COMSATS University, Islamabad.
Department of Computer Science

Class: BCT VII **Submission Date: 17-02-26**
Subject: CSC418 - DevOps and Cloud Computing
Instructor: Dr. Muhammad Imran **Marks: 10**
[CLO4]: Apply DevOps pipeline automation techniques for code deployment.

Lab Task

Students have to deploy a cloud-based Employee Management web application using AWS EC2, Amazon RDS (MySQL), and Amazon S3. Students will learn how to integrate compute, database, storage, and IAM services to host a real-world application on AWS.

1. Launch and configure an EC2 instance.
2. Create and configure an Amazon RDS MySQL database.
3. Create and use an S3 bucket.
4. Connect EC2 with RDS securely.
5. Clone the Employee portal code provided on https://github.com/hshar94/aws-live.git
6. Deploy a Python Flask application on EC2.
7. Configure IAM roles for service access.

Solution:

1. Create EC2 machine.
2. Create RDS database.
   a. Choose a database creation method->Easy Create
   b. Engine options->MySql
   c. Templates->Free-tier
   d. Settings
      i. ->DB instance identifier:employee.
      ii. ->Credentials Settings
         1. ->Master username:admin;
         2. ->Master password:adminadmin,
         3. ->Confirm master password:adminadmin
   e. Set up EC2 connection - optional ->Connect to an EC2 compute resource.
   f. In the EC2 instance dropdown-> select EC2 instance.
   g. Click on create Database button.

3. Create S3 bucket->addemployee
4. Connect to EC2 machine.
   a. sudo apt update
   b. sudo apt install mysql-client
   c. mysql -h employee.cvlrhiclnrl4.us-east-1.rds.amazonaws.com -u admin -p employee
   d. mysql> show databases;
   e. mysql> create database employee;
   f. mysql> use employee;

Class: BCT VII **Submission Date: 17-02-26**
Subject: CSC418 - DevOps and Cloud Computing
Instructor: Dr. Muhammad Imran **Marks: 10**
[CLO4]: Apply DevOps pipeline automation techniques for code deployment.

Lab Task

g. mysql> create table employee(
h. -> empid varchar(20),
i. -> fname varchar(20),
j. -> lname varchar(20),
k. -> pri_skill varchar(20),
l. -> location varchar(20));
m. mysql-> show tables;
n. mysql-> exit

5. ubuntu> git clone https://github.com/hshar94/aws-live.git

6. ubuntu> cd aws-live
7. ubuntu/aws-live> nano config.py
   a. customhost = "employee.cvlrhiclnrl4.us-east-1.rds.amazonaws.com"
   b. customuser = "admin"
   c. custompass = "adminadmin"
   d. customdb = "employee"
   e. custombucket = "addemployee"
   f. customregion = "us-east-1"

8. ubuntu/aws-live>sudo apt-get install python3
9. ubuntu/aws-live>sudo apt-get install python3-flask
10. ubuntu/aws-live>sudo apt-get install python3-pymysql
11. ubuntu/aws-live>sudo apt-get install python3-boto3

12. ubuntu/aws-live>sudo python3 EmpApp.py

13. In IAM service, create a role (EC2S3role) for EC2 for AdministrativeAccess
14. In IAM service
    a. Access Management-> Roles->CreateRole->AWS Service
    b. ->Service or usecase->EC2->Next
    c. ->Check Administrative Access->Next
    d. ->Rolename-> EC2S3role->Click on createRole button.
15. Go to EC2 service.
16. Inside Actions->Security-ModifyIAMRole
17. Select EC2S3role->Click on UpdateRole button.