

Shikshana Prasarak Mandali's
SIR PARASHURAMBHAU COLLEGE
(Empowered Autonomous)
Department of Computer Science
Lab-Book
M. Sc. (Computer Science) I Semester I
Lab Course (MSCOSDSE304) – Cloud Computing
(Academic year 2024-25)

Name: _____

College Name: Sir Parashurambhau College, pune

Roll No.: _____ Division: _____

Academic Year: 2024-2025



Shikshana Prasarak Mandali's

SIR PARASHURAMBHAU COLLEGE

(Autonomous)

TILAK ROAD, PUNE – 411030

Department of Computer Science

Lab Course (MSCOSDSE304) – Cloud Computing

Certificate

*This is to certify that Mr./Ms. _____ of
M.Sc. (Computer Science) has completed _____ practical's out of 07 in the
subject Lab Course (MSCOSDSE304) – Cloud Computing
Semester – I during the academic year 2024-2025.*

Teacher In-Charge

Head

Date:

Department of Computer Science

Examiner 1

Examiner 2

Assignment Completion Sheet

	Assignments based on Cloud Computing		
Sr. No.	Assignment Name	Date	Marks
1	Ec2 Compute service.		
2	Apache Tomcat & Node.js.		
3	Elastic Block Storage & Simple Storage Service.		
4	Elastic Load Balancer & Identity and Access Management.		
5	Relational Database services & DynamoDB		
6	Jenkins & Lambda Serverless Functions		
7	Docker		
Marks (out of 35)			
Marks (out of 15)			
Sign of Teacher In-Charge :			

Assignment 1: EC2 compute service

- 1) Create an EC2 instance using the Ubuntu Linux AMI with a storage capacity of up to 10 GiB.
- 2) Edit user data of above instance & Specify user data to include a command script that will display the instance's public IPv4 address & availability zone upon launch.
- 3) Launch an EC2 instance with the Ubuntu AMI and install the latest Long-Term Support (LTS) version of Java using the appropriate commands.
- 4) Write a Python code to read the 'city.txt' file and search for a city name using any searching algorithm.

Signature of the instructor

Date: ____/____/____

Assignment Evaluation0: Not done ☐2: Late Complete ☐4: Complete ☐1: Incomplete ☐3: Needs improvement ☐5: Well Done ☐

Assignment 2: Apache Tomcat & Node.js

- 1) Install Apache Tomcat LTS version on an instance with Ubuntu AMI and create a form with the specifications provided in the attached image.

The image shows a web form titled "Personal Details" enclosed in a light gray border. Inside the form, there are several input fields and a submit button. The fields are: "Salutation" with a dropdown menu showing "--None--"; "First name:" with a text input field; "Last name:" with a text input field; "Gender :" with two radio buttons labeled "Male" and "Female"; "Email:" with a text input field; "Date of Birth:" with a date picker showing "mm/dd/yyyy" and a calendar icon; and "Address :" with a larger text input field. At the bottom left of the form is a "Submit" button.

- 2) Write a JavaScript code that accepts the file name as input and prints the contents of the file on a web page. (Use NodeJs)

Signature of the instructor

Date: ____/____/____

Assignment Evaluation

0: Not done 2: Late Complete 4: Complete

1: Incomplete 3: Needs improvement 5: Well Done

Assignment 3: Elastic block storage & Simple Storage Service

- 1) Create a volume of 10 GiB and attach it to an EC2 instance.
- 2) After creating a volume of 8 GiB, attaching it to an EC2 instance, making a file system on the volume, and creating a mount point directory, proceed to mount it.
- 3) Create an S3 bucket containing objects, specifically images, and configure it to be publicly accessible. (Note: bucket should not be public.)
- 4) Upload a video to the same S3 bucket and create an HTML document with the necessary markup to embed and display that video.

Signature of the instructor

Date: ____/____/____

Assignment Evaluation0: Not done ☐2: Late Complete ☐4: Complete ☐1: Incomplete ☐3: Needs improvement ☐5: Well Done ☐

Assignment 4: Elastic Load Balancer & Identity and Access Management

- 1) Create an application load balancer with a security group allowing inbound rules for HTTP, and add two instances to the target group. Use the proper shell script in user data to display the instances' public IP addresses and availability zones.
- 2) IAM
 - I. Create user IAM1.
 - II. Create Group Dev with Policies Ec2 full access.
 - III. Create user IAM2 with Policies full admin access & S3 full access.
 - IV. Add user IAM1 to the group Dev.
 - V. Create group Tester.
 - VI. Create user IAM3 & copy Policies of IAM2.
 - VII. Add IAM3 to group Tester.
 - VIII. Set up extra security for IAM1 using Multi-Factor Authentication (MFA).(Create proper Account alias to access all IAM Users).

Signature of the instructor

Date: ____/____/____

Assignment Evaluation

0: Not done <input type="checkbox"/>	2: Late Complete <input type="checkbox"/>	4: Complete <input type="checkbox"/>
1: Incomplete <input type="checkbox"/>	3: Needs improvement <input type="checkbox"/>	5: Well Done <input type="checkbox"/>

Assignment 5: Relational Database Services & DynamoDB

- 1) Establish a database with the MySQL engine type and the Free Tier template, including the initial database. Subsequently, connect to the database using mysql connection string with the necessary credentials.
 - a. Consider the following Entities and Relationships and create a RDB in 3NF.
 Student (sno int, sname varchar (20), class varchar (10), marks float)
 Teacher (tno int, tname varchar (20), specialization varchar (20))

Table level constraint: Primary key, sname should be not null.

Add column salary to teacher table.

- b. Insert appropriate data and construct queries in MySQL.
 - I. Delete the record of teacher “Mrs. Riya”.
 - II. Find student who got specialization in “AI”.
- 2) Create a table in the DynamoDB database, specifying the appropriate partition key during its creation, and include items in it.

Signature of the instructor

Date: ____/____/____

Assignment Evaluation

0: Not done ☐ 2: Late Complete ☐ 4: Complete ☐

1: Incomplete ☐ 3: Needs improvement ☐ 5: Well Done ☐

Assignment 6: Jenkins & Lambda Serverless Functions

- 1) Launch an EC2 instance, install and configure Jenkins, and ensure it is running on port number 8080 by modifying the inbound rules of the respective instance's security group. Then, establish a freestyle project to check JDK version.
- 2) Write a lambda function that takes an array and the element to search as input through the payload, performs a binary search on it, and returns the appropriate response. (use AWS Lambda natively supported language)

Signature of the instructor

Date: ____ / ____ / ____

Assignment Evaluation

0: Not done <input type="checkbox"/>	2: Late Complete <input type="checkbox"/>	4: Complete <input type="checkbox"/>
1: Incomplete <input type="checkbox"/>	3: Needs improvement <input type="checkbox"/>	5: Well Done <input type="checkbox"/>

Assignment 7: Docker

1) Set 1

- A) Pull the "hello-world" image from Docker Hub, display its details using the "Docker images" command, and then run a container of it.
- B) Pull the 'Debian' image from Docker Hub, display its details using the 'Docker image ls' command, and then run a container of it.
- C) Pull the 'redhat/ubi8' image from Docker Hub, display its details using the 'Docker image ls' command, and then run a container of it.
- D) Create a custom Docker image by using Alpine as the base image in a Dockerfile. In the Dockerfile, include a command to print 'hello world.' Build the image using the command 'docker build -t my-custom-image' and then run a container from the built image.

2) Set 2

- A) To build a Docker image with Ubuntu as the base, use the following instruction: "FROM ubuntu". Additionally, set the default command to launch an interactive Bash shell when a container is started: "CMD ["/bin/bash"]".
- B) Update the Dockerfile from the previous instructions (Q2. A) in another folder to be compatible with Alpine Linux. Set the default command to launch an interactive Bash shell when a container starts and install OpenJDK 8 using the Bash shell.
- C) Run alpine container interactively using docker exec command.

3) Set 3

- A) Create a Dockerfile with the official Python 3.8 runtime as the base image also it should run script.py file when container starts.
- B) Create a Dockerfile with the official Ubuntu as the base image, write proper commands within Dockerfile to install Openjdk8 Build the Docker image then run container from image.
- C) Push any image to Docker hub with proper steps.

4) Set 4

- A) Containerize java web application running on apache tomcat server (proper file structure is mandatory).

