



Gokhale Education Society's  
**R. H. Sapat College of Engineering, Management Studies & Research,**  
**Nashik-422005**  
**Department of Computer Engineering A.Y.2021-22**

**R.H.SAPAT COLLEGE OF ENGINEERING, MANAGEMENT  
& RESEARCH, NASHIK**

**DEPARTMENT OF COMPUTER ENGINEERING**

SUBJECT CODE: 310258

# **LAB MANUAL**

## **Laboratory Practice-II**

### **(Artificial Intelligence & Cloud Computing)**

**Semester – II, Academic Year 2021-2022**

**Subject Teachers:-**

**Artificial Intelligence: - Dr Neeta Deshpande and Mr. Rahul Chakre**

**Cloud Computing: - Dr.S.R.Jadhao and Mrs.Vrushali Nikam**

<b>Department:</b> Computer Engineering	<b>Class:</b> TE-A Batch:TA1,TA2,TA3,TA4 TE-B Batch:TB1,TB2,TB3,TB4
<b>Subject Name:</b> Laboratory Practice-II <b>Academic Year:</b> 2021 -22	<b>Subject Code:</b> 310258

<b>Teaching Scheme:-</b>	<b>Pract /week:4 Hrs</b>	
<b>Examination Scheme:-</b>	<b>Practical: 25 Marks</b>	<b>Term-Work:50 Marks</b>
<b>Name of Faculty:</b>	<b>Artificial Intelligence:-Dr. Neeta Deshpande and Mr. Rahul Chakre Cloud Computing :-Dr.S.R.Jadhao and Mrs. Vrushali Nikam</b>	

### **Course Objectives:**

- To learn and apply various search strategies for AI
- To Formalize and implement constraints in search problems
- To understand the concepts of Information Security / Augmented and VirtualReality/Cloud Computing/Software Modeling and Architectures

### **Course Outcomes:**

<b>CO</b>	<b>Statements</b>	<b>Cognitive levelof learning</b>
<b>C318.1</b>	Design a system using different informed search / uninformed search or heuristic approaches	(Design)
<b>C318.2</b>	Apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning	(Apply)
<b>C318.3</b>	Design and develop an interactive AI application	(Apply)
<b>C318.4</b>	Use tools and techniques in the area of Cloud Computing	(Apply)
<b>C318.5</b>	Use cloud computing services for problem solving	(Apply)
<b>C318.6</b>	Design and develop applications on cloud	(Apply)

### **List of Laboratory Assignments**

<b>Sr. No.</b>	<b>Group</b>	<b>Page No.</b>	<b>CO</b>
1	Implement depth first search algorithm and Breadth First Search algorithm, Use an undirected graph and develop a recursive algorithm for searching all the vertices of a Graph or tree data structure.	3	C318.1
2	Implement A star Algorithm for any game search problem	7	C318.1

3	Implement Greedy search algorithm for any of the following application: I. Selection Sort II. Minimum Spanning Tree III. Single-Source Shortest Path Problem IV. Job Scheduling Problem V. Prim's Minimal Spanning Tree Algorithm VI. Kruskal's Minimal Spanning Tree Algorithm Dijkstra's Minimal Spanning Tree Algorithm	12	C318.1
4	Implement a solution for a Constraint Satisfaction Problem using Branch and Bound and Backtracking for n-queens problem or a graph coloring problem.	16	C318.2
5	Develop an elementary chatbot for any suitable customer interaction application.	19	C318.3
6	Implement any one of the following Expert System I. Information management II. Hospitals and medical facilities III. Help desks management IV. Employee performance evaluation Stock market OR Design a Miniproject on any one of the following topics not limited to:- <ul style="list-style-type: none"><li>• Airline scheduling and cargo schedules</li><li>• Hospital Management system</li><li>• Ticket booking system</li><li>• Food Ordering System</li><li>• Task management system</li></ul>	21	C318.3

## Assignment No.:01

### Problem Statement:

Implement depth first search algorithm and Breadth First Search algorithm, Use an undirected graph and develop a recursive algorithm for searching all the vertices of a graph or tree data structure.

### Objectives:

1. To study the various Search Algorithms.
2. To study the depth first search algorithm.
3. To study the breadth first search algorithm.

### Theory:

#### Uninformed Search Algorithms:

The search algorithms in this section have no additional information on the goal node other than the one

provided in the problem definition. The plans to reach the goal state from the start state differ only by the order and/or length of actions. Uninformed search is also called **Blind search**. The following uninformed search algorithms are discussed in this section.

1. Depth First Search
2. Breadth First Search
3. Uniform Cost Search

Each of these algorithms will have:

- A problem **graph**, containing the start node S and the goal node G.
- A **strategy**, describing the manner in which the graph will be traversed to get to G.
- A **fringe**, which is a data structure used to store all the possible states (nodes) that can be reached from the current states.
- A **tree**, that results while traversing to the goal node.
- A solution **plan**, which is the sequence of nodes from S to G.

### **Depth First Search:**

Depth-first search (DFS) is an algorithm for traversing or searching tree or graph data structures. The algorithm starts at the root node (selecting some arbitrary node as the root node in the case of a graph) and explores as far as possible along each branch before backtracking.

#### **Performance Measure:**

$d = \text{the depth of the search tree} = \text{the number of levels of the search tree}$ .  $n^i = \text{number of nodes in level } i$ .

**Time complexity:** Equivalent to the number of nodes traversed in DFS.  $T(n) = 1 + n^2 + n^3 + n^4 + \dots + n^d = O(n^d)$

**Space complexity:** Equivalent to how large can the fringe get.  $S(n) = O(n^*d)$

**Completeness:** DFS is complete if the search tree is finite, meaning for a given finite search tree, DFS will come up with a solution if it exists.

**Optimality:** DFS is not optimal, meaning the number of steps in reaching the solution, or the cost spent in reaching it is high.

### **Breadth First Search:**

Breadth-first search (BFS) is an algorithm for traversing or searching tree or graph data structures. It starts at the tree root (or some arbitrary node of a graph, sometimes referred to as a ‘search key’), and explores all of the neighbor nodes at the present depth prior to moving on to the nodes at the next depth level.

$d = \text{the depth of the shallowest solution}$ .  $n^i = \text{number of nodes in level } i$ .

**Time complexity:** Equivalent to the number of nodes traversed in BFS until the shallowest solution.

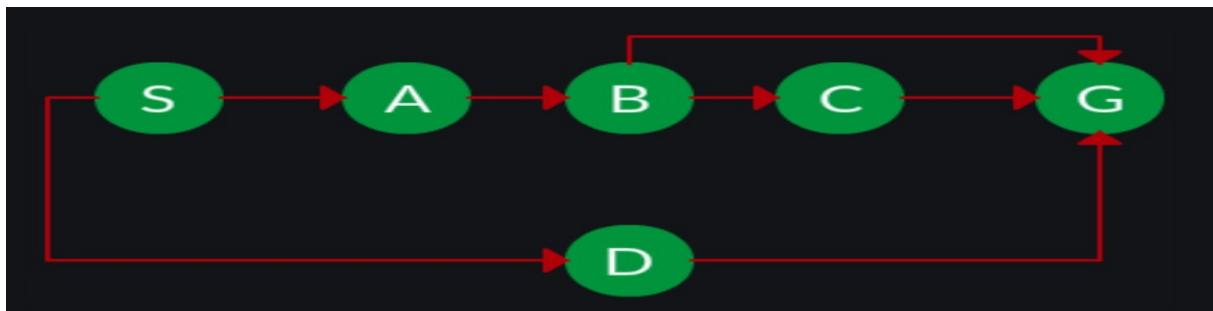
$T(n) = 1 + n^2 + n^3 + n^4 + \dots + n^d = O(n^d)$

**Space complexity:** Equivalent to how large can the fringe get.  $S(n) = O(n^d)$

**Completeness:** BFS is complete, meaning for a given search tree, BFS will come up with a solution if it

exists.

**Example** Find path to move from node S to node G using DFS and BFS.



DFS

BFS

jawa

### **Algorithm/Flowchart:**

**DFS:**

- **Step 1** – Push a starting node on stack, mark it visited.
- **Step 2** - Visit the adjacent unvisited vertex of start node. Mark it as visited. Display it. Push it in a stack.
- **Step 3** – If no adjacent vertex is found, pop up a vertex from the stack. Repeat Step 2
- **Step 4** – Repeat Step 2 and Step 3 until the stack is empty.

**BFS**

- **Step 1** – Insert start node in Queue, mark it visited.
  - **Step 2** – Visit the adjacent unvisited vertex. Mark it as visited. Display it. Insert it in a queue.
  - **Step 3** – If no adjacent vertex is found, remove the first vertex from the queue.
- Step 4** – Repeat Step 3 and Step 4 until the queue is empty.

### **Input:**

Graph: no of nodes, no of edges

$n = 4, e = 6$

Enter adjacent node information

$0 \rightarrow 1, 0 \rightarrow 2, 1 \rightarrow 2, 2 \rightarrow 0, 2 \rightarrow 3, 3 \rightarrow 3$

### **Output:**

**DFS from vertex 2 – 2, 0, 1, 3**

**BFS from vertex 2 – 2, 0, 3, 1**

## **Software Requirement:**

1. Python3/Java ... **GOOGLE COLAB**

Code Editor or IDE – Eclipse/ IntelliJ IDEA/ VSCode

## **Frequently Asked Questions:**

1. What are the commonly used uninformed search algorithms?
2. Which data structure is used in BFS?
3. Which data structure is used in DFS?

What is the performance measure for DFS and BFS?

## **Conclusion:**

We have successfully implemented depth first search and breadth first search algorithm for a graph

## **Assignment No.:02**

### **Problem Statement:**

Implement A star Algorithm for any game search problem.

### **Objectives:**

1. To study various Search Algorithms
2. To study the A\* Algorithm

### **Theory:**

**A\* Algorithm** is the advanced form of the BFS algorithm (Breadth-first search), which searches for the shorter path first than, the longer paths. It is a **complete** as well as an **optimal** solution for solving path and grid problems.

The key feature of the A\* algorithm is that it keeps a track of each visited node which helps in ignoring the nodes that are already visited, saving a huge amount of time. It also has a list that holds all the nodes that are left to be explored and it chooses the most optimal node from this list, thus saving time not exploring unnecessary or less optimal nodes.

So we use two lists namely ‘open list’ and ‘closed list’ the open list contains all the nodes that are being generated and are not existing in the closed list and each node explored after its neighboring nodes are discovered is put in the closed list and the neighbors are put in the open list this is how the nodes expand. Each node has a pointer to its parent so that at any given point it can retrace the path to the parent.

Initially, the open list holds the start(Initial) node. The next node chosen from the open list is based on its **f score (f(n))**, the node with the 1

Heuristic used in A\* Where

$g(n)$  : The actual cost path fro

$$f(n) = g(n) + h(n) \quad n :$$

The actual cost path from the current node to goal node.

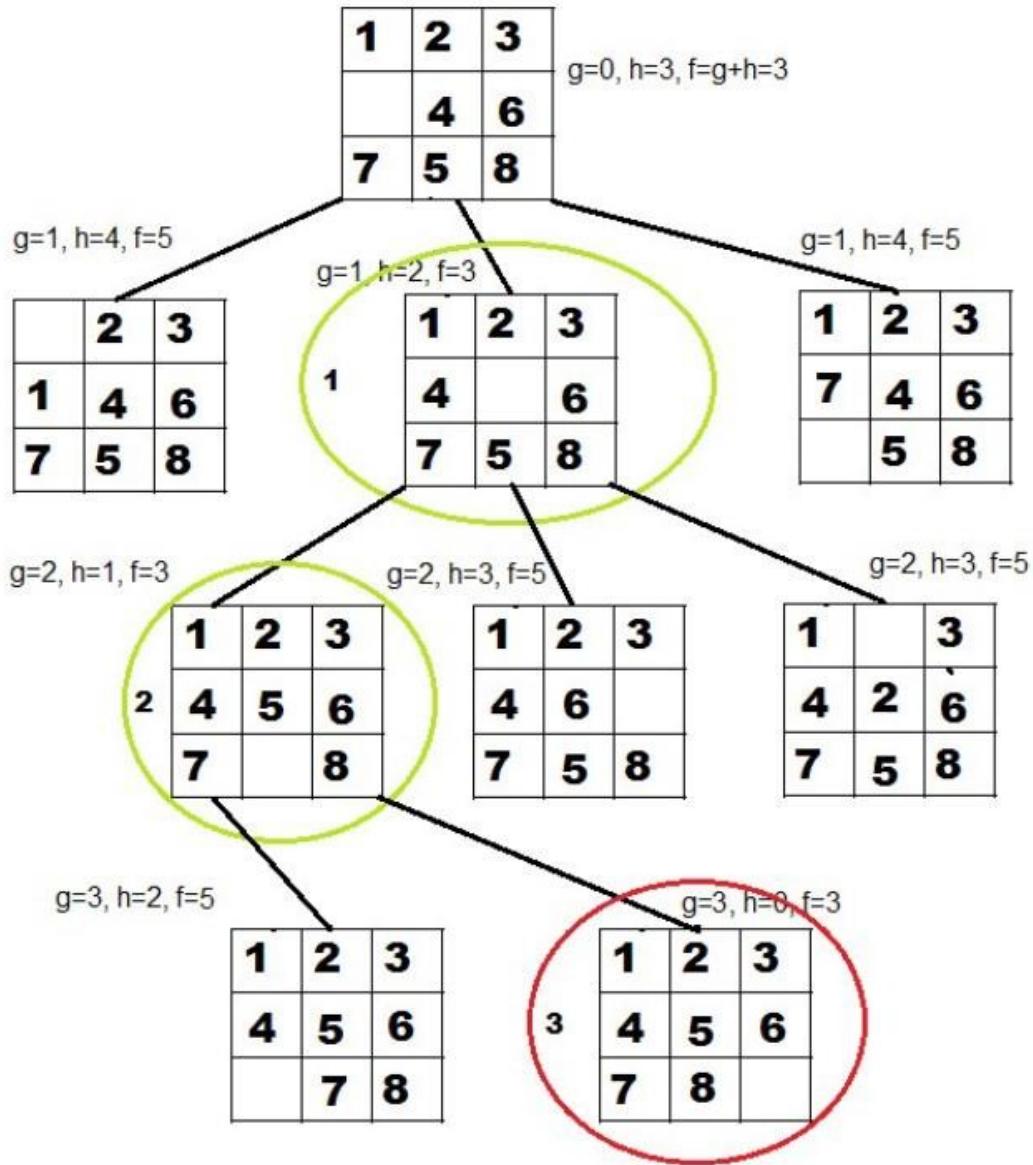
$f(n)$  : The actual cost path from the start node to the goal node.

Performance Measure:

**Optimal** – find the least cost from the starting point to the ending point.

**Complete** – It means that it will find all the available paths from start to end.

A\* uses a combination of heuristic value (h-score: how far the goal node is) as well as the g-score(i.e. the number of nodes traversed from the start node to current node).



selected tiles by comparing misplaced nodes.g-node.

goal state and counting

ode to the currentnode

put the starting node on the open list

(you can leave its f at zero)

3. while the open list is not empty
  - a) find the node with the least f on the open list, call it "q"
  - b) pop q off the open list
  - c) generate q's 8 successors and set their parents to q
  - d) for each successor
    - i) if successor is the goal, stop search
    - ii) else, compute both g and h for successor  
successor.g = q.g + distance between

```

successor and q
successor.h = distance from goal to
successor (This can be done using many ways, we
will discuss three heuristics-Manhattan,
Diagonal and Euclidean Heuristics)

successor.f = successor.g + successor.h

iii) if a node with the same position as successor
is in the OPEN list which has a lower f than
successor, skip this successor

iv) if a node with the same position as
Successor is in the CLOSED list which has a
lower f than successor, skip this successor
otherwise, add the node to the open list
end (for loop)

e) push q on the closed listend (while loop)

```

**Input:**

Start State:

1 2 3  
 $\underline{4}$  6  
7 5 8

Goal State:

1 2 3  
4 5 6  
7 8  $\underline{\quad}$

**Frequently Asked Questions:**

1. What is the heuristic used in A\* Algorithm?
2. Explain the performance measures of A\* Algorithm.
3. List applications of A\* Algorithms.
4. What are Informed Search Algorithms? Name the commonly used Informed Search Algorithms.

**Conclusion:**

Successfully implemented A\* Algorithm for 8-Puzzles Problem

## Assignment No.:03

### **Problem Statement:**

#### **Implement Greedy search algorithm for any of the following application:**

- VII. Selection Sort
- VIII. Minimum Spanning Tree
- IX. Single-Source Shortest Path Problem
- X. Job Scheduling Problem
- XI. Prim's Minimal Spanning Tree Algorithm

Kruskal's Minimal Spanning Tree Algorithm Dijkstra's Minimal Spanning Tree Algorithm

### **Objectives:**

To study various Greedy Search Algorithms

### **Theory:**

Greedy is an algorithmic paradigm that builds up a solution piece by piece, always choosing the next piece that offers the most obvious and immediate benefit. So the problems where choosing locally optimal also leads to global solution are best fit for Greedy.

The greedy method is one of the strategies like Divide and conquer used to solve the problems. This method is used for solving optimization problems. An optimization problem is a problem that demands either maximum or minimum results. Let's understand through some terms.

The Greedy method is the simplest and straightforward approach. It is not an algorithm, but it is a technique. The main function of this approach is that the decision is taken on the basis of the currently available information. Whatever the current information is present, the decision is made without worrying about the effect of the current decision in future.

This technique is basically used to determine the feasible solution that may or may not be optimal. The feasible solution is a subset that satisfies the given criteria. The optimal solution is the solution which is the best and the most favorable solution in the subset. In the case of feasible, if more than one solution satisfies the given criteria then those solutions will be considered as the feasible, whereas the optimal solution is the best solution among all the solutions.

### **Characteristics of Greedy method**

The following are the characteristics of a greedy method:

- o construct the solution in an optimal way, this algorithm creates two sets where one set contains all the chosen items, and another set contains the rejected items.
- o A Greedy algorithm makes good local choices in the hope that the solution should be either feasible or optimal.

### **Components of Greedy Algorithm**

The components that can be used in the greedy algorithm are:

- o Candidate set: A solution that is created from the set is known as a candidate set.

Selection function: This function is used to choose the candidate or subset which can be added in the solution.

- o Feasibility function: A function that is used to determine whether the candidate or

subset can be used to contribute to the solution or not.

- Objective function: A function is used to assign the value to the solution or the partial solution.
- Solution function: This function is used to intimate whether the complete function has been reached or not.

### Applications of Greedy Algorithm

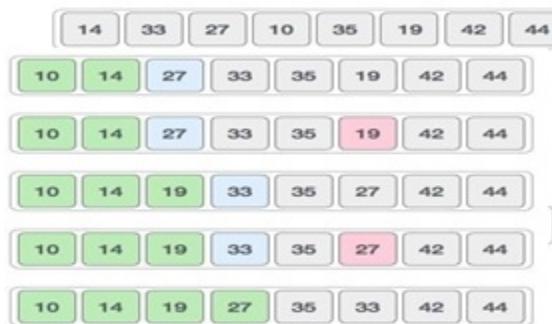
- It is used in finding the shortest path.
- It is used to find the minimum spanning tree using the prim's algorithm or the Kruskal's algorithm.
- It is used in a job sequencing with a deadline.

Selection sort is a simple sorting algorithm. This sorting algorithm is an in-place comparison-based algorithm in which the list is divided into two parts, the sorted part at the left end and the unsorted part at the right end. Initially, the sorted part is empty and the unsorted part is the entire list.

The smallest element is selected from the unsorted array and swapped with the leftmost element, and that element becomes a part of the sorted array. This process continues moving unsorted array boundary by one element to the right.

This algorithm is not suitable for large data sets as its average and worst case complexities are of  $O(n^2)$ , where  $n$  is the number of items. Consider the following depicted array as an example.

For the first position in the sorted list, the whole list is scanned sequentially. The first position where 14 is stored presently, we search the whole list and find that 10 is the lowest value. So we replace 14 with 10. After one iteration 10, which happens to be the minimum value in the list, appears in the first position of the sorted list. For the second position, where 33 is residing, we start scanning the rest of the list in a linear manner. We find that 14 is the second lowest value in the list and it should appear at the second place. We swap these values. After two iterations, two least values are positioned at the beginning in a sorted manner. The same process is applied to the rest of the items in the array in a linear manner.



Step 1 – Set MIN to location 0

Step 2 – Search the minimum element in the list

– Swap with value at location MIN

Step 4 – Increment MIN to point to next element

Step 5 – Repeat until list is sorted

Algorithm/Flowchart:

```
procedure selection sortlist :  
    array of items n: size of list  
for i = 1 to n - 1  
    /* set current element as minimum*/min =  
        i  
  
    /* check the element to minimum */for j =  
  
        i+1 to n  
        if list[j] < list[min] then  
            mi=j;end if  
        end for  
    /* swap the minimum element with the current element*/if  
    indexMin != i then  
        swap list[min] and list[i]end if
```

**Frequently Asked Questions:**

1. What is the Greedy Algorithm?
2. Explain the selection sort.

**Conclusion:**

Successfully implemented greedy approach for selection sort algorithm

## Assignment No.:04

### Problem Statement:

Implement a solution for a Constraint Satisfaction Problem using Branch and Bound and Backtracking for n-queens problem or a graph coloring problem.

### Objectives:

1. To study Constraint Satisfaction Problem using Branch and Bound Algorithms

### Theory:

We have seen so many techniques like Local search, Adversarial search to solve different problems. The objective of every problem-solving technique is one, i.e., to find a solution to reach the goal.

Although, in adversarial search and local search, there were no constraints on the agents while solving the problems and reaching to its solutions.

In this section, we will discuss another type of problem-solving technique known as Constraint satisfaction technique. By the name, it is understood that constraint satisfaction means *solving a problem under certain constraints or rules*.

*Constraint satisfaction is a technique where a problem is solved when its values satisfy certain constraints or rules of the problem.* Such type of technique leads to a deeper understanding of the problem structure as well as its complexity.

Constraint satisfaction depends on three components, namely:

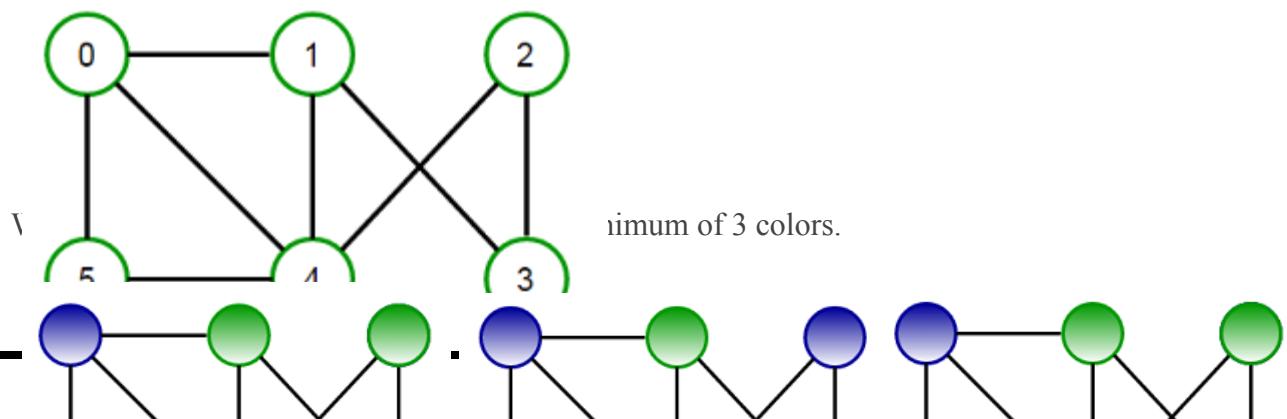
- X: It is a set of variables.
- D: It is a set of domains where the variables reside. There is a specific domain for each variable.
- C: It is a set of constraints which are followed by the set of variables.

In constraint satisfaction, domains are the spaces where the variables reside, following the problem specific constraints. These are the three main elements of a constraint satisfaction technique. The constraint value consists of a pair of {scope, rel}. The scope is a tuple of variables which participate in the constraint and rel is a relation which includes a list of values which the variables can take to satisfy the constraints of the problem.

**Graph coloring** problem involves assigning colors to certain elements of a graph subject to certain restrictions and constraints. In other words, the process of assigning colors to the vertices such that no two adjacent vertexes have the same color is called Graph Coloring. This is also known as **vertex coloring**.

**Graph coloring** (also called vertex coloring) is a way of coloring a graph's vertices such that no two adjacent vertices share the same color. This post will discuss a greedy algorithm for graph coloring and minimize the total number of colors used.

For example, consider the following graph:



Please note that we can't color the above graph using two colors. Before discussing the [greedy algorithm](#) to color graphs, let's talk about basic graph coloring terminology.

### K-colorable graph:

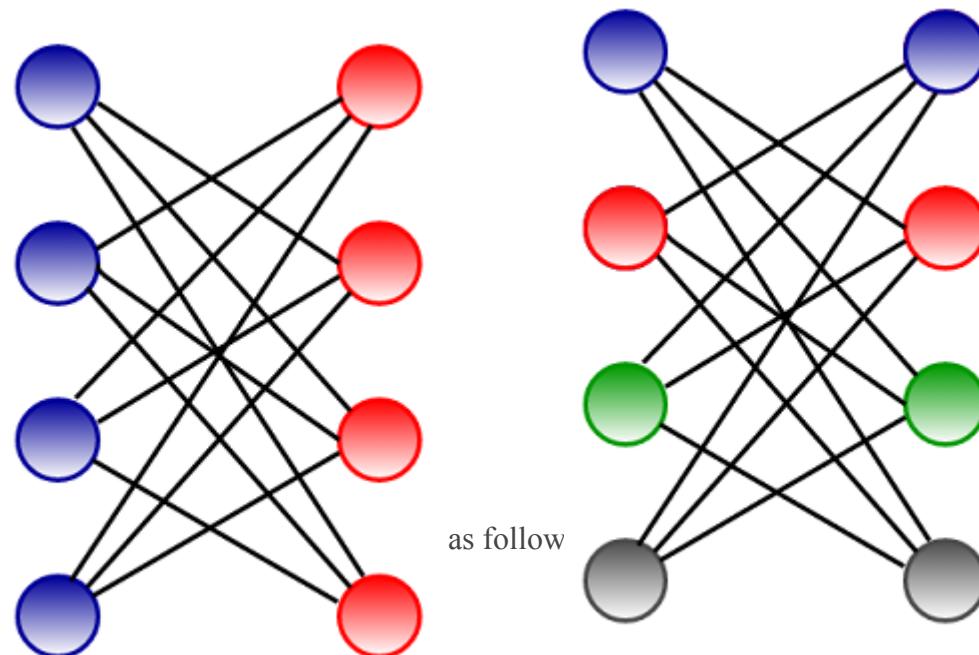
A coloring using at most  $k$  colors is called a (proper)  $k$ -coloring, and a graph that can be assigned a (proper)  $k$ -coloring is  $k$ -colorable.

### K-chromatic graph:

The smallest number of colors needed to color a graph  $G$  is called its chromatic number, and a graph that is  $k$ -chromatic if its chromatic number is exactly  $k$ .

Greedy coloring *considers the vertices of the graph in sequence and assigns each vertex its first available color*, i.e., vertices are considered in a specific order  $v_1, v_2, \dots, v_n$ , and  $v_i$  is assigned the smallest available color which is not used by any of  $v_i$ 's neighbors.

**Greedy coloring doesn't always use the minimum number of colors possible to color a graph.** For a graph of maximum degree  $x$ , greedy coloring will use at most  $x+1$  colors. Greedy coloring can be arbitrarily bad; for example, the following crown graph (a complete bipartite graph), having  $n$  vertices, can be 2-colored (refer left image), but greedy coloring resulted in  $n/2$  colors (refer right image).



### Implement N Queen's problem using Back Tracking

**Backtracking** is finding the solution of a problem whereby the solution depends on the previous steps taken. For example, in a maze problem, the solution depends on all the steps you take one-by-one. If any of those steps is wrong, then it will not lead us to the solution. In a maze problem, we first choose a path and continue moving along it. But once we understand that the particular path is incorrect, then we just come back and change it. This is what backtracking basically is.

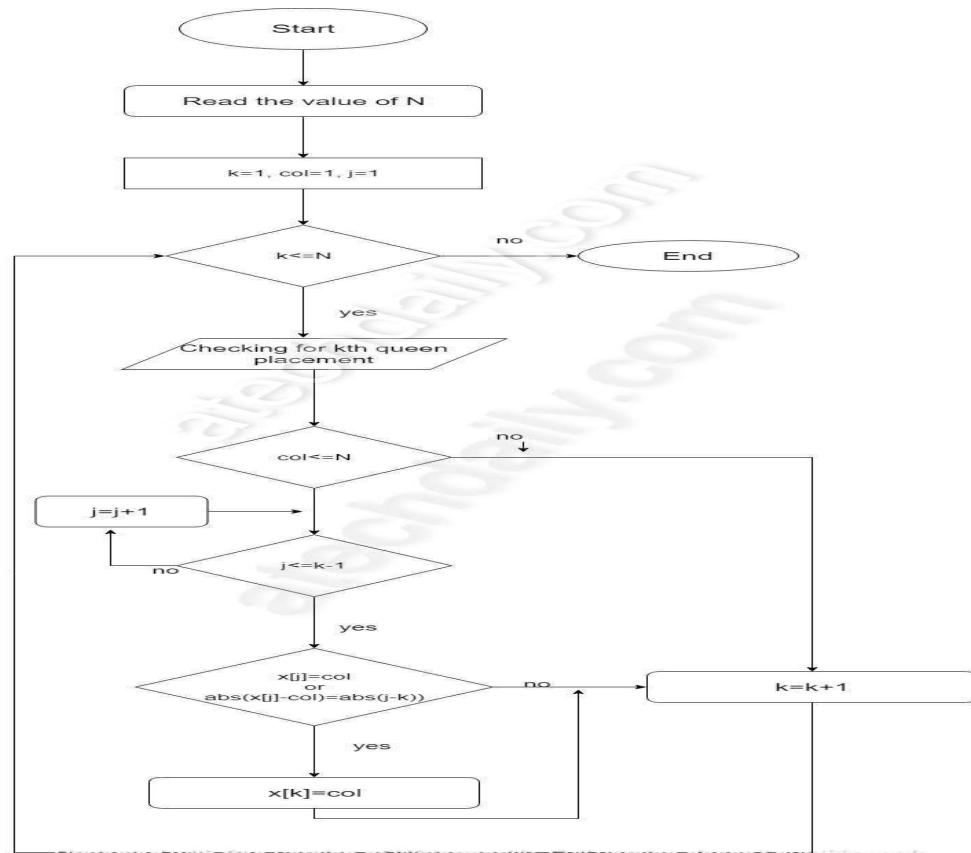
In backtracking, we first take a step and then we see if this step taken is correct or not i.e., whether it will give a correct answer or not. And if it doesn't, then we just come back and change our first step. In general, this is accomplished by recursion. Thus, in backtracking, we first start with a partial sub-solution of the problem (which may or may not lead us to the solution) and then check if we can proceed further with this sub-solution or not. If not, then we just come back and change it.

Thus, the general steps of backtracking are:

- start with a sub-solution
- check if this sub-solution will lead to the solution or not
- If not, then come back and change the sub-solution and continue again

**N Queen Problem :** N Queens Problem is a famous puzzle in which n-queens are to be placed on a  $n \times n$  chess board such that no two queens are in the same row, column or diagonal. The N Queen is the problem of placing N chess queens on an  $N \times N$  chessboard so that no two queens attack each other.

For example, following is a solution for 4 Queen problem Flowchart for N Queen problem



### Algorithm for N Queens Problem

Step 1: Start

Step 2: Given n queens, read n from user and let us denote the queen number by k.  $k=1, 2, \dots, n$ .

Step 3: We start a loop for checking if the  $k^{th}$  queen can be placed in the respective column of the  $k^{th}$  row.

Step 4: For checking that whether the queen can be placed or not, we check if the previous queens are not in diagonal or in same row with it.

Step 5: If the queen cannot be placed backtracking is done to the previous queens until a feasible solution is not found.

Step 6: Repeat the steps 3-5 until all the queens are placed.

Step 7: The column numbers of the queens are stored in an array and printed as a n-tuple solution

Step 8: Stop

### **Explanation:**

In the above algorithm,

1. For the n queen problem we take input of n, lets say n=4 so, k=1,2,3,4.
2. For placing the first queen i.e k=1, we start a loop for n columns i.e n=4 so till the fourth column.
3. The first queen can be placed at first column only.
4. Then we move for the second queen and place it seeing that the first queen is not in the same column or in diagonal with the second queen.
5. Similarly, the third queen and the fourth queen are placed. But if the fourth queen cannot be placed as it lies in same column or is in diagonal with other queens then back-tracking is done to the previous queens in order of 3,2,1 to achieve the unique feasible solution.
6. For an n problem queen the same way all the n queens are placed and if the  $n^{th}$  cannot be placed back-tracking is done and the queens are re-ordered and solution is obtained.

### **Frequently Asked Questions:**

1. What is the Constraint satisfaction Algorithm?
2. Explain Graph coloring Problem in detail

**Conclusion;** Successfully implemented constraint satisfaction algorithm

### **Assignment No.:05**

#### **Problem Statement:**

Develop an elementary chatbot for any suitable customer interaction application.

#### **Objectives:**

1. To study elementary chatbot for any suitable customer interaction application

### Theory:

[Artificial intelligence chatbots](#) are text- or voice-based interfaces that provide support and connect human users with the services or information they need by simulating a traditional person-to-person conversation. Text-based chatbots are often deployed online on websites and social media platforms to provide customer support and outreach. Voice-based chatbots, on the other hand, are most typically used for call deflection and sorting or over-the-phone customer service.

Most smartphones come equipped with a built-in chatbot, and smart speakers with chatbot functionality have been trendy gift-giving items for several years.

The most typical chatbot interaction occurs on a business site. These customer service bots usually pop up after a human user navigates around a site for a few minutes or exhibits behaviors that show that they have become “lost” or are having trouble connecting with the information they need.

Once the chatbot window presents itself, the user can enter their question in plain, syntactical English. [The bot's language recognition functions break down the question](#) and, at the speed of light, compares the query to its data bank of previously asked questions to look for ways customers have achieved satisfying results in similar situations.

When that search is complete, the chatbot shares the best and most relevant information with the user.

### Applications

Chatbots are valuable for both businesses and consumers, as [they reduce barriers to data access created by everything from physical disability to tech savviness, streamline navigation to connect users with results](#) as quickly as possible, and provide a cost-effective alternative to staffing massive numbers of support professionals or contracting with expensive call centers.

In fact, in many scenarios, [artificial intelligence chatbot services](#) can actually provide a faster, more straightforward experience than dealing with a human professional. If that sounds too good to be true, here are some examples of ways AI-based chatbots are deployed every day:

- **Software Requirements: Google ColabConclusion:**

Successfully implemented elementary chatbot for any suitable customer interaction application.

**Conclusion:** Successfully Developed an elementary chatbot for any suitable customer interaction application.

### Assignment No.:06

#### Problem Statement:

Implement any one of the following Expert System

- I. Information management
- II. Hospitals and medical facilities

- III. Help desks management
- IV. Employee performance evaluation
- V. Stock market trading

OR

Design a Miniproject on any one of the following topics not limited to:-

- Airline scheduling and cargo schedules
- Hospital Management system
- Ticket booking system
- Food Ordering System
- Task management system

Library Management System

### **Objectives:**

To study Expert System/ Designed System

### **Theory:**

A system that uses human expertise to make complicated decisions. Simulates reasoning by applying knowledge and interfaces. Uses expert's knowledge as rules and data within the system. Models the problem solving ability of a human expert.

Components of an ES:

1. Knowledge Base
  - i. Represents all the data and information inputed by experts in the field.
  - ii. Stores the data as a set of rules that the system must follow to make decisions.
2. Reasoning or Inference Engine
  - i. Asks the user questions about what they are looking for.
  - ii. Applies the knowledge and the rules held in the knowledge base.
  - iii. Appropriately uses this information to arrive at a decision.
3. User Interface
  - i. Allows the expert system and the user to communicate.
  - ii. Finds out what it is that the system needs to answer.
  - iii. Sends the user questions or answers and receives their response.
4. Explanation Facility
  - i. Explains the systems reasoning and justifies its conclusions.

### **Cloud Computing:List Of assignment**

Sr.No	Problem Statement	CO
1	Case study on Microsoft azure to learn about Microsoft Azure is a cloud computing platform and infrastructure, created by Microsoft, for	CO4,CO5

	<p>building, deploying and managing applications and services through a global network of Microsoft-managed data centers.</p> <p>OR</p> <p>Case study on Amazon EC2 and learn about Amazon EC2 web services.</p>	
2	<p>Installation and configure Google App Engine.</p> <p>OR</p> <p>Installation and Configuration of virtualization using KVM</p>	CO4,CO5
3	Creating an Application in SalesForce.com using Apex programming Language	CO4,CO5
4	Design and develop custom Application (Mini Project) using Sales force Cloud.	CO4,CO5
5	<p><b>Mini-Project</b></p> <p>Setup your own cloud for Software as a Service (SaaS) over the existing LAN in your laboratory. In this assignment you have to write your own code for cloud controller using open- source technologies to implement with <b>Web Application</b>. Implement the basic operations may be like to divide the file in segments/blocks and upload/ download file on/from cloud in encrypted form</p>	CO6

### Assignment No 1

#### **Problem Statement:**

Case study on Microsoft azure to learn about Microsoft Azure is a cloud computing platform and infrastructure, created by Microsoft, for building, deploying and managing applications and services through a global network of Microsoft-managed data centers.

OR

Case study on Amazon EC2 and learn about Amazon EC2 web services

**Objective:** To study Microsoft azure with cloud platforms

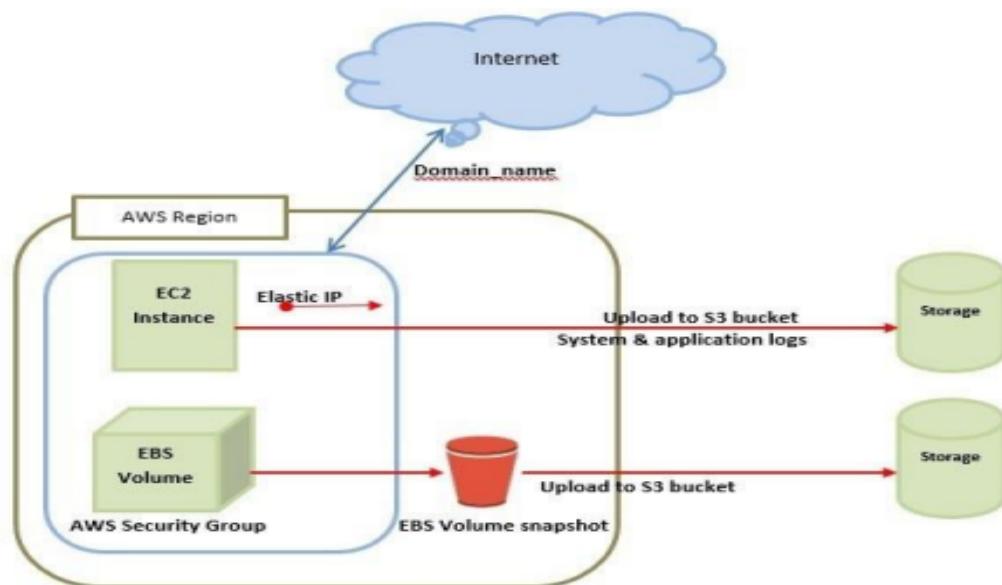
**Theory:**

### Introduction

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers. The simple web interface of Amazon EC2 allows users to obtain and configure capacity with minimal friction. It provides them with complete control of their computing resources and lets them run on Amazon's proven computing environment. Amazon EC2 reduces the time required to obtain and boot new server instances, which are otherwise called Amazon EC2 instances, to minutes, allowing the users to quickly scale capacity, both up and down, as their computing requirements change. Amazon EC2 changes the economics of computing by allowing users to pay only for capacity that they actually use. Amazon EC2 provides developers and system administrators the tools to build failure resilient applications and isolate themselves from common failure scenarios.

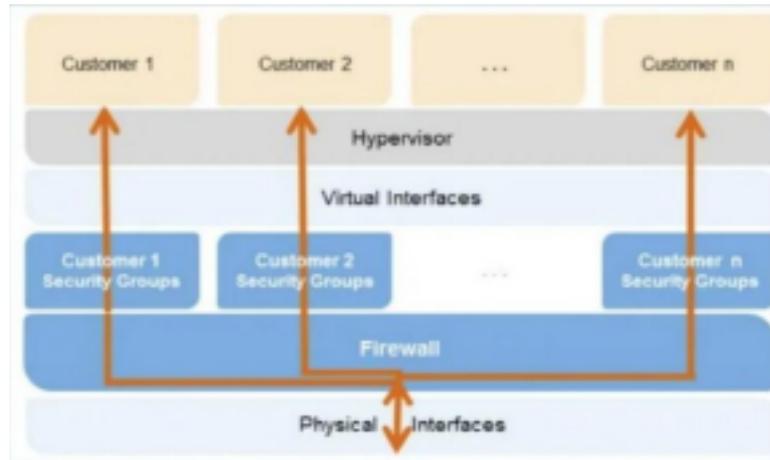
### Structure of EC2

The basic structure (diagrammatic representation) of AWS EC2 architecture, where EC2 stands for Elastic Compute Cloud is shown in the above diagram. EC2 allows users to use virtual machines of different configurations as per their requirement. It allows various configuration options, mapping of individual servers, various pricing options and so much more.



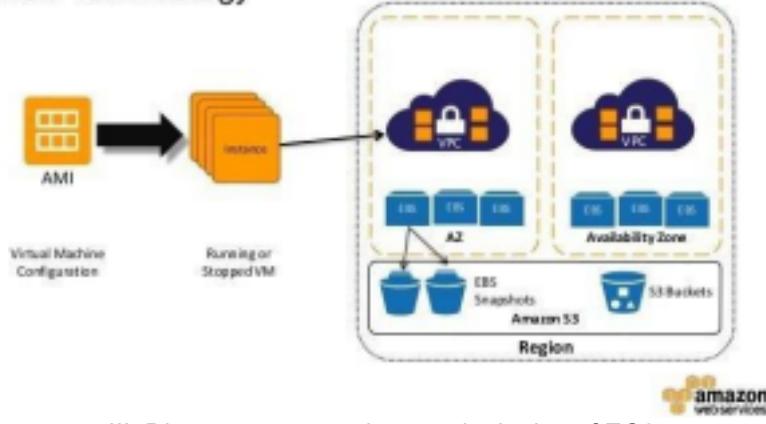
In the above diagram, S3 stands for Simple Storage Service. It allows the users to store and retrieve various types of data using API calls however, it doesn't contain any computing element.

The below images give a short representation of the layers of EC2, terminologies in EC2 and the diagrammatic representation of the same.

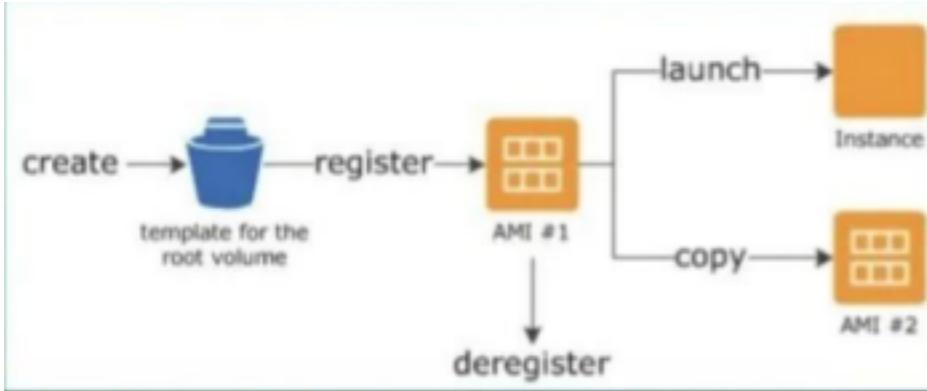


II. Diagram representing layers of EC2

### EC2 Terminology



III. Diagram representing terminologies of EC2



IV. Diagrammatic representation of EC2

## EC2 Components

In AWS EC2, the users should be aware about the EC2 components, their operating systems support, security measures, pricing structures and many more. They are as described below :

- **Operating System Support**

Amazon EC2 supports multiple OS in which users need to pay additional licensing fees like : Red Hat Enterprise, SUSE Enterprise and Oracle Enterprise Linux, UNIX,

Windows Server, etc. These OS need to be implemented in conjunction with Amazon Virtual Private Cloud (VPC).

- **Security**

Users have complete control over the visibility of their AWS account. In AWS EC2, the security system allows users to create groups and place running instances into it as per their requirement. One can specify the groups with which other groups may communicate, as well as the groups with which IP subnets on the Internet may talk.

- **Pricing**

AWS offers a variety of pricing options, depending on the type of resources, types of applications and database. It allows the users to configure their resources and compute the charges accordingly.

- **Fault Tolerance**

Amazon EC2 allows the users to access its resources to design fault-tolerant applications. EC2 also comprises geographic regions and isolated locations known as availability zones for fault tolerance and stability. It doesn't share the exact locations of regional data centers for security reasons. When the users launch an instance, they must select an AMI that's in the same region where the instance will run. Instances are distributed across multiple availability zones to provide continuous services in failures, and Elastic IP (EIPs) addresses are used to quickly map failed instance addresses to concurrent running instances in other zones to avoid delay in services.

- **Migration**

Migration service allows the users to move existing applications into EC2. It costs \$80.00 per storage device and \$2.49 per hour for data loading. This service suits those users having large amounts of data to be moved.

## **Features of EC2**

Amazon EC2 comes with a lot of features for an efficient and friendly experience for the users. Some of the prominent features of EC2 are discussed below :

- **Reliable :**

Amazon EC2 offers a highly reliable environment where replacement of instances is rapidly possible. Service Level Agreement commitment is 99.9% availability for each Amazon EC2 region.

- **Designed for Amazon Web Services :**

Amazon EC2 works fine with Amazon services like Amazon S3, Amazon RDS, Amazon DynamoDB and Amazon SQS. It provides a complete solution for computing, query processing and storage across a wide range of applications.

- **Secure :**

Amazon EC2 works in Amazon Virtual Private Cloud to provide a secure and robust network to resources. It provides secure login information for user's instances using key pairs. AWS stores the public key, and users store the private key in a secure place.

- **Flexible Tools :**

Amazon EC2 provides the tools for developers and system administrators to build failure applications and isolate themselves from common failure situations.

- **Inexpensive:**

Amazon EC2 wants users to pay only for the resources that they use. It includes multiple purchase plans such as On-Demand Instances, Reserved Instances, Spot Instances and many more which we can choose as per our requirement.

EC2 provides the following features in addition to the above discussed features. They are as follows:

- **Virtual computing environments:**

Amazon EC2 provides Virtual Computing Environments also known as instances.

- **Amazon Machine Images:**

Amazon EC2 provides preconfigured templates for user instances, known as Amazon Machine Images (AMIs), that package the bits the user needs for their server (including the operating system and additional software).

- **Instance Types:**

Amazon EC2 provides various configurations of CPU, memory, storage and networking capacity for user's instances, known as instance types.

- **Instance Store Volumes :**

Amazon EC2 also provides storage volumes for temporary data that's deleted when users stop, hibernate or terminate their instance, known as instance store volumes.

- **Amazon EBS Volumes :**

Amazon EC2 provides persistent storage volumes for user's data using Amazon Elastic Block Store (Amazon EBS), also known as Amazon EBS volumes.

- **Regions and Availability Zones (AZs) :**

Amazon EC2 also provides multiple physical locations for user's resources, such as instances and Amazon EBS volumes, known as Regions and Availability Zones.

- **Firewall :**

Amazon EC2 provides a firewall that enables users to specify the protocols, ports and source IP ranges that can reach their instances using security groups.

- **Elastic IP Addresses :**

Amazon EC2 also provides static IPv4 addresses for dynamic cloud computing, known as Elastic IP addresses.

- **Virtual Private Clouds (VPCs) :**

Amazon EC2 provides virtual networks which users can create that are logically isolated from the rest of the AWS Cloud and that they can optionally connect to their own network, known as Virtual Private Clouds (VPCs).

## **EC2 Instances**

Amazon EC2 provides Virtual Computing Environments also known as instances. EC2 instances can be resized and the number of instances can be scaled up or down as per the user requirements. These instances can be launched in one or more geographical locations or regions and Availability Zones (AZs). Each region comprises several AZs at distinct locations, connected by low latency networks in the same region.

- **On-Demand Instances :**

With On-Demand instances, users pay for compute capacity by the hour or the second depending on which instances they run. No longer-term commitments or upfront payments are needed. One can increase or decrease their compute capacity depending on the demands of their application and only pay the specified amount at hourly rates for the instance they use. On-Demand instances are recommended for :

- Users that prefer the low cost and flexibility of Amazon EC2 without any up-front payment or long term commitment
- Applications with short-term, spiky or unpredictable workloads that cannot be interrupted
- Applications being developed or tested on Amazon EC2 for the first time

- **Spot Instances :**

Spot Instances are available at much higher discount as compared to On-Demand prices and let users take advantage of unused Amazon EC2 capacity in the AWS Cloud. One can significantly reduce the cost of running their applications, grow their application's compute capacity and throughput for the same budget, and enable new types of cloud computing applications. Spot instances are recommended for :

- **Reserved Instances :**

Reserved Instances provide users with a significant discount (up to 72%) compared to On-Demand instance pricing. One has the flexibility to change families, operating system types and tenancies while benefiting from Reserved Instance pricing when one uses Convertible Reserved Instances.

- **Savings Plans :**

Savings Plans are a flexible pricing model that offer low prices on EC2 and Fargate usage, in exchange for a commitment to a consistent amount of usage (measured in \$/hour) for a 1 or 3 year term.

- **Dedicated Hosts :**

A Dedicated Host is a physical EC2 server dedicated for users to use. Dedicated Hosts can help one reduce costs by allowing them to use their existing server-bound software licenses, including Windows Server, SQL Server and SUSE Linux Enterprise Server (subject to their license terms) and can also help them meet compliance requirements.

## **Amazon EC2 Functionalities**

Some of the functionalities provided by Amazon EC2 are discussed as below :

- **Load Balancing**

Load balancing simply means hardware or software load over web servers, that improves the efficiency of the server as well as the application. Hardware load balancer is a very common network appliance used in traditional web application architectures.

AWS provides the Elastic Load Balancing service, it distributes the traffic to EC2 instances across multiple available sources, and dynamic addition and removal of Amazon EC2 hosts from the load-balancing rotation. Elastic Load Balancing can dynamically grow and shrink the load-balancing capacity to adjust to traffic demands and also support sticky sessions to address more advanced routing needs.

- **Amazon Cloud-front**

It is responsible for content delivery, i.e. used to deliver websites. It may contain dynamic, static, and streaming content using a global network of edge locations. Requests for content at the user's end are automatically routed to the nearest edge location, which improves the performance.

Amazon Cloud-front is optimized to work with other Amazon Web Services, like Amazon S3 and Amazon EC2. It also works fine with any non-AWS origin server and stores the original files in a similar manner. In Amazon Web Services, there are no contracts or monthly commitments. The pay is only for as much content is being delivered through the service.

- **Elastic Load Balancer**

It is used to spread the traffic to web servers, which improves performance. AWS provides the Elastic Load Balancing service, in which traffic is distributed to EC2 instances over multiple available zones, and dynamic addition and removal of Amazon EC2 hosts from the load-balancing rotation. Elastic Load Balancing can dynamically grow and shrink the load-balancing capacity as per the traffic conditions.

- **Security Management**

Amazon's Elastic Compute Cloud (EC2) provides a feature called security groups, which is similar to an inbound network firewall, in which users have to specify the protocols, ports, and source IP ranges that are allowed to reach their EC2 instances. Each EC2 instance can be assigned one or more security groups, each of which routes the appropriate traffic to each instance. Security groups can be configured using specific subnets or IP addresses which limits access to EC2 instances.

- **Elastic Caches**

Amazon Elastic Cache is a web service that manages the memory cache in the cloud. In memory management, cache has a very important role and helps to reduce the load on the services, improves the performance and scalability on the database tier by caching frequently used information.

- **Amazon RDS**

Amazon RDS (Relational Database Service) provides a similar access as that of MySQL, Oracle or Microsoft SQL Server database engine. The same queries, applications and tools can be used with Amazon RDS. It automatically patches the database software and manages backups as per the user's instruction. It also supports point-in-time recovery. There are no up-front investments required and the pay is only for the resources that are being used.

- **Hosting RDMS on EC2 Instances**

Amazon RDS allows users to install RDBMS (Relational Database Management System) of their choice like MySQL, Oracle, SQL Server, DB2 and many more on an EC2 instance and can manage them as required.

Amazon EC2 uses Amazon EBS (Elastic Block Storage) similar to network-attached storage. All data and logs running on EC2 instances should be placed on Amazon EBS volumes, which will be available even if the database host fails.

Amazon EBS volumes automatically provide redundancy within the availability zone, which increases the availability of simple disks. Further if the volume is not sufficient for the user's database needs, volume can be added to increase the performance for their database. Using Amazon RDS, the service provider manages the storage and users can only focus on managing the data.

- **Storage & Backups**

AWS cloud provides various options for storing, accessing and backing up web application data and assets. The Amazon S3 (Simple Storage Service) provides a simple web-services interface that can be used to store and retrieve any amount of data, at any time, from anywhere on the web.

Amazon S3 stores data as objects within resources called buckets. The user can store as many objects as per requirement within the bucket, and can read, write and delete objects from the bucket.

Amazon EBS is effective for data that needs to be accessed as block storage and requires persistence beyond the life of the running instance, such as database partitions and application logs. Amazon EBS volumes can be maximized up to 1 TB, and these volumes can be striped for larger volumes and increased performance. Provisioned

IOPS volumes are designed to meet the needs of database workloads that are sensitive to storage performance and consistency. Amazon EBS currently supports up to 1,000 IOPS per volume.

Users can stripe multiple volumes together to deliver thousands of IOPS per instance to an application.

- **Auto Scaling**

The difference between AWS cloud architecture and the traditional hosting model is that AWS can dynamically scale the web application fleet on demand to handle changes in traffic.

In the traditional hosting model, traffic forecasting models are generally used to provision hosts ahead of projected traffic. In AWS, instances can be provisioned on the fly according to a set of triggers for scaling the fleet out and back in. Amazon Auto Scaling can create capacity groups of servers that can grow or shrink on demand.

### **Amazon EC2 Web Services**

- **Amazon EC2 Auto Scaling**

The difference between AWS cloud architecture and the traditional hosting model is that AWS can dynamically scale the web application fleet on demand to handle changes in traffic.

In the traditional hosting model, traffic forecasting models are generally used to provision hosts ahead of projected traffic. In AWS, instances can be provisioned on the fly according to a set of triggers for scaling the fleet out and back in. Amazon Auto Scaling can create capacity groups of servers that can grow or shrink on demand.

Amazon EC2 Auto Scaling helps users to maintain application availability and allows them to automatically add or remove EC2 instances according to the conditions they define. They can use the fleet management features of Amazon EC2 Auto Scaling to maintain the health and availability of their fleet. Users can also use the dynamic and predictive scaling features of Amazon EC2 Auto Scaling to add or remove EC2 instances. Dynamic scaling responds to changing demand and predictive scaling automatically schedules the right number of EC2 instances based on predicted demand. Dynamic scaling and predictive scaling can be used together to scale faster.

- **Amazon EC2 Image Builder**

EC2 Image Builder simplifies the building, testing and deployment of Virtual Machine and container images for use on AWS or on-premises. Keeping Virtual Machine and container images up-to-date can be time consuming, resource intensive and error-prone..

## Amazon Web Services

<b>Compute</b>	<b>Administration &amp; Security</b>	<b>Application Services</b>
 <b>EC2</b> Virtual Servers in the Cloud	 <b>Directory Service</b> Managed Directories in the Cloud	 <b>SQS</b> Message Queue Service
 <b>Lambda</b> Run Code in Response to Events	 <b>Identity &amp; Access Management</b> Access Control and Key Management	 <b>SWF</b> Workflow Service for Coordinating Application Components
 <b>EC2 Container Service</b> Run and Manage Docker Containers	 <b>Trusted Advisor</b> AWS Cloud Optimization Expert	 <b>AppStream</b> Low Latency Application Streaming
<b>Storage &amp; Content Delivery</b>	 <b>CloudTrail</b> User Activity and Change Tracking	 <b>Elastic Transcoder</b> Easy-to-use Scalable Media Transcoding
 <b>S3</b> Scalable Storage in the Cloud	 <b>Config</b> Resource Configurations and Inventory	 <b>SES</b> Email Sending Service
 <b>Elastic File System</b> PREVIEW Fully Managed File System for EC2	 <b>CloudWatch</b> Resource and Application Monitoring	 <b>CloudSearch</b> Managed Search Service
 <b>Storage Gateway</b> Integrates On-Premises IT Environments with Cloud Storage	 <b>Service Catalog</b> Personalized Catalog of AWS Resources	 <b>API Gateway</b> Build, Deploy and Manage APIs
 <b>Glacier</b> Archive Storage in the Cloud	<b>Deployment &amp; Management</b>	<b>Mobile Services</b>
 <b>CloudFront</b> Global Content Delivery Network	 <b>Elastic Beanstalk</b> AWS Application Container	 <b>Cognito</b> User Identity and App Data Synchronization
<b>Database</b>	 <b>OpsWorks</b> DevOps Application Management Service	 <b>Device Farm</b> Test Android, Fire OS, and iOS apps on real devices in the Cloud
 <b>RDS</b> MySQL, Postgres, Oracle, SQL Server, and Amazon Aurora	 <b>CloudFormation</b> Templated AWS Resource Creation	 <b>Mobile Analytics</b> Collect, View and Export App Analytics
 <b>DynamoDB</b> Predictable and Scalable NoSQL Data Store	 <b>CodeDeploy</b> Automated Deployments	 <b>SNS</b> Push Notification Service
 <b>ElastiCache</b> In-Memory Cache	 <b>CodeCommit</b> Managed Git Repositories	<b>Enterprise Applications</b>
 <b>Redshift</b> Managed Petabyte-Scale Data Warehouse Service	 <b>CodePipeline</b> Continuous Delivery	 <b>WorkSpaces</b> Desktops in the Cloud
<b>Networking</b>	<b>Analytics</b>	 <b>WorkDocs</b> Secure Enterprise Storage and Sharing Service
 <b>VPC</b> Isolated Cloud Resources	 <b>EMR</b> Managed Hadoop Framework	 <b>WorkMail</b> PREVIEW Secure Email and Calendering Service
 <b>Direct Connect</b> Dedicated Network Connection to AWS	 <b>Kinesis</b> Real-time Processing of Streaming Big Data	
 <b>Route 53</b> Scalable DNS and Domain Name Registration	 <b>Data Pipeline</b> Orchestration for Data-Driven Workflows	
	 <b>Machine Learning</b> Build Smart Applications Quickly and Easily	

## AWS Management Console

AWS Management Console is a web application for managing Amazon Web Services. AWS Management Console consists of a list of various services to choose from. It also provides all the information related to user's account like billing. This console provides an inbuilt user interface to perform AWS tasks like working with Amazon S3 buckets, launching and connecting to Amazon EC2 instances, setting Amazon CloudWatch alarms and many more.

**Conclusion:**Hence we successfully studied microsoft azure

## Assignment No 2

### **Problem Statement:**

Installation and configure Google App Engine.

OR

Installation and Configuration of virtualization using KVM.

**Objective:** To study, install and configure google app Engine.

### **Theory:**

Before begin

1. Create a Google Cloud Platform project, if you don't have one already.

2. Make sure that Python 2.7 is installed on your system:

```
python -V
```

Note: As of Cloud SDK version 206.0.0, the gcloud CLI has experimental support for running using a Python 3.4+ interpreter (run gcloud topic startup for exclusions and more information on configuring your Python interpreter). All other Cloud SDK tools still require a Python 2.7 interpreter.

3. Download the archive file best suited to your operating system. Most machines will run the 64-bit package. If you'd like to check, run uname -m to verify if you're running a 64-bit system.

Platform Package Size SHA256 Checksum

Linux

64-bit

(x86\_64)

google-cloud-sdk-  
229.0.0-linux-  
x86\_64.tar.gz

25.6

MB

b1c87fc9451598a76cf66978dd8aa06482bfced639b56cf31559dc2c7f8b7b90

Linux

32-bit

(x86)

google-cloud-sdk-  
229.0.0-linux-  
x86.tar.gz

25.2

MB

ee8c45f8018d0fee92b07c32cc6d8c891241da0b88bfe289d4e58e6746c3f668

Alternatively, to download the Linux 64-bit archive file from your command-line, run:

```
curl -O https://dl.google.com/dl/cloudsdk/channels/rapid/downloads/google-cloud-sdk-229.0.0-linux-x86_64.tar.gz
```

For the 32-bit archive file, run:

```
curl -O https://dl.google.com/dl/cloudsdk/channels/rapid/downloads/google-cloud-sdk-229.0.0-linux-x86.tar.gz
```

4. Extract the archive to any location on your file system; preferably, your Home folder. On Linux, you can extract the archive file by running this command:

```
tar zxvf [ARCHIVE_FILE] google-cloud-sdk
```

5. If you're having trouble getting the gcloud command to work, ensure your \$PATH is defined appropriately. Use the install script to add Cloud SDK tools to your path. You will also be able to opt-in to command-completion for your bash shell and usage statistics collection during the installation process. Run the script using this command:

```
./google-cloud-sdk/install.sh
```

Restart your terminal for the changes to take effect.

Alternatively, you can call Cloud SDK after extracting the downloaded archive by invoking its executables via the full path.

#### Initialize the SDK

Use the gcloud init command to perform several common SDK setup tasks. These include authorizing the SDK tools to access Google Cloud Platform using your user account credentials and setting up the default SDK configuration.

To initialize the SDK:

1. Run the following at a command prompt:

```
gcloud init
```

Note: To prevent the command from launching a web browser, use gcloud init --console-only instead. To authorize without a web browser and non-interactively, create a service

account with the appropriate scopes using the Google Cloud Platform Console and use gcloud auth activate-service-account with the corresponding JSON key file.

2. Accept the option to log in using your Google user account:

To continue, you must log in. Would you like to log in (Y/n)? Y

3. In your browser, log in to your Google user account when prompted and click Allow to grant permission to access Google Cloud Platform resources.

4. At the command prompt, select a Cloud Platform project from the list of those where you have Owner, Editor or Viewer permissions:

Pick cloud project to use:

[1] [my-project-1]

[2] [my-project-2]

...

Please enter your numeric choice:

If you only have one project, gcloud init selects it for you.

5. If you have the Google Compute Engine API enabled, gcloud init allows you to choose a default Compute Engine zone:

Which compute zone would you like to use as project default?

- [1] [asia-east1-a]
- [2] [asia-east1-b]

...

- [14] Do not use default zone

Please enter your numeric choice:

gcloud init confirms that you have complete the setup steps successfully:  
gcloud has now been configured!

You can use [gcloud config] to change more gcloud settings.

Your active configuration is: [default]

Run core gcloud commands

Run these gcloud commands to view information about your SDK installation:

1. To list accounts whose credentials are stored on the local system:

gcloud auth list

gcloud displays a list of credentialed accounts:

Credentialed Accounts

ACTIVE ACCOUNT

\* example-user-1@gmail.com

example-user-2@gmail.com

2. To list the properties in your active SDK configuration:

gcloud config list

gcloud displays the list of properties:

[core]

account = example-user-1@gmail.com

disable\_usage\_reporting = False

project = example-project

3. To view information about your Cloud SDK installation and the active SDK configuration:

gcloud info

gcloud displays a summary of information about your Cloud SDK installation. This includes information about your system, the installed SDK components, the active user account and current project, and the properties in the active SDK configuration.

4. To view information about gcloud commands and other topics from the command line:

gcloud help

For example, to view the help for gcloud compute instances create:

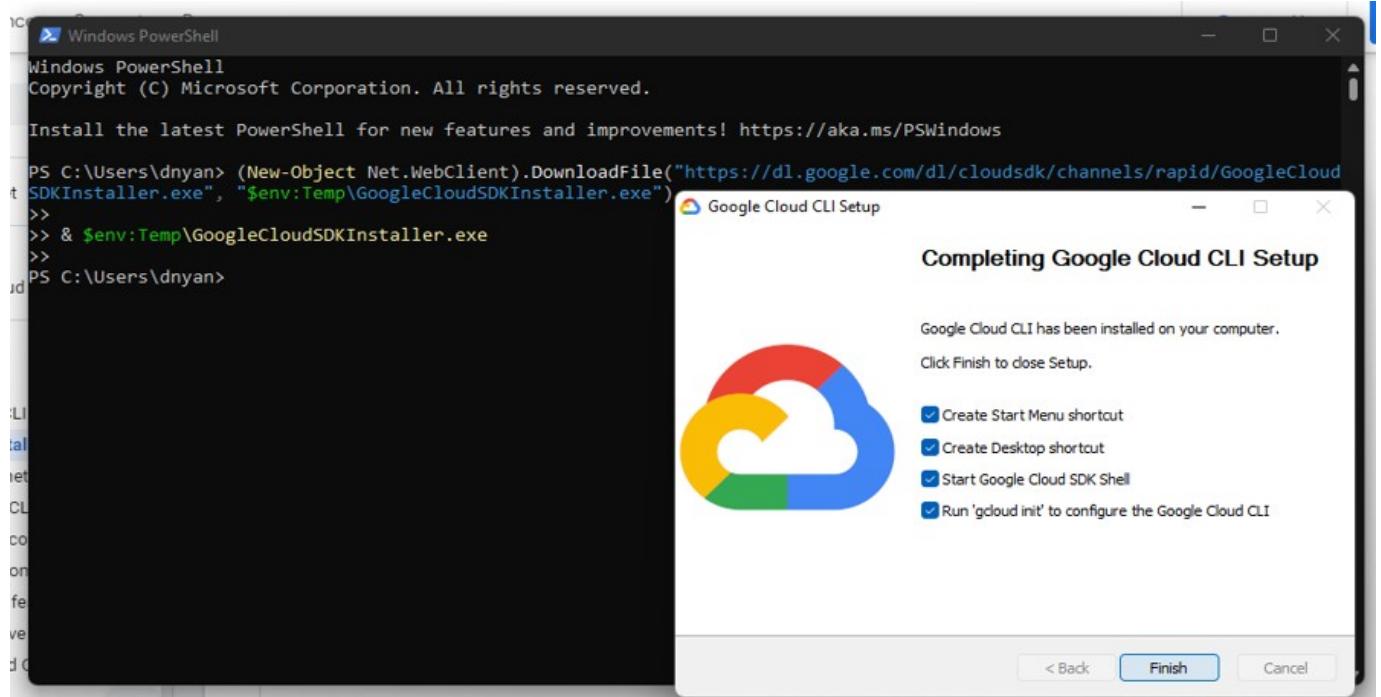
gcloud help compute instances create

gcloud displays a help topic that contains a description of the command, a list of command flags and arguments, and examples of how to use it.

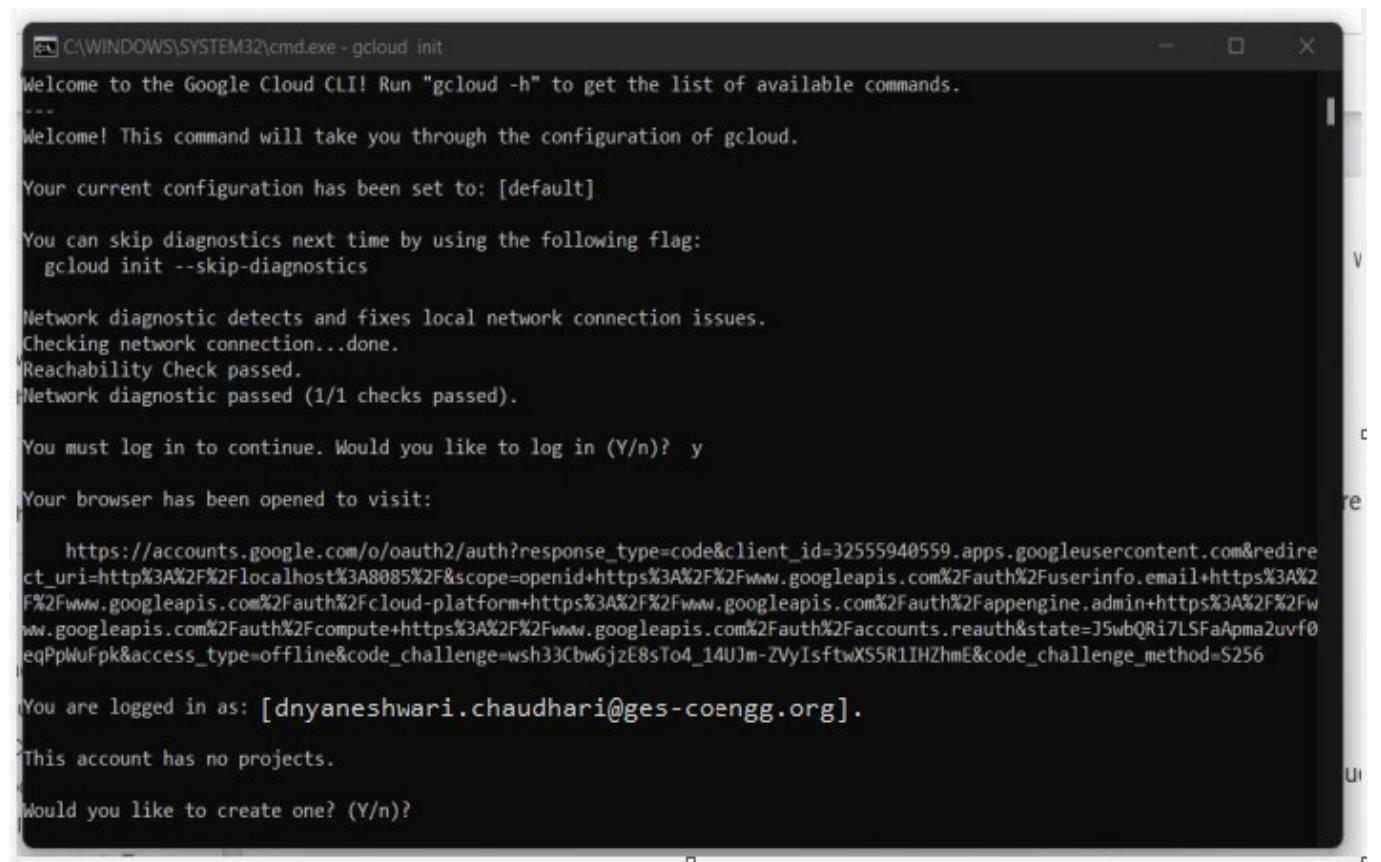
How to Run Program:

Now as we have finished installing app engine, now it's time to create and upload an app. In this case we will be taking example of a "HELLO WORLD" app in python.

1. As we already have made sure that we have python installed in our system, It will be easier for us to clone existing code and deploy it rather than creating our own so we will use python docs-sample. Run the command "git clone https://github.com/GoogleCloudPlatform/python-docs-samples".



2. cd python docs- samples/appengine/standard/hello\_world  
3. dev\_appserver.py app.yaml



## Assignment No 3

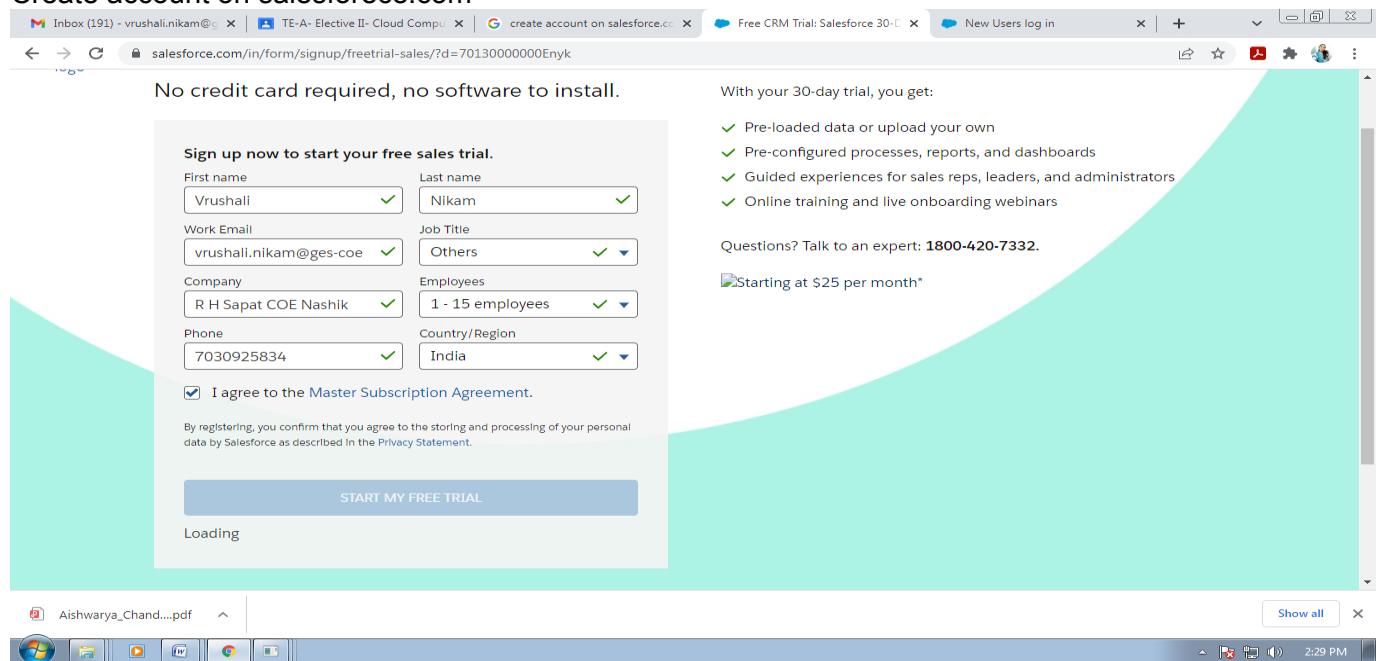
### Problem Statement:

Creating an Application in SalesForce.com using Apex programming Language.

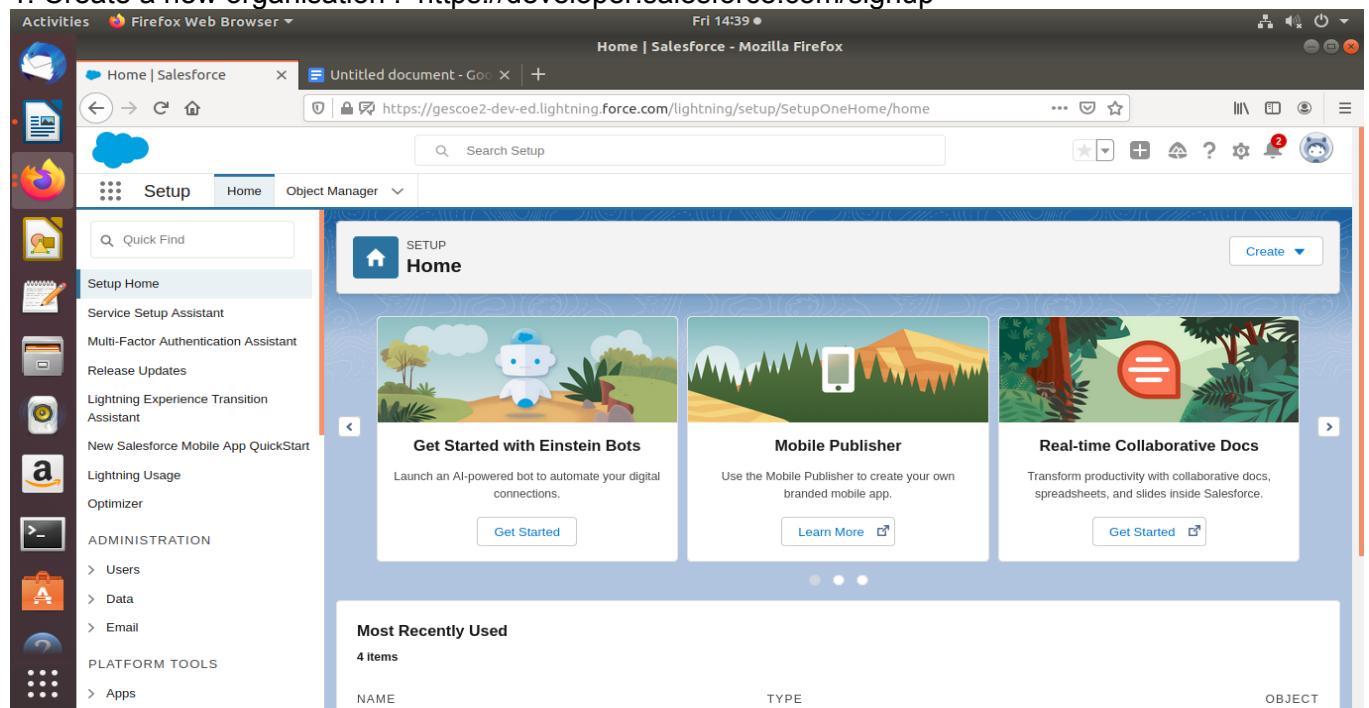
**Objective:** To create account of SalesForce.com using apex programming language

### Theory:

Create account on salesforce.com



1: Create a new organisation : <https://developer.salesforce.com/signup>

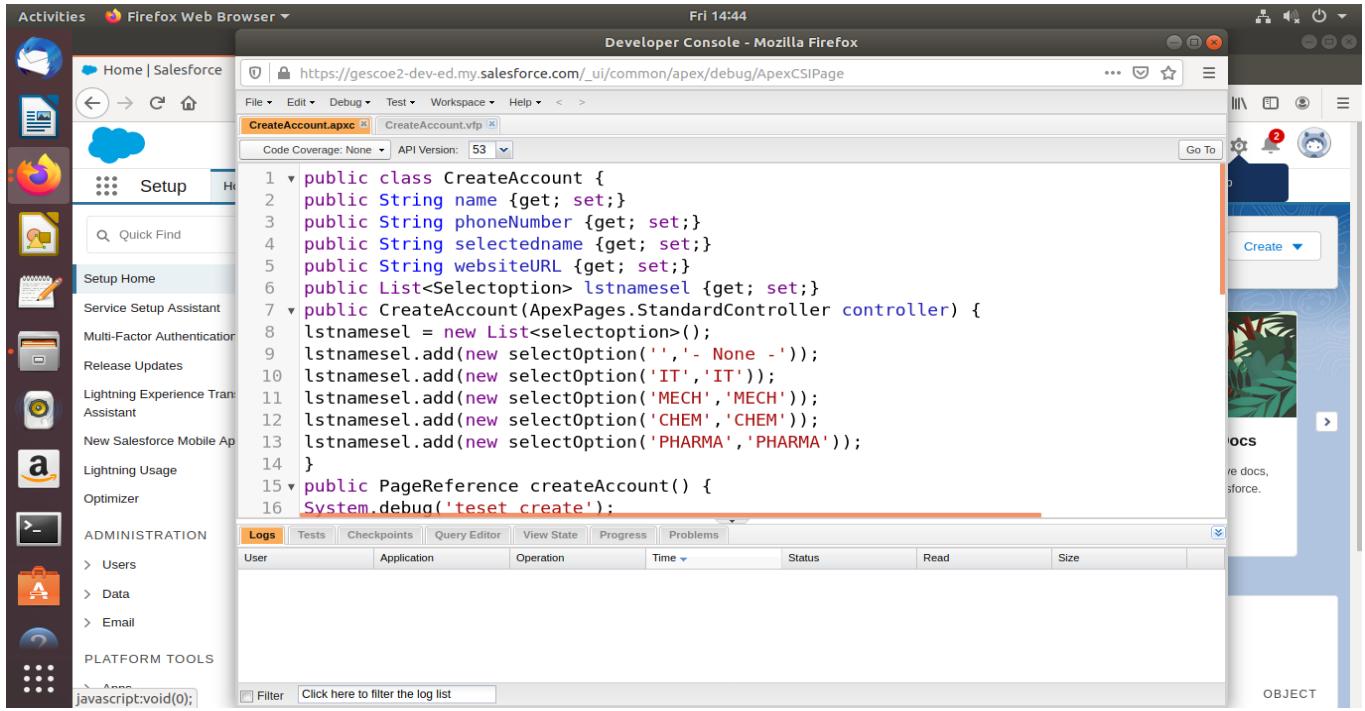


2: Login To Salesforce.com

3: Go To developer console

4: Create New Apex Class with name CreateAccount.apxc and paste the following code

```
public class CreateAccount {  
    public String name {get; set;}  
    public String phoneNumber {get; set;}  
    public String selectedname {get; set;}  
    public String websiteURL {get; set;}  
    public List<Selectoption> lstnamesel {get; set;}  
    public CreateAccount(ApexPages.StandardController controller) {  
        lstnamesel = new List<selectoption>();  
        lstnamesel.add(new selectOption("- None -"));  
        lstnamesel.add(new selectOption('IT','IT'));  
        lstnamesel.add(new selectOption('MECH','MECH'));  
        lstnamesel.add(new selectOption('CHEM','CHEM'));  
        lstnamesel.add(new selectOption('PHARMA','PHARMA'));  
    }  
    public PageReference createAccount() {  
        System.debug('teset create');  
        if(!String.isEmpty(name)) {  
            Account accountRecord = new Account(Name = name,  
                Phone = phoneNumber,  
                Industry = selectedname,  
                Website = websiteURL);  
            INSERT accountRecord;  
            PageReference pg = new PageReference('/'+accountRecord.Id);  
            pg.setRedirect(true);  
            return pg;  
        } else {  
            ApexPages.addmessage(new ApexPages.message(ApexPages.severity.CONFIRM,'Please enter Account Name'));  
        }  
        return NULL;  
    }  
    public PageReference cancelAccount() {  
        PageReference pg = new PageReference('/'+Schema.SObjectType.Account.getKeyPrefix()+'o');  
  
        return pg;  
    }  
}
```



Create new -> visualforce page file -> CreateAccount.vfp  
Paste the following code :

```
<apex:page standardController="Account" extensions="CreateAccount">
<apex:form id="apexFrom">
<apex:pageBlock title="Create Account:" id="pageBlockId">
<apex:pageMessages id="showmsg"></apex:pageMessages>
<apex:pageBlockSection columns="2">
<!--<div class = "requiredInput">
<div class = "requiredBlock"></div>
<apex:inputText value="{!name}" label="Account Name" required="true"/>
</div-->
<apex:inputText value="{!name}" label="Account Name" required="true"/>
<apex:inputText value="{!phoneNumber}" label="Phone" />
<apex:selectList size="1" value="{!selectedname}" label="Industry">
<apex:selectOptions value="{!lstnamesel}"/>
</apex:selectList>
<apex:inputText value="{!websiteURL}" label="Website" />
</apex:pageBlockSection>
<apex:pageBlockButtons >
<apex:commandButton action="{!createAccount}" value="Save"/>
<apex:commandButton action="{!cancelAccount}" value="Cancel"/>
</apex:pageBlockButtons>
</apex:pageBlock>
<!-- ACTION FUNCTIONS-->
<apex:actionFunction name="createFunction" action="{!createAccount}"/>
<apex:actionFunction name="cancelFunction" action="{!cancelAccount}"/>
</apex:page>
```

Activities Firefox Web Browser ▾ Fri 14:47

Developer Console - Mozilla Firefox

https://gescoe2-dev-ed.my.salesforce.com/\_ui/common/apex/debug/ApexCSIPage

CreateAccount.apxc CreateAccount.vfp

File Edit Debug Test Workspace Help < >

Preview API Version: 53

```

1 <apex:page standardController="Account" extensions="CreateAccount">
2 <apex:form id="apexFrom" >
3 <apex:pageBlock title="Create Account:" id="pageBlockId">
4 <apex:pageMessages id="showmsg"></apex:pageMessages>
5 <apex:pageBlockSection columns="2" >
6 <!--<div class = "requiredInput">
7 <div class = "requiredBlock"></div>
8 <apex:inputText value="{!!name}" label="Account Name" required="true"/>
9 </div>-->
10 <apex:inputText value="{!!name}" label="Account Name" required="true"/>
11 <apex:inputText value="{!!phoneNumber}" label="Phone" />
12 <apex:selectList size="1" value="{!!selectedname}" label="Industry">
13 <apex:selectOptions value="{!!lstnamesel}" />
14 </apex:selectList>
15 <apex:inputText value="{!!websiteURL}" label="Website" />
16 </apex:pageBlockSection>

```

Logs Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size

Filter Click here to filter the log list

## Go to Tabs Section

Activities Firefox Web Browser ▾ Fri 14:53

Tabs | Salesforce - Mozilla Firefox

https://gescoe2-dev-ed.lightning.force.com/lightning/setup/CustomTabs/page?address=%2Fset

Setup Home Object Manager

tabs

User Interface

Rename Tabs and Labels

**Tabs**

Didn't find what you're looking for? Try using Global Search.

Choose the page for this new tab. Fill in other details.

Error: Invalid Data. Review all error messages below to correct your data.

Select an existing page or [create a new page now](#).

Visualforce Page: CreateAccount [CreateAccount]

Tab Label: Quick Account

Tab Name: Quick\_Account

Tab Style: Airplane

Error: Please choose a tab style.

(Optional) Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab.

Splash Page Custom Link: --None--

Enter a short description.

Description: Create Sample account

The screenshot shows the Salesforce Setup interface. On the left, there's a sidebar with various icons and a search bar. The main area is titled "Visualforce Pages" under the "SETUP" tab. It displays a list of Visualforce Pages with columns for Action, Label, Name, Namespace Prefix, Api Version, Description, Created By Alias, Created Date, Last Modified By Alias, and Last Modified Date. A single row is selected, showing "CreateAccount" as the Label and Name, with "53.0" in the Api Version column.

Preview the webpage:

The screenshot shows a browser window displaying a Visualforce page titled "Create Account". The page has fields for "Account Name", "Industry" (set to "-None-"), "Phone", and "Website". There are "Save" and "Cancel" buttons at the top and bottom of the form.

Enter the data and save

The screenshot shows a Salesforce account page titled "abcd | Salesforce - Mozilla Firefox". The page displays basic account information: Type (Account), Phone ((123) 456-7890), Website (google.com), Account Owner (john doe), Account Site, and Industry (IT). Below this, the "Related" section lists "Contacts (0)", "Opportunities (0)", and "Cases (0)". On the right, there are sections for "Activity" (New Task, Log a Call, New Event, Email) and "Chatter" (Admin Notice: "It's Time for MFA"). The browser's address bar shows the URL: https://gescoe2-dev-ed.lightning.force.com/lightning/r/Account/0015j00000Yn9FDAAZ/view.

**Conclusion:** Hence we have successfully created account of SalesForce.com using apex programming language

## Assignment No 4

### Problem Statement:

Design and develop custom Application (Mini Project) using Sales force Cloud..

**Objective:** To Study ,Design and develop custom Application (Mini Project) using Sales force Cloud..

**Theory:** Design and develop custom Application (Mini Project) using Salesforce Cloud Step-1: Click on Lightning Experience. Step-2: Click on Setup. Step-3: Click on Create, Select Custom Object/Create Object. Step-4: Fill the Given Fields,

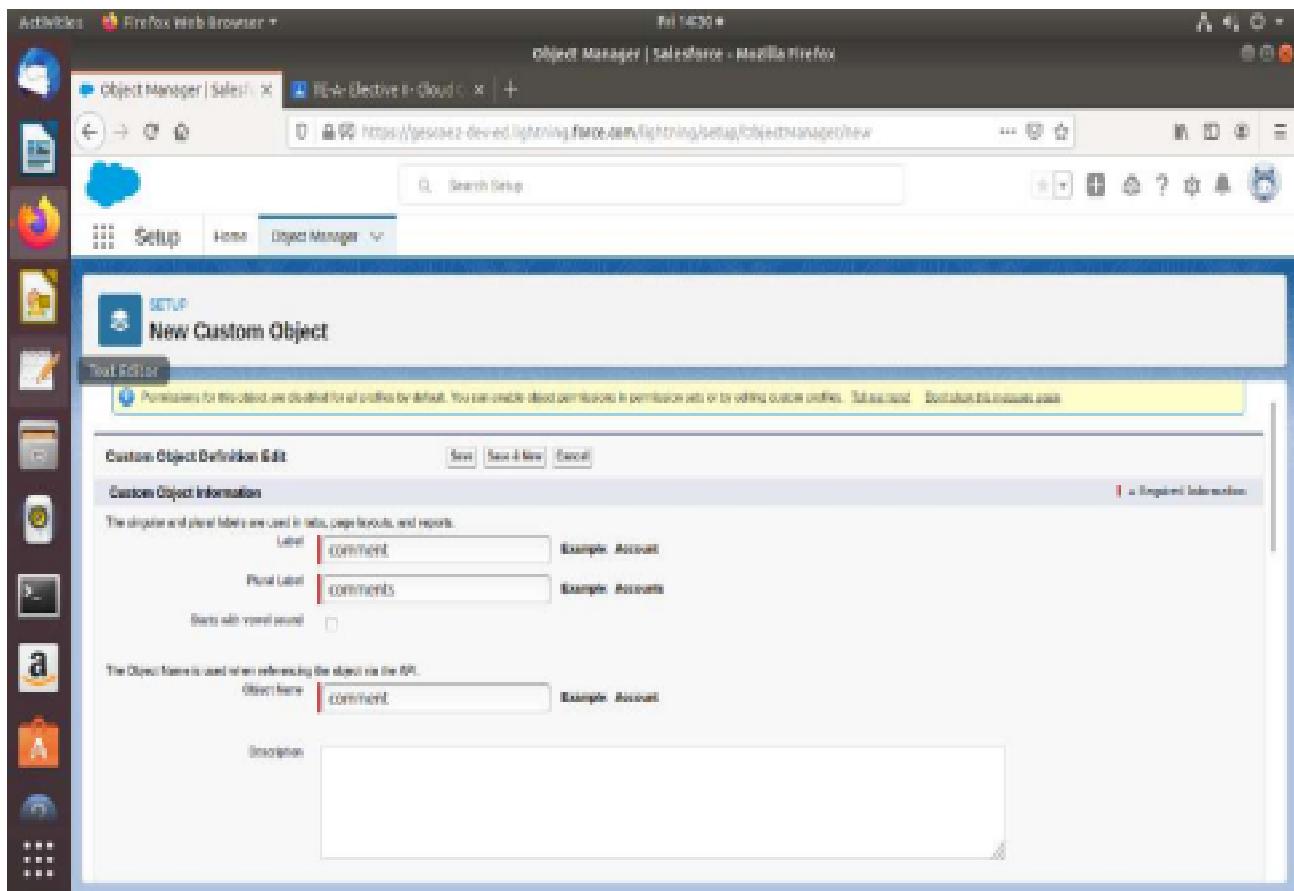
Label: comment,

Plural label: comments,

Data type: text,

Select Allow Reports Check Box,

Click on Save.



The screenshot shows two views of the Salesforce Object Manager interface.

**Top View:** A 'New Custom Object' setup page. The 'Record Name' field is set to 'comment\_\_Name' and the 'Data Type' is 'Text'. Under 'Optional Features', 'Allow Reports' is checked. Under 'Object Classification', 'Allow Sharing' and 'Allow Web API Access' are checked. The URL in the browser is <https://test.salesforce.com/lightning/setup/ObjectManager/new>.

**Bottom View:** The 'comment' object details page. The 'Details' tab is selected, showing fields like 'API Name' (comment\_\_c), 'Custom' (true), 'Label' (comment), and 'Plural Label' (comments). Other tabs include 'Fields & Relationships', 'Page Layouts', 'Lightning Record Pages', 'Buttons, Links, and Actions', 'Compact Layouts', 'Field Sets', 'Object Limits', 'Record Types', 'Related Lookup Filters', and 'Validation Rules'. The URL in the browser is <https://test.salesforce.com/lightning/sobject/ObjectManager/015/000001>.

Step-5: Go to Home.

Step-6: Click on Setup, Click on Create, Select Custom Tabs.

Activities Firefox Web Browser

Fri Nov 18 2016 10:41:00 GMT+0530 (IST)

File Insert Edit View Favorites Tools Help

Tab [ Salesforce - Mozilla Firefox ] Tab [ TE-A-Electricity-Cloud - Mozilla Firefox ] Tab [ Classes - Mozilla Firefox ] +

http://0.0.0.0:8080/lightning-force.com/lightning/setup/CustomTabs/pageId

Search Setup

Setup Home object manager

User Interface Tabs

Now Custom Object Tab

Step 1: Enter the Details Step 1 of 3

Choose the custom object for this new custom tab. Fill in other details.

Select an existing custom object or create a new custom object now

Object: comment

Tab Name: Comment

(Optional) Choose a Home Page Content Link to show on a splash page the first time your users click on this tab.

Home Page Content Link: None

Enter a short description.

Description:

Help for this Page

Activities Firefox Web Browser

Fri Nov 18 2016 10:41:00 GMT+0530 (IST)

File Insert Edit View Favorites Tools Help

Tab [ Salesforce - Mozilla Firefox ] Tab [ TE-A-Electricity-Cloud - Mozilla Firefox ] Tab [ Classes - Mozilla Firefox ] +

http://0.0.0.0:8080/lightning-force.com/lightning/setup/CustomTabs/pageId

Search Setup

Setup Home object manager

User Interface Tabs

Tabs

Action	Label	Tab Style	Description
New	Comments	Comment	

Web Tabs

New Web Tab

No Web Tabs have been defined

Visualforce Tabs

New Visualforce Tab

Action	Label	Tab Style	Description
New	Comment	Comment	Create simple comment

Lightning Component Tabs

New Lightning Component Tab

No Lightning component tabs have been defined

Lightning Page Tabs

New Lightning Page Tab

No Lightning Page Tabs have been defined

Step-7: For Object Select Comment,  
For Tab Style Select Any Icon,  
Click-Next-Next-Save.

**Lightning Experience App Manager**

App Name	Developer Name	Description	Last Modified	App Type	Ver.
All Tabs	AllTabSet	Build tabs in CRM dashboards and apps	2019/02/27, 9:14 ...	Classic	✓
Analytics Studio	Insights	Build business CRM dashboards and apps	2019/02/27, 9:14 ...	Classic	✓
App Launcher	AppLauncher	App Launcher table	2019/02/27, 9:14 ...	Classic	✓
Bolt Solutions	LightningBolt	Discover and manage business solutions designed for you	2019/02/27, 9:17 ...	Lightning	✓
Community	Community	Salesforce CRM Communities	2019/02/27, 9:14 ...	Classic	✓
Content	Content	Salesforce CRM Content	2019/02/27, 9:14 ...	Classic	✓
Digital Experiences	SalesforceCMS	Manage content and media for all of your sites	2019/02/27, 9:14 ...	Lightning	✓
Lightning Usage App	LightningUsageInstrument	View Adoption and Usage Metrics for Lightning Experience	2019/02/27, 9:14 ...	Lightning	✓
Marketing	Marketing	Build-in class audience and marketing automation	2019/02/27, 9:14 ...	Classic	✓
Platforms	Platform	The fundamental Lightning Platform	2019/02/27, 9:14 ...	Classic	✓
Queue Management	QueueManagement	Create and manage queues for your business	2019/02/27, 9:14 ...	Lightning	✓

Step-7: Search App Manager in Quick Search and Select App manager

**New Lightning App**

NAVIGATION ITEMS

Choose the items to include in the app, and arrange the order in which they appear. Users can personalize the navigation by add or move items, but users can't remove or re-order the items that you set. Some navigation items are available only for phone or only for desktop. These items are dropped from the navigation bar when the app is moved to a format that the item doesn't support.

**Available Items**

- All Tabs
- All Sites
- Alternative Payment Methods
- App Launcher

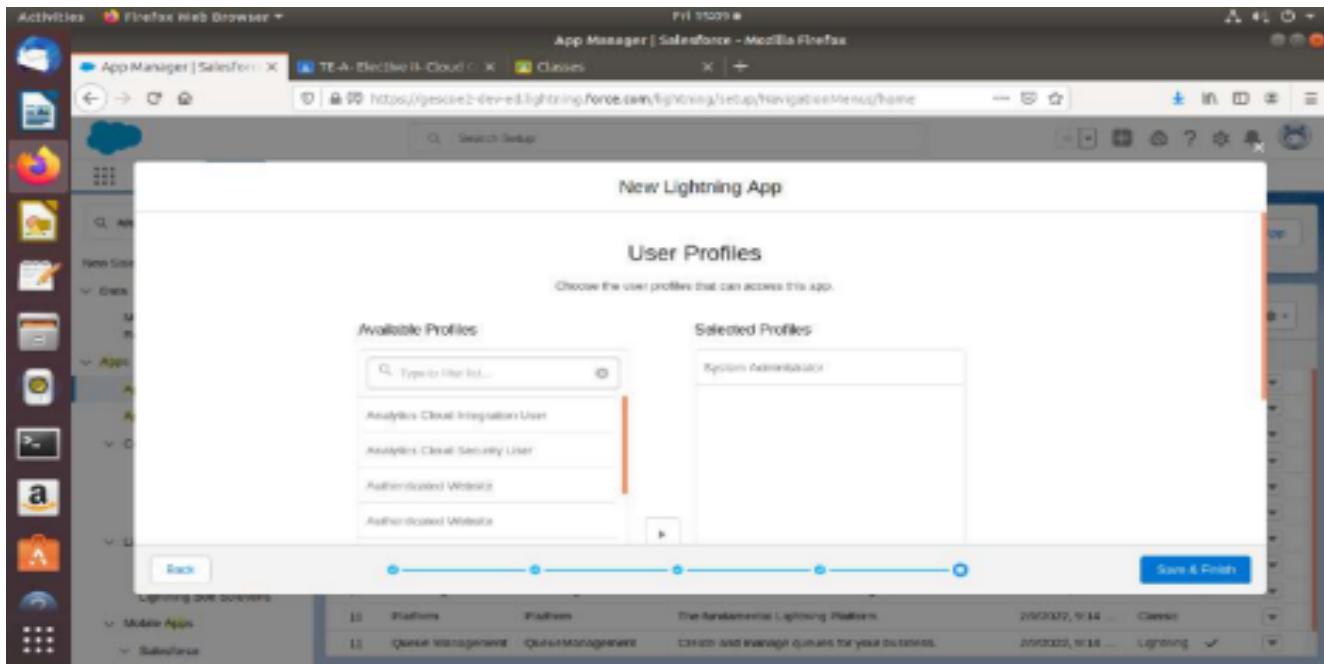
**Selected Items**

- Accounts

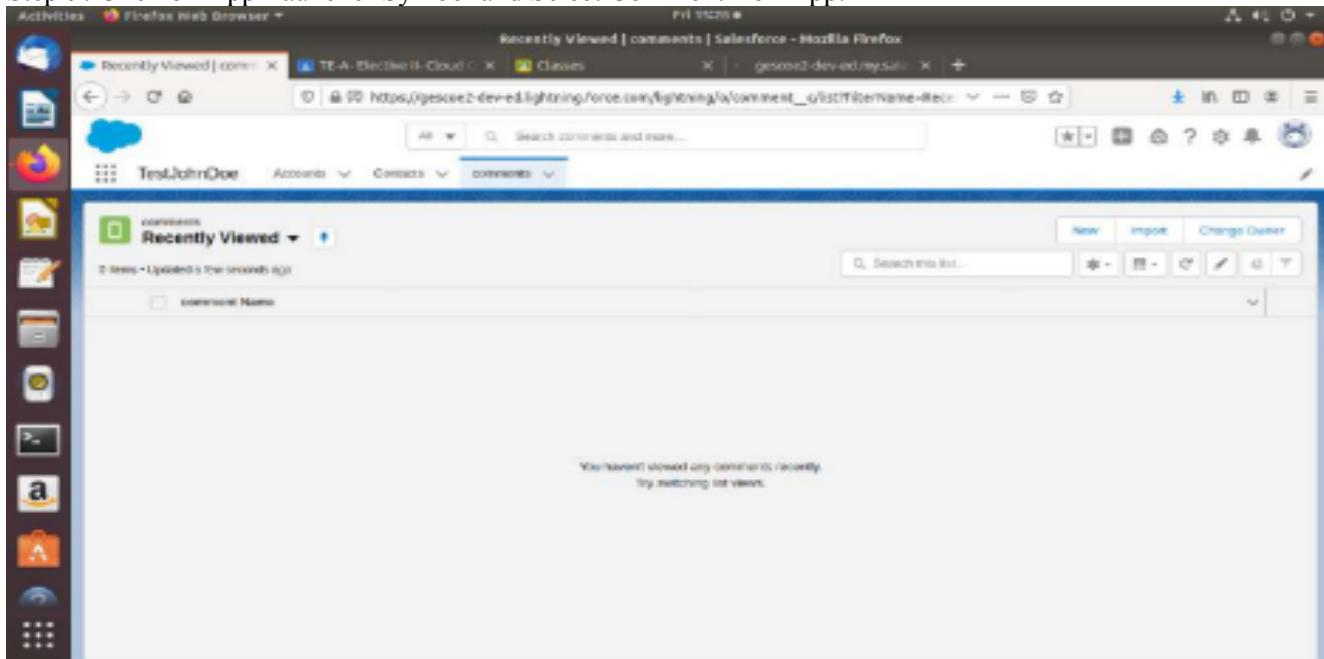
**Next**

Step-8: Enter name to app name,

- Click on Next-Next-Next,
- Select Items (Contacts,Comment)
- Click on Next.
- Select Profiles ( System Administrator) and move to the selected profile. - Click on Save and Finish.



Step-9: Click on App Launcher Symbol and Select Comment Box App.



Step-11: Tour the app

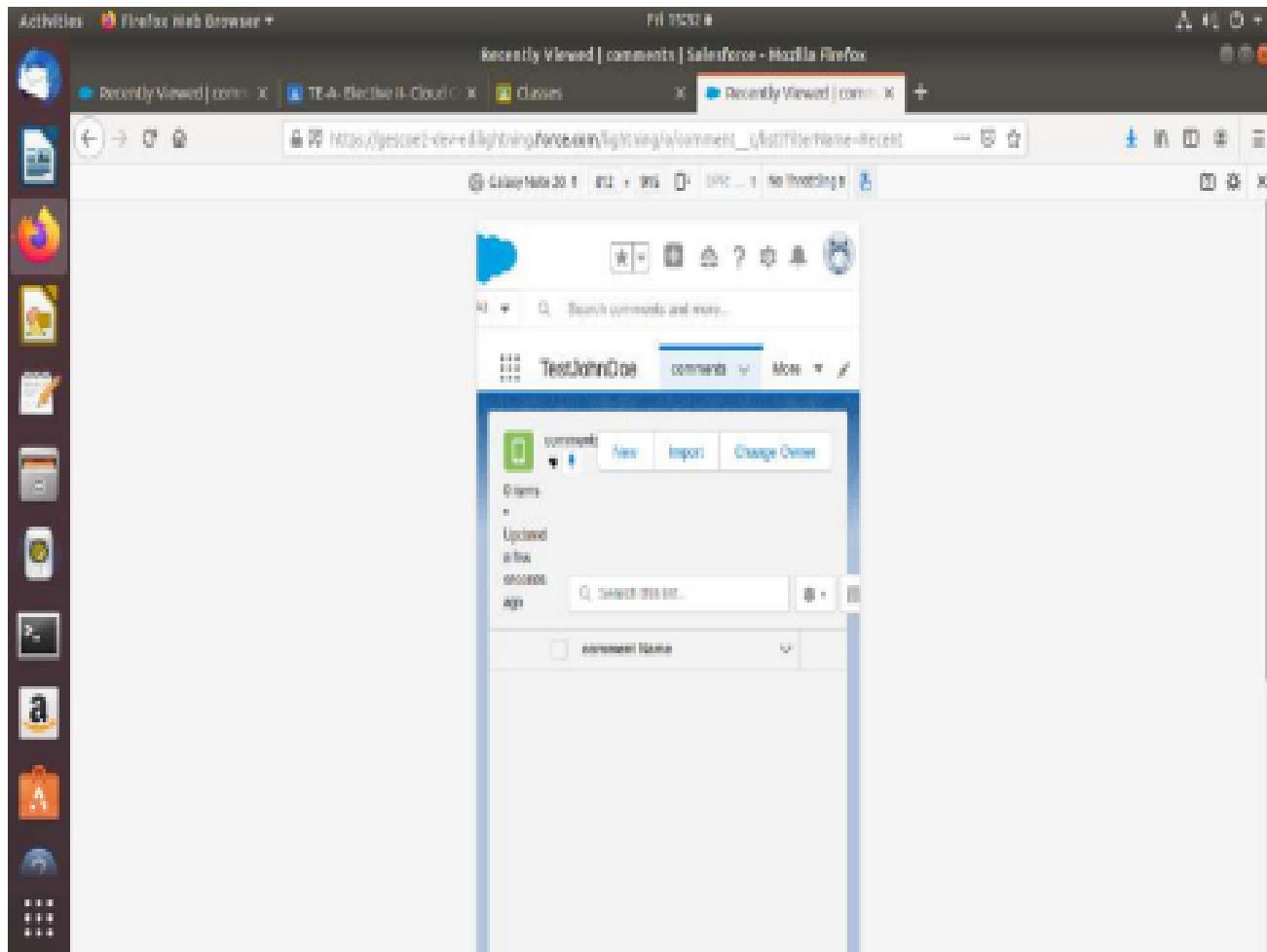
Step-12:

- Try out mobile app,
- Select Chrome developer tools,
- Open Chrome-Right Click on Chrome page- Select Inspect, -Click Toggle Device Mode Button to simulate your browser as a mobile device.

Step-13

- To simulate the sales force mobile app in your browser, copy and paste in url from the previous tab.

- Delete the part of the url immediately,
- Click on Left navigation bar,
- Find comment object under recent and click on it,
- Click new to create a comment.



**Conclusion:** Hence we have Studied custom Application (Mini Project) using Sales force.

### **Mini-Project**

Setup your own cloud for Software as a Service (SaaS) over the existing LAN in your laboratory. In this assignment you have to write your own code for cloud controller using open- source technologies to implement with **Web Application**. Implement the basic operations may be like to divide the file in segments/blocks and upload/ download file on/from cloud in encrypted form

Link for py file= <https://github.com/mayurpatil96k/laptop-price-prediction>