follow

**Step 1:** At first we should have appropriate IDE with your project to upload in AWS environment in which I am using **Eclipse neon .3 version**

**Step 2:** Next we should have AWS software installed to your pc in which I am having AWS application with updated version

**Step 3:** In which AWS application is not enough to upload a project we should have an account in AWS then only we can open AWS environment

**Step 4:** After all these steps are finished we need to do some more steps to upload our project into AWS environment

**Step 5:** We should have to know some services in AWS to upload our project there are 90 services in AWS based on different domains but to upload our web application project we should have to know some services they are

**FEATURES About aws added on (25/10/2018)**

**Why do we use AWS?**

* It is easy to use even those who don’t know about program can also work on AWS and simple authentication process
* Third party services like cloud ranger and tango card are available to help you to utilize all the time and cost saving features
* Scheduling means that you can start stop AWS services at predetermined times at any time when you not in the work it is time timer
* Customization such as customer defined tagging allows user to easily monitor and manage the their specific records
* **Full stack services** – compute, storage, db, iam etc…… about 90 services
* Data privacy is offered to every customer regularly of their business size

1. Through physical security
2. Data locality control

* There is no additional charge for IAM
* **Availability –** the aws cloud span of about 54 availability zones within the geographical region around the world
* **Reliability –**  the reliability is achieved through the backup system in aws they have multiple backup servers thus we can store our code in s3 also called as simple storage service. Called as aws cloud with backup
* **Flexibility and stability –** flexibility is the sense in which we can start any time anywhere and stop according to it. stability is the sense it can be increased the storage as per our requirement in workspace
* Payment is about how much we are using according to which they charge
* AWS Ec2(Amazon Web Services Elastic cloud compute)
* AWS beanStalk
* AWS codeStar
* AWS Redshift(data warehouse)
* AWS s3
* AWS RDS

**Note :** The main feature is to use Ec2 but we need to know about beanstalk and codeStar we work on codeStar it includes beanStalk and Ec2. We want to use Ec2 but if we use ec2 we want to manually install servers to run so that we move to beanstalk and codestar which automate the process but it includes ec2 instance.

* After opening the account sign in with management console.
* After login in compute select AWS codestar in which create new project in which our project should be in type of “java spring with beanstalk web application”
* After that it ask to create project name so create the project name
* After that it ask which repository want to use it shows github and aws code commit use aws code commit
* After that it shows the pipe line of build, test, deploy on beanstalk and monitor on server.
* We just want to build our project and want to commit in IDE the amazing thing in AWS is it automated the build test and monitor on server
* After that it ask amazon key pair give it and than the project is created
* After that to rebuild our project we need to connect our project into an IDE where we can rearrange our project to that we should have connecting tool in our pc
* As I told earlier I have eclipse neon.3 version in which download plugin for connecting aws with eclipse by using eclipse market place
* And it ask to restart the eclipse after that the new icon display on eclipse though which if we click that it ask aws user name and password through which we can connect our project with eclipse IDE
* If we change anything in code via eclipse right click the project go to team and commit it will deploy in aws.

**AWS Redshift**

* Amazon redshift is OLAP – online analytical process
* It provides the services in amazon redshift is all about data warehouse.
* Database is where we store data and data warehouse is the place where we process the data for analytical purpose
* The amazon redshift is most powerful and scalable the data warehouse uses database only for example (Mysql, oracle, etc).
* Data warehouse can sit over top of file system also
* What is file system file is the one which we can store data without structure the database is the thing that we can store values in structured manner so that we use structured manner called structured quire language.
* Amazon redshift is the “distributed cluster” which sits on the top of the database the distributed cluster is in the sense of parallelism
* Parallelism is the meaning of if we want the data from database that if we type quires it will not move in one path it moves in different path of parallel ways and getting the values it is very fast than any other method
* The main advantage in amazon redshift is the columnar manner the data all are stored in column manner thus if we store values in row manner means the values move to the individual row which takes more memory and the operation will be slow
* So that in redshift they use column manner in which it moves to the particular column and fetch the values it is fast and reliable than any other method
* The values can store in s3 so that most of them using these.

**Amazon RDS**

* The amazon RDS is the relational database services
* Amazon **RDS** Database Engines. Amazon **RDS** makes it easy to go from project conception to deployment. **Use** the **AWS** Management Console
* Why amazon rds is used if anything happen to your local database the values and the document may get crashed and cannot able to recover in rds we can use snapshots and simple storage services called s3 where we can store the values will be safe
* How to open rds?
* In concole go to rds select engine in which it shows many engine example: mysql, oracle, mariaDB,etc choose which you want to work.
* And next it ask specify DB details by default is has 20Gb of storage
* And then db engine,licence model,db engine version are default
* Db instance class dbt2 micro free of cost and 1gb ram
* It shows settings master user which is important and than password and confirm password
* And next configure advance setting in which put database nameit shows backup option it default takes 7 days
* And main thing is keeping database publically assessable then launch
* Note the end point which is used to connect our mysql with aws rds
* Go to my sql new connection in which mysql address give end point which connects the mysql with aws rds
* It ask new database name give as per master user in aws as u give earlier and password and then click ok ‘
* It connects the mysql with aws rds and we can create table and done as per owr modification

**AMAZON S3**

* It is defined as the Amazon s3 is simple storage services
* Amazon Simple Storage Service is storage for the Internet. It is designed to make web-scale computing easier for developers. Amazon **S3** has a simple web services interface that you can **use** to store and retrieve any amount of data, at any time, from anywhere on the web.
* It is easy to create s3 in aws just want to create a bucket to create folder and upload files
* Go to console select s3
* And then it shows the icon of creating bucket
* Click that and create new one but the bucket name should be unique within the other buckets
* After creating we can upload any files videos images which can assessable to public as well as private aslo.
* It is an simple storage unit
* To keep our files safe and that file should use by particular person we can create assess key id and assess security key which which useful for security purpose
* This can be done by help of IAM in AWS it is the one of services in AWS for security purpose

**AMAZON IAM**

* The IAM is defined as the identity and access management
* **AWS** Identity and Access Management (**IAM**) is a web service that helps you securely control access to **AWS** resources. You use **IAM** to control who is authenticated (signed in) and authorized (has permissions) to use resources.
* Resources means varies services
* To create go to console
* Go to user in which add user create user name
* Next access type programmatic
* Allways create group to user and add user to group
* And then filter polices in which the user want to read only, user want to write only or full access can be set as per setting it will work
* And then get success and it shows access key and access id
* To be noted and while entering storage it ask the user to put access key and secrat id.

**AWS LAMBDA**

It is an AWS compute Domain

* In aws Lambda No servers to maintain serverless supports code in C#, java, python, nodeJs ang go
* Offers many levels of resources
* 128 MB of memory and lowest CPU power , to 3008 MB of memory and highest CPU
* Functions can own between 100ms to 5min in length
* Pricing depends on

1. Resources chosen
2. Time of execution
3. Supports IAM and VPC type

* When should use Lambda instead of EC2?
* For standalone (stateless) code execution which gets executed and then stores the result somewhere in database or s3.
* When you don’t want to maintain server (os update, security, scalability etc).
* How to work on it
* Sign in with aws and go to lambda create function give name runtime and role
* It is called as automated version of EC2
* AWS Lambda as a serverless compute service meaning

**AWS SQS AND SES (Simple queue services and simple email services)**

**Simple Queue Services**

There are two types of queue:

* Fifo queue
* Configure queue upto 2 min

**What is sqs?**

* **Amazon** Simple Queue Service (**Amazon SQS**) is a pay-per-use web service for storing messages in transit between computers.
* Developers use **SQS** to build distributed applications with decoupled components without having to deal with the overhead of creating and maintaining message queues.
* Amazon Simple Queue Service (**SQS**) is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications.
* Get started with **SQS** in minutes using the **AWS** console, Command Line Interface or SDK of your choice, and three simple commands
* Using sqs we can send store and rescue messages between software components any value without lookinh message or requiring other services to always available
* To configure sqs
* Create queue
* Send message
* Pull queue
* Retrieve message

**How to do it?**

* Go to sqs get started give an queue name

**Queue attributes**

* Default visibility timeout
* Maximum message services
* Delivery delay( they are customize we can change anything)

**What is SES?**

* **Amazon** Simple Email Service (**Amazon SES**) is a highly scalable and cost-effective platform for sending and receiving email.
* **Amazon SES** eliminates the complexity and expense of building an in-house email solution or licensing, installing, and operating a third-party email solution.
* Amazon Simple Email Service (**SES**) is a cost-effective email service built on the reliable and scalable infrastructure that Amazon.com developed to serve its own customer base.
* With Amazon **SES**, you can send transactional email, marketing messages, or any other type of high-quality content to your customers
* SES works about want to give email id and then the configuration message get through email want to confirm that the services begin
* And then we can customize that if any file uploaded in s3 can get the email verification or else an message routing system.

**AWS CLOUDFRONT**

What is AWS cloudfront?

Amazon **CloudFront** is a content delivery network (CDN) offered by Amazon Web Services. Content delivery networks provide a globally-distributed network of proxy servers which cache content, such as web videos or other bulky media, more locally to consumers, thus improving access speed for downloading the content

**Laas** – infrastructure as a services

Laas provides a virtual computing resources over the internet Eg: EC2

**Paas** – platform as a services in a category of cloud computing services that provides a platform allowing services that provides a platform customer to develop own and manage application without the complexity of building and maintaining eg : elastic beanstalk

Amazon cloudfront is a web services that speeds up distribution of your static and dynamic web content such as HTML, css, javascript and image files

**How does aws cloudfront deliver content?**

At first cloudfront sends request to edge location it is near to client request thus the s3 and EC2 may be in somewhere country as far as it has. So it has edge location it checks it chache(request) if the data is there in edge location means it gives response quickly and process is very fast and so if there is no information in edge location means means it moves to its originate place and fetch the data that want to display in client machine

Thus the edge location is used to frequently used data transfer mode if the data is not used frequently it gets deleted automatically.

**APPLICATION**

* Accelerate static websites content delivery
* Serve on demand on live stream video
* Encrypt specific field throughout system processing

* Customize at the edge
* Serve private content by using lambda @ edge customization

**AWS CLOUDFRONT DISTRIBUTION**

* We can take existing webpage or source or else create new bucket in s3
* And then upload the file content want to be in server
* And then go to cloudfront go to web distributor origin domain name
* Origin path
* And then make restrict bucket id
* Create new identity
* Chache browser http and https.
* And then copy url and search in search engine it shows the website that u want to know about it

**AWS VPC (Virtual Private Cloud)**

**Key Concept:**

* A virtual private cloud is the sub cloud inside the AWS public cloud. Sub cloud means it is inside the on isolated network.
* Other services cannot see the instance that are inside the vpc you can launch your aws resources such as amazon EC2 instances into your vpc
* You can configure your vpc select its ipaddress range create subnets and configure the routes tables, network, gateway and security settings
* A subnet is the range of ipaddress in your vpc you can launch your vpc we can lauch aws resources into a subnet for resources that must be connected to the internet and a private subnet for the resources that wont connect to the internet.
* What is vpc?
* Amazon Virtual Private Cloud is a commercial cloud computing service that provides users a virtual private cloud, by "provision a logically isolated section of Amazon Web Services Cloud". Enterprise customers are able to access the Amazon Elastic Compute Cloud over an IPsec based virtual private network
* **Security Group**
* Act as a firewall for associating amazon EC2 instance, controlling both inbound and outbound traffic and instance level - static
* **Network access control List (NACL)**
* Act as a firewall as associate subnet controlling both inbound and outbound traffic – non static
* **Internet gateway**
* Is the horizontal scaled redundant that allows communication between instance in your vpc and internet in it therefore imposes no availability risk or bandwidth constrains on your network traffic.
* **Nat device**
* Is to enable instances in a private subnets to connect to internet ( for example software updates or other aws services but present internet for initiating connecting instance…
* For that create new vpc in vpc and and give connection over s3
* Create bucket in s3 and add folder and add what ever you want to upload and then it will give you a https link using that we can enter in url and then we get the value uploaded in the s3.