

SAKTHI POLYTECHNIC COLLEGE

SAKTHI NAGAR-638315

**DEPARTMENT
OF
COMPUTER ENGINEERING**



**4052550
CLOUD COMPUTING
&
INTERNET OF THINGS PRACTICAL**

INDEX

S.No	DATE	NAME OF THE EXPERIMENTS	MARKS	SIGNATURE
<u>PART-A CLOUD COMPUTING</u>				
1		IMPLEMENTATION OF SAAS USING WORD		
2		IMPLEMENTATION OF SAAS USING SPREADSHEET		
3		IMPLEMENTING WEB SERVICES		
4		INSTALLING GOOGLE APP ENGINE		
5		INSTALLING VIRTUAL BOX		
6		INSTALLING OPEN STACK		
7		CASE STUDY AMAZON EC2 AND AZURE		
<u>PART-B INTERNET OF THINGS</u>				
8		LED BLINK AND LED PATTERN		
9		LED PATTERN WITH PUSH BUTTON		
10		DISPLAYING HELLO WORLD		
11		IMPLEMENTING SERVO MOTOR CONTROL		
12		IMPLEMENTATION OF LM35 TEMPERATURE SENSOR & ULTRA SONIC DISTANCE MEASUREMENT WITH ARDUINO		
13		IMPLEMENTATION OF IR SENSOR WITH ANALOG INPUT		
14		USING THINKSPEAK CLOUD READING TEMPERATURE SENSOR MONITORING WITH NODEMCU		

Ex No: 1

Date:

PROGRAM ON SaaS TO CREATE WORD DOCUMENT

Aim:

To implement program on SaaS to Create an word document of your class timetable and store locally and on cloud with doc and PDF format

Procedure:

1. With Google Docs, you can create and edit text documents right in your web browser—no special software is required. Even better, multiple people can work at the same time, you can see people's changes as they make them, and every change is saved automatically.
2. To start, you need a document to work with. In this section

You learn how to:

- a. Create a new document
- b. Import and convert old documents to Docs

Create a new document

(i) You can create a new document right in Docs or in Google Drive. In Docs, click Create new document.

(ii) In Drive, click New > Google Docs > Blank document or From a template.
Import and convert old documents to Docs

(iii) If you have existing text documents, such as Microsoft Word or Adobe PDF files, you can import and convert them to Docs.

- Go to Drive.
- Click New > File Upload and choose a text document from your computer. Supported files include .doc, .docx, .dot, .html, plain text (.txt), .odt, and .rtf.
- Right-click the file you want to convert and select Open with > Google Docs. Converting your document from another program creates a copy of your original file in Docs format. You can then edit it in your browser like any other document.

Create Class timetable

Share documents

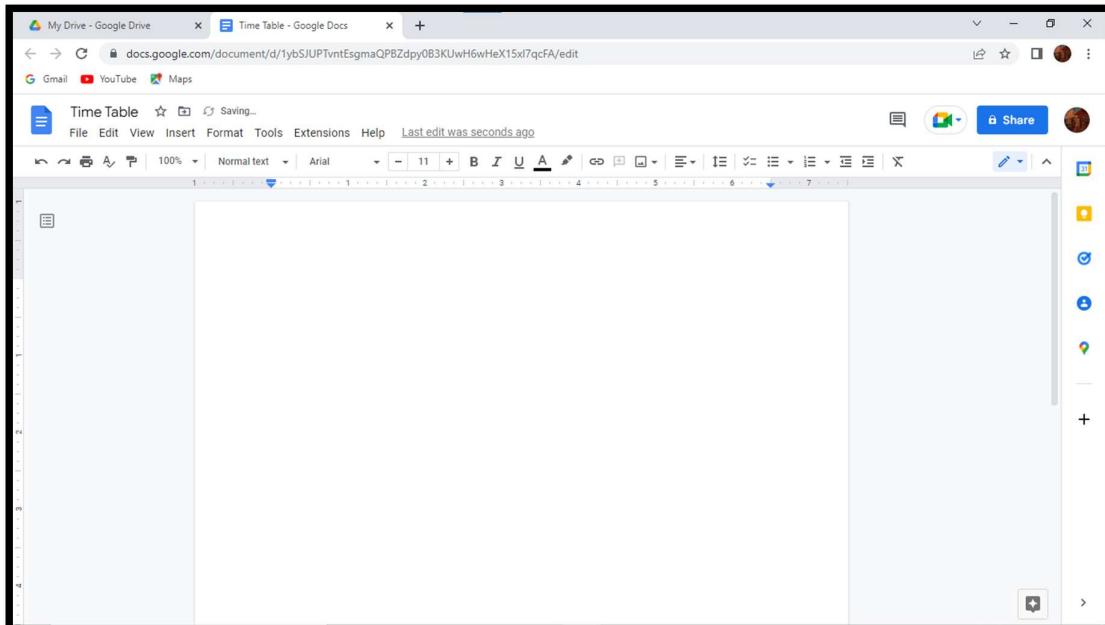
1. Open the file you want to share.
2. Click share.
3. Enter the email addresses or Google Groups you want to share with.

Note: If you can't add people outside your company, see your G Suite administrator.

4. Choose what kind of access you want to grant people:
 - **Can edit**—Collaborators can add and edit content as well as add comments.
 - **Can comment**—Collaborators can add comments, but not edit content.
 - **Can view**—People can view the file, but not edit or add comments. Click Send.

Everyone you shared the document with receives an email with a link to the document.

OUTPUT:



A screenshot of a Google Docs Word document titled "Time Table.docx". The document contains a "CLASS TIME TABLE" section with a grid for theoritical and practical classes across five days. Below this is a "THEORY" section with a table for practicals, and a "PRACTICALS:" section with a table for theory. The document also includes sections for "Department/Course", "Class In-charge", and "Year / Semester". A note at the top states: "Some Word features can't be displayed in Google Docs and will be dropped if you make changes".

RESULT:

Thus the word document is created, stored and accessed using cloud

Ex No: 2

Date:

PROGRAM ON SAAS TO CREATE SPREADSHEET

Aim:

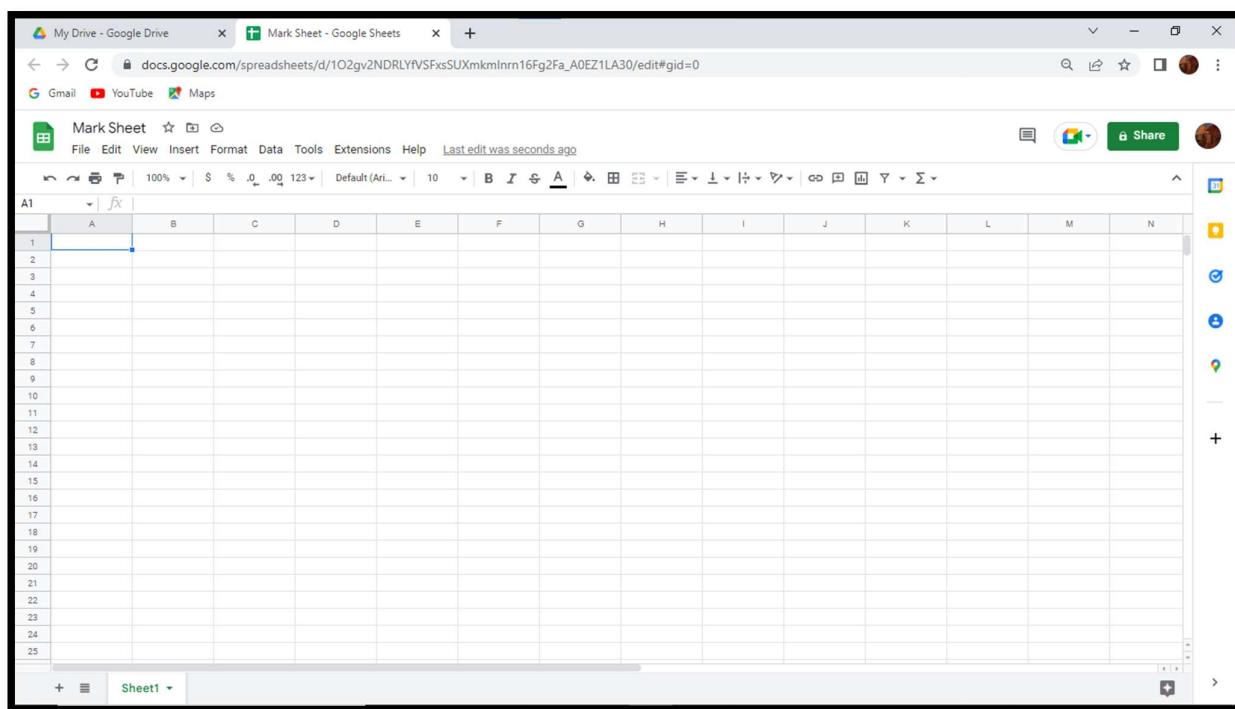
To implement program on SaaS to Create a spread sheet to generate a marksheet for student progress report

Procedure:

If you're accustomed to creating your spread sheets using an office suite or software like Microsoft Excel, you won't have any issue in creating a Google Spread sheet. Google Spread sheet works the same as Excel, and you can do most of the important spread sheet tasks with it. You can use Google Spread sheet directly from your web browser or from its mobile app

1. Sign into Google Sheets. Visit docs.google.com/spreadsheets and sign in with your Google or Gmail account. Your Gmail account gives you free access to Google Sheets.
2. View your existing sheets. Upon logging in, you will be brought to the main directory. If you already have existing spreadsheets, you can see and access them from here.
3. Create a new spreadsheet. Click the large red circle with a plus sign on the lower right corner. A new window or tab will be opened with the web-based spreadsheet.
4. Name the spreadsheet. "Untitled spreadsheet" appears on the top left corner. This is the current name of the spreadsheet. Click on it, and a small window will appear. Type in the name of the spreadsheet here, and click the "OK" button. You will see the name immediately change.

OUTPUT:



My Drive - Google Drive Mark Sheet.xlsx - Google Sheets

docs.google.com/spreadsheets/d/1aMPsC7nkcjhbtr4p-8ZpBjVC7R4cjU/edit#gid=818432202

Gmail YouTube Maps

Mark Sheet **.XLSX** in cloud computing

File Edit View Insert Format Data Tools Help Last edit was 4 hours ago

A1:11 SAKTHI POLYTECHNIC COLLEGE ,SAKTHI NAGAR - 638315

1 SAKTHI POLYTECHNIC COLLEGE ,SAKTHI NAGAR - 638315

2 COMPUTER ENGINEERING (2021-2022)

3 BOARD EXAMINATION MARK ANALYSIS - APRIL 2022

4 YEAR / SEM : II / IV

S.NO.	Roll No	NAME	CA	WP	JAVA	RDBM	WP LAB	JAVA LAB	RDBMS LAB	TOTAL	RANK
6	1	874	NAVEEN KUMAR T	99	96	76	93	99	97	95	655 1
7	2	801	ANANDHI E	88	86	83	89	95	93	98	632 2
8	3	810	KRISHNAMOORTHI T	80	87	82	87	98	95	99	628 3
9	4	821	SUDARSHAN P	54	99	79	95	100	100	99	626 4
10	5	803	DHARUN KUMAR J	69	93	83	78	99	98	99	619 5
11	6	814	PASUPATHI RAJ KUMAR	84	75	72	92	100	96	100	619 5
12	7	873	KOKILA M	77	93	57	92	98	94	99	610 7
13	8	872	KAVIN SANKAR P	77	69	69	87	100	99	99	600 8
14	9	871	DEEPAK RAGUL A	72	66	68	85	97	98	98	584 9
15	10	824	YOGESHWARI K	86	56	68	86	96	78	98	568 10
16	11	813	PANJAVARANAM S	88	83	76	66	84	91	78	566 11
17	12	823	VIGNESH N	79	60	58	48	91	98	86	520 12
18	13	806	KARTHIK K	63	71	53	63	93	76	74	493 13
19	14	808	KARUNESH M P	74	65	52	49	93	72	73	478 14
20	15	807	KARTHIKEYAN K S	47	70	51	49	94	78	71	460 15
21	16	804	DINESHKUMAR S	52	75	50	48	75	71	67	439 16
22		Total No. Students	24	24	24	24	24	24	24	24	
23		No. Of Students Present	23	23	23	23	23	23	23	23	
24		No. Of Students Absent	1	1	1	1	1	1	1	1	
25		No. Of Students Passes	22	21	20	19	23	23	23	23	
26		No. Of Students Fail	1	2	3	4	NIL	NIL	NIL	NIL	
27		Percentage (%)	95.65	91.3	86.96	82.61	100	100	100	100	

+ **Sheet1** > Explore

RESULT:

Thus the Program on saas is implemented by creating spreadsheet and details are entered successfully.

Ex No:3
Date:

BLOGSPOT CREATION

Aim:

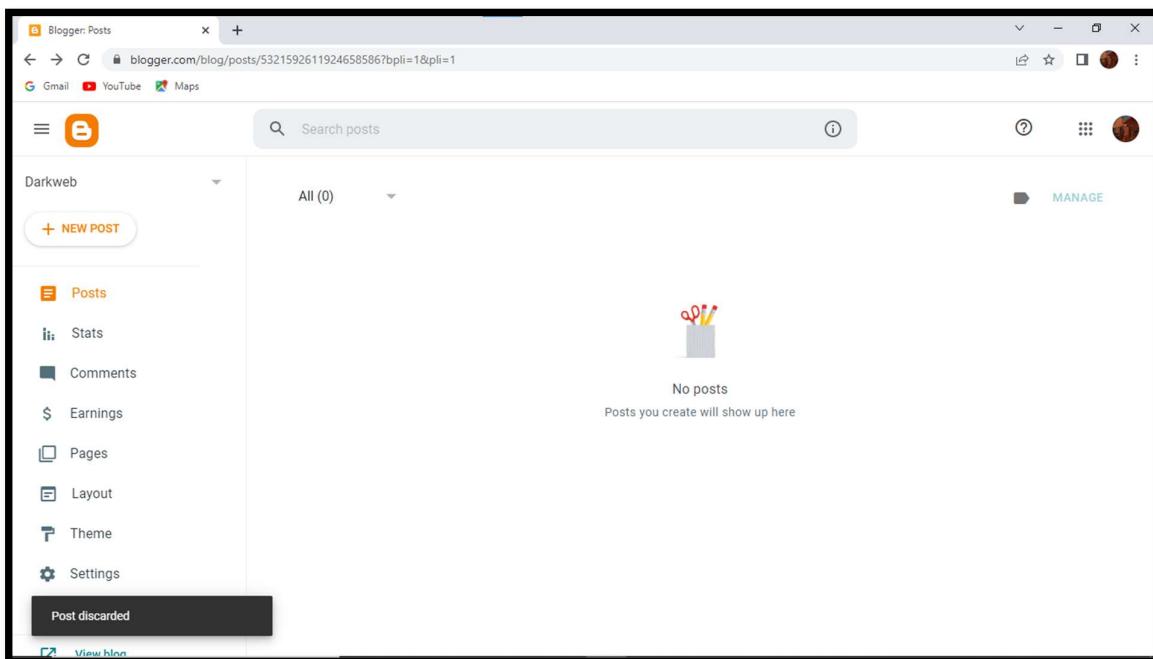
To implement web services by create your BlogSpot and Collaborating via Wikis.

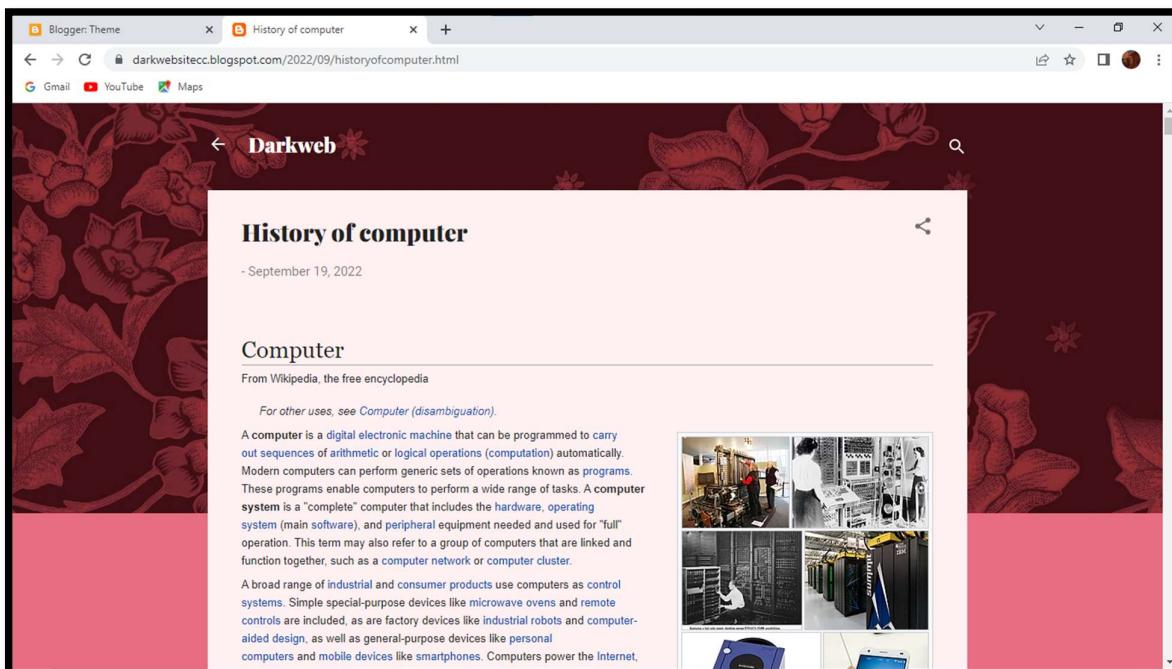
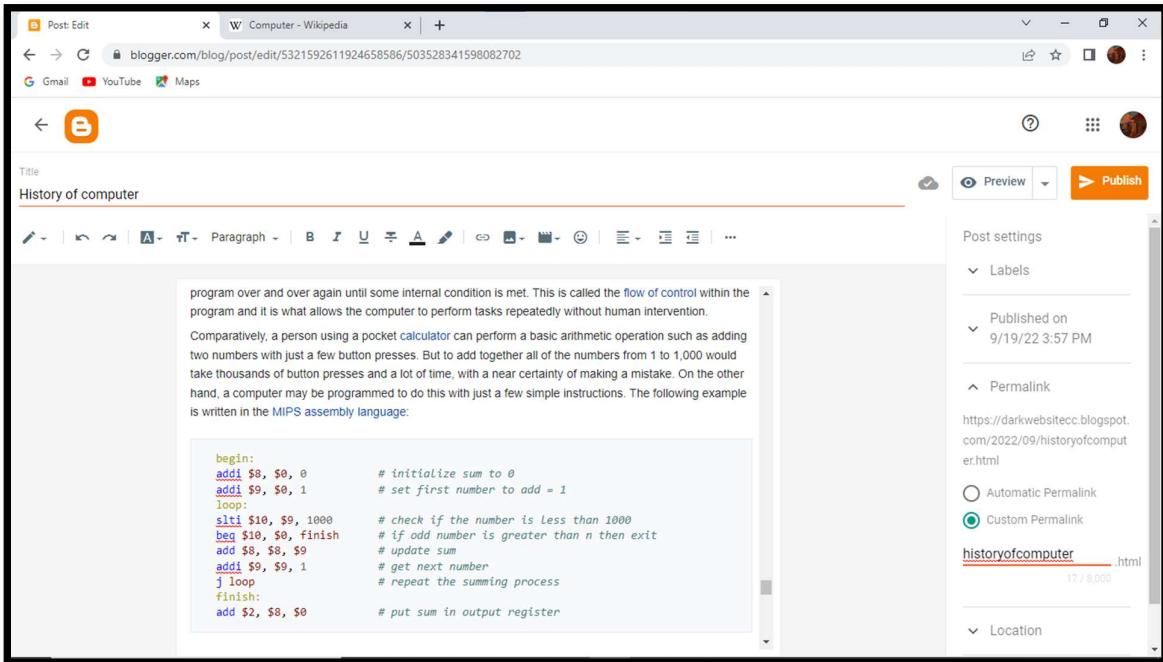
PROCEDURE:

CREATE A BLOG:

1. Search in Google as "blogger", there are many websites to create blogs. Let us take www.blogger.com.
2. Click on create blog. Sign in to your Google account
3. Choose a name for your blog and click on next.
4. Create the blog name click on next.
5. Type the display name and click on Finish
6. Click on create post and update your post in the displayed file and publish.
7. You can apply theme and create many post and labels in your blog and customize it.
8. Finally click on view blog from the menu and you can view the blog which you have created
9. In the blog site ,you can share on your own blog in any social medias and wikkis

OUTPUT:





RESULT:

Thus the log is created and collaborated successfully

Ex No:4

Date:

GOOGLE APP ENGINE

Aim:

To implement on PaaS to Install Google App Engine, create a program to validate user; create a database login(username, password)in mysql and deploy to cloud

Procedure:

- Create a Compute Engine instance
- Install MySQL
- Connect to MySQL

Create a Compute Engine instance

Create a new project in the Google Cloud console. You can use an existing project, but creating a new project makes cleanup easier.

You can complete all of the steps in this document using the Google Cloud console, but if you prefer to use the gcloud CLI, follow these steps to enable the Compute Engine API and install the Google Cloud CLI.

- Use the Google Cloud console to enable the Compute Engine API.
- Install the gcloud CLI
- Configure your workspace to make commands less verbose. Substitute your project's values for PROJECT_ID and ZONE in the following commands. For the full list of zones, see Available regions & zones.

```
gcloud config set project PROJECT_ID
```

```
gcloud config set compute/zone ZONE
```

Create a Compute Engine instance for MySQL and establish an SSH connection to the newly created instance. The default operating system is Debian version 10. If you prefer to use a different operating system for this tutorial, you can choose from the options described on the public images page in the Compute Engine documentation.

To create a Compute Engine instance in the Google Cloud console:

1. Open the Google Cloud console.
2. Select your newly created project and click **Continue**.
3. Click **Create instance** (**New instance** if you have existing instances). Name the instance **mysql-test**.
4. To specify an operating system other than the default value, in the **Boot disk** section, click **Change** to configure the properties for the boot disk. In the **Public images** tab, select an operating system and then click **Save**.
5. Click **Create**.

To establish an SSH connection:

1. On the **VM instances** page, find your new VM instance in the list.
2. In the **Connect** column, click **SSH**. The SSH terminal opens in a browser window.

Install MySQL

The following steps describe how to install MySQL on your Compute Engine instance.

Versions 10 and later of Debian contain MariaDB instead of MySQL as part of its package management system. MariaDB maintains compatibility with the MySQL protocol, but has an independently evolving feature set. For more details, see [MariaDB vs. MySQL](#).

To install MySQL, download the release package and manually install using the `dpkg` command.

1. Install the `wget` dependency.

```
sudo apt-get install -y wget
```

2. Download the MySQL Community Server release package.

```
export DEB_FILE=mysql-apt-config_0.8.20-1_all.debc /tmp  
curl -L --output ${DEB_FILE} \ https://dev.mysql.com/get/${DEB_FILE}
```

3. Verify the integrity of the release package file. cat >

```
 ${DEB_FILE}.md5 << EOL  
799bb0aefb93d30564fa47fc5d089aeb ${DEB_FILE}EOL  
md5sum --check ${DEB_FILE}.md5
```

4. The authenticity and integrity of the file are verified if you see the following output.

```
mysql-apt-config_0.8.20-1_all.deb: OK
```

5. After you have verified the file, add the MySQL package to the local package repository.

```
sudo dpkg -i ${DEB_FILE}
```

6. With the top MySQL Server & Cluster menu option selected, press Return and then use the arrow keys to choose a server version.

7. This guide expects you to choose either MySQL 8.0 or 5.7. Press Return on your keyboard after you have selected the version.

8. When you are satisfied with the options selected in the configuration menu, use the arrow keys to select Ok in the menu and press Return on your keyboard.

9. Update the package cache.

```
sudo apt-get update
```

10. Install MySQL. The installation process starts the MySQL service for you.

```
sudo apt-get -y install mysql-community-server
```

11. You are prompted to provide some details for the installation such as the root password.

Connect to MySQL

1. Connect to MySQL using the MySQL client.

```
sudo mysql -u root -p
```

When you connect to MySQL, the prompt changes to mysql>

You can then run MySQL commands. For example, the following command shows the threads running, including the current connection.

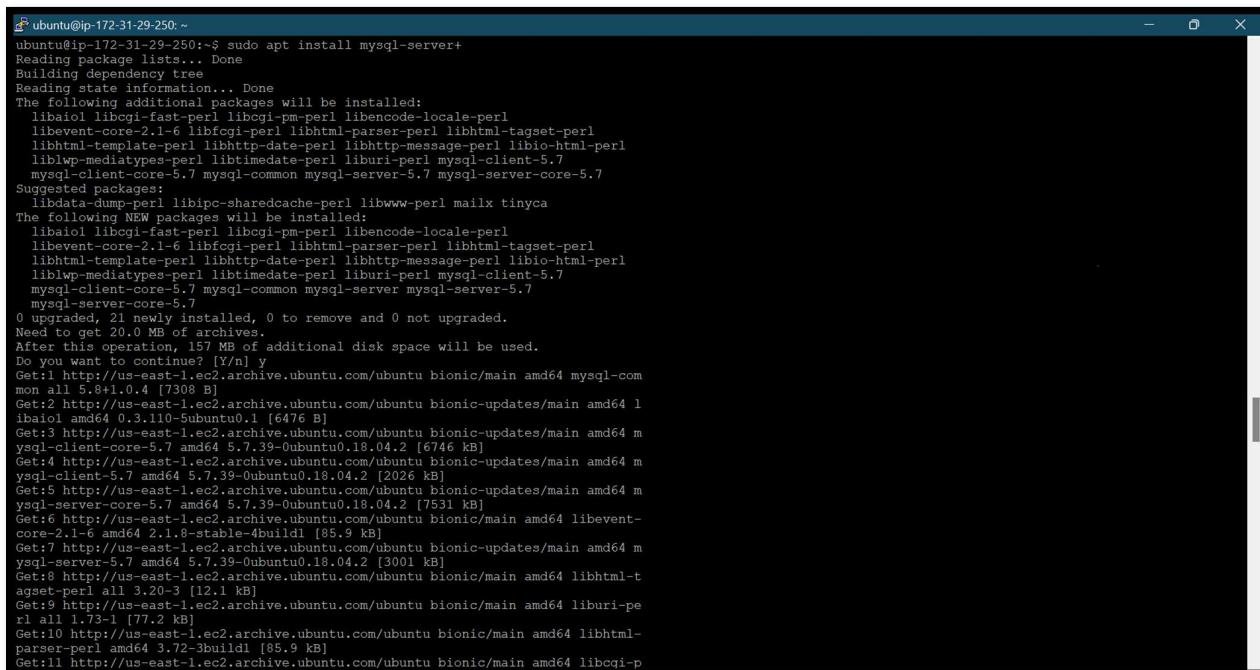
```
Mysql> SHOW processlist;
```

You can use the following command to generate a list of users.

```
Mysql> SELECT User, Host, authentication_string FROM mysql.user;
```

```
Mysql> exit
```

OUTPUT:



```
ubuntu@ip-172-31-29-250:~$ sudo apt install mysql-server+
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
libaio1 libcgifast-perl libcgipm-perl libencode-locale-perl
libevent-core-2.1-6 libfcgi-perl libhtml-parser-perl libhtml-tagset-perl
libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl
liblwp-mediatypes-perl libtimedate-perl liburi-perl mysql-client-5.7
mysql-client-core-5.7 mysql-common mysql-server-5.7 mysql-server-core-5.7
Suggested packages:
libdata-dump-perl libipc-sharedcache-perl libwww-perl mailx tinyca
The following NEW packages will be installed:
libaio1 libcgifast-perl libcgipm-perl libencode-locale-perl
libevent-core-2.1-6 libfcgi-perl libhtml-parser-perl libhtml-tagset-perl
libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl
liblwp-mediatypes-perl libtimedate-perl liburi-perl mysql-client-5.7
mysql-client-core-5.7 mysql-common mysql-server mysql-server-5.7
mysql-server-core-5.7
0 upgraded, 21 newly installed, 0 to remove and 0 not upgraded.
Need to get 20.0 MB of archives.
After this operation, 157 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/main amd64 mysql-com
mon all 5.8+1.0.4 [7300 B]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/main amd64 l
ibai1 amd64 0.3.110-5ubuntu0.1 [6476 B]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/main amd64 m
ysql-client-core-5.7 amd64 5.7.39-0ubuntu0.18.04.2 [6746 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/main amd64 m
ysql-client-5.7 amd64 5.7.39-0ubuntu0.18.04.2 [2026 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/main amd64 m
ysql-server-core-5.7 amd64 5.7.39-0ubuntu0.18.04.2 [7531 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/main amd64 libevent-
core-2.1-6 amd64 2.1.8-stable-4ubuntu1 [85.9 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/main amd64 m
ysql-server-5.7 amd64 5.7.39-0ubuntu0.18.04.2 [3001 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/main amd64 libhtml-t
agset-perl all 3.20-3 [12.1 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/main amd64 liburi-pe
rl all 1.73-1 [77.2 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/main amd64 libhtml-
parser-perl amd64 3.72-3ubuntu1 [85.9 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/main amd64 libcapi-p
```

```
ubuntu@ip-172-31-29-250: ~
NO)
ubuntu@ip-172-31-29-250:~$ mysql --version
mysql Ver 14.14 Distrib 5.7.39, for Linux (x86_64) using EditLine wrapper
ubuntu@ip-172-31-29-250:~$ sudo mysql_secure_installation

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD PLUGIN can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD plugin?

Press y|Y for Yes, any other key for No: y

There are three levels of password validation policy:

LOW  Length >= 8
MEDIUM Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary
file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 0
Please set the password for root here.

New password:
Re-enter new password:

Estimated strength of the password: 50
Do you wish to continue with the password provided?(Press y|Y for Yes, any other
key for No) : y
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : y
Success.


```

```
ubuntu@ip-172-31-29-250: ~
Remove anonymous users? (Press y|Y for Yes, any other key for No) : y
Success.

Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y
Success.

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No)
: n
... skipping.
Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

All done!
ubuntu@ip-172-31-29-250:~$ sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 6
Server version: 5.7.39-Ubuntu0.18.04.2 (Ubuntu)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

RESULT:

Thus mysql is installed successfully in Google compute engine using Paas services

Ex No:5
Date:

LINUX INSTALLATION USING VMWARE

Aim:

To Install VMware Workstation with different flavours of Linux or Windows OS on top of windows 7 or 8

Procedure:

Step 1: Install VMware

Step 2: Download ISO file from linux website

Step 3: Open VMWare

Step 4: Click "Open a Virtual Machine "

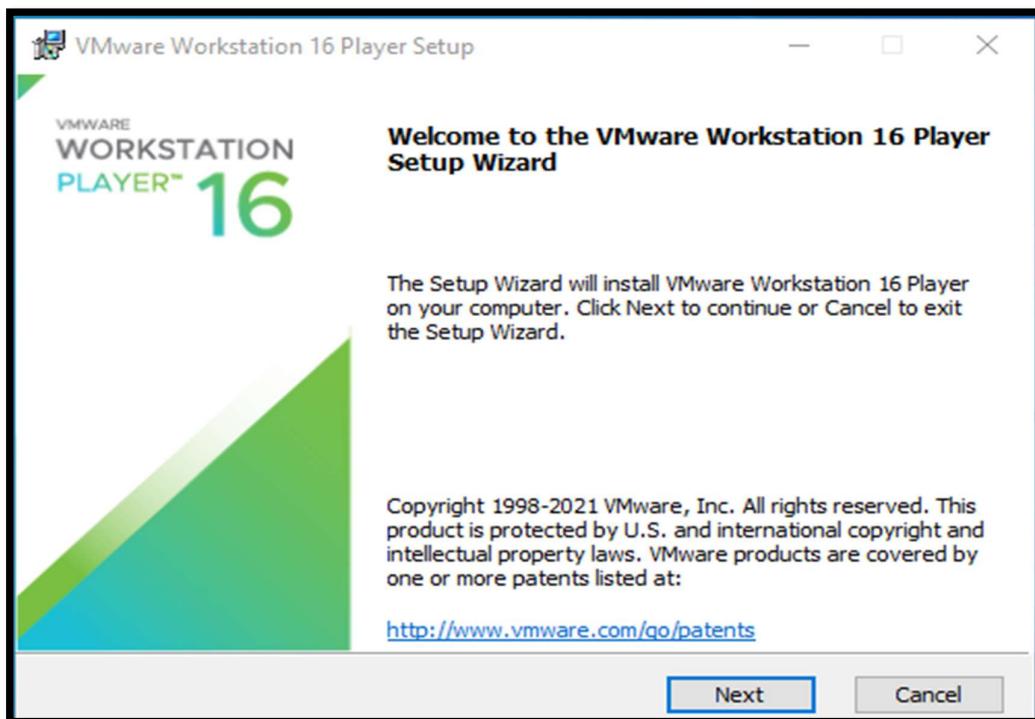
Step 5: Brows VM ware File in File Manager

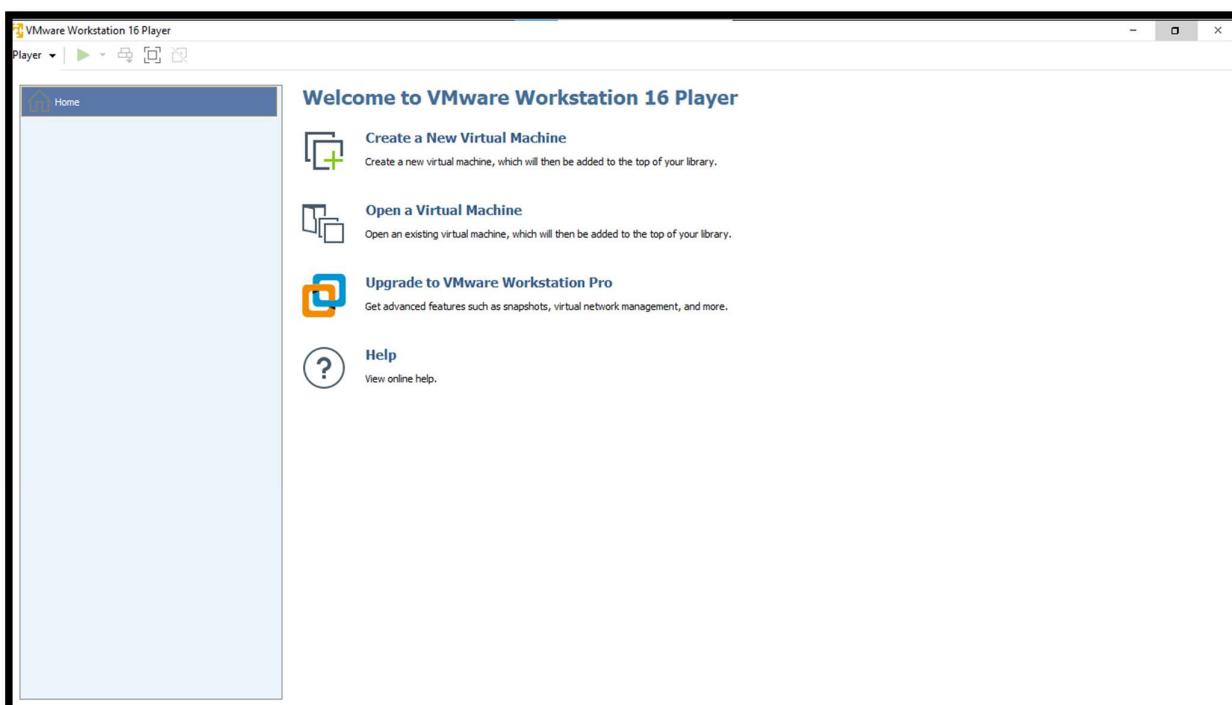
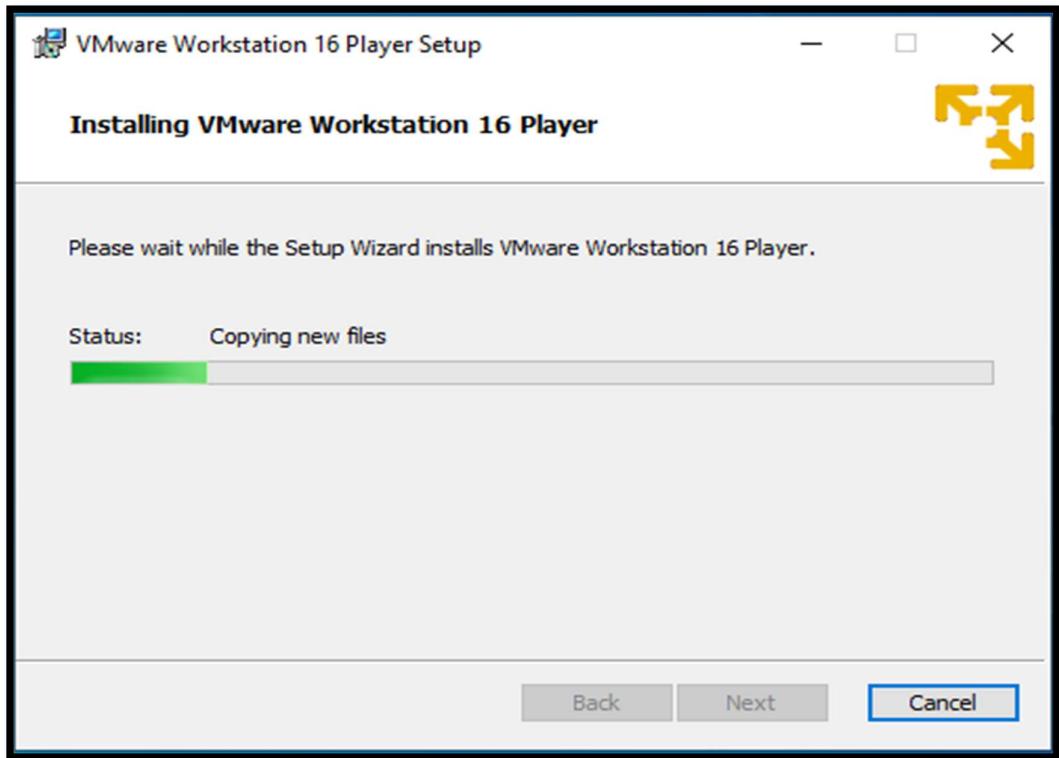
Step 6: Click Open

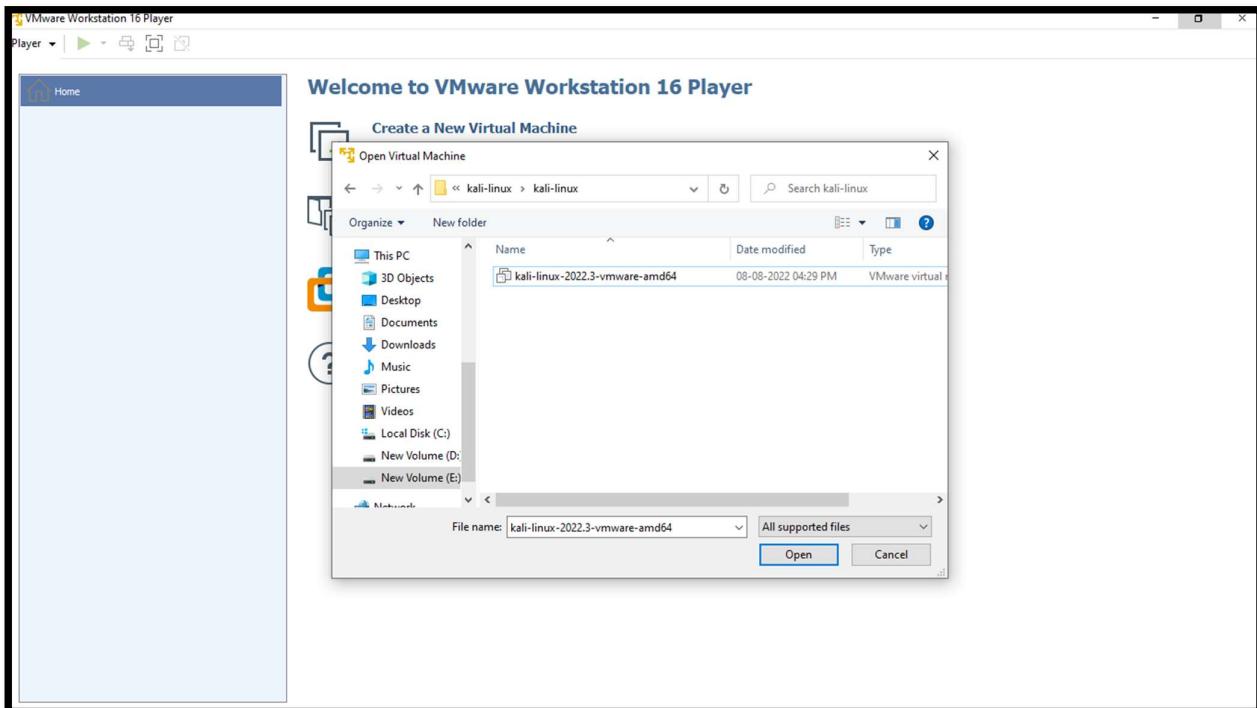
Step 7: Click Run Virtual Machine.

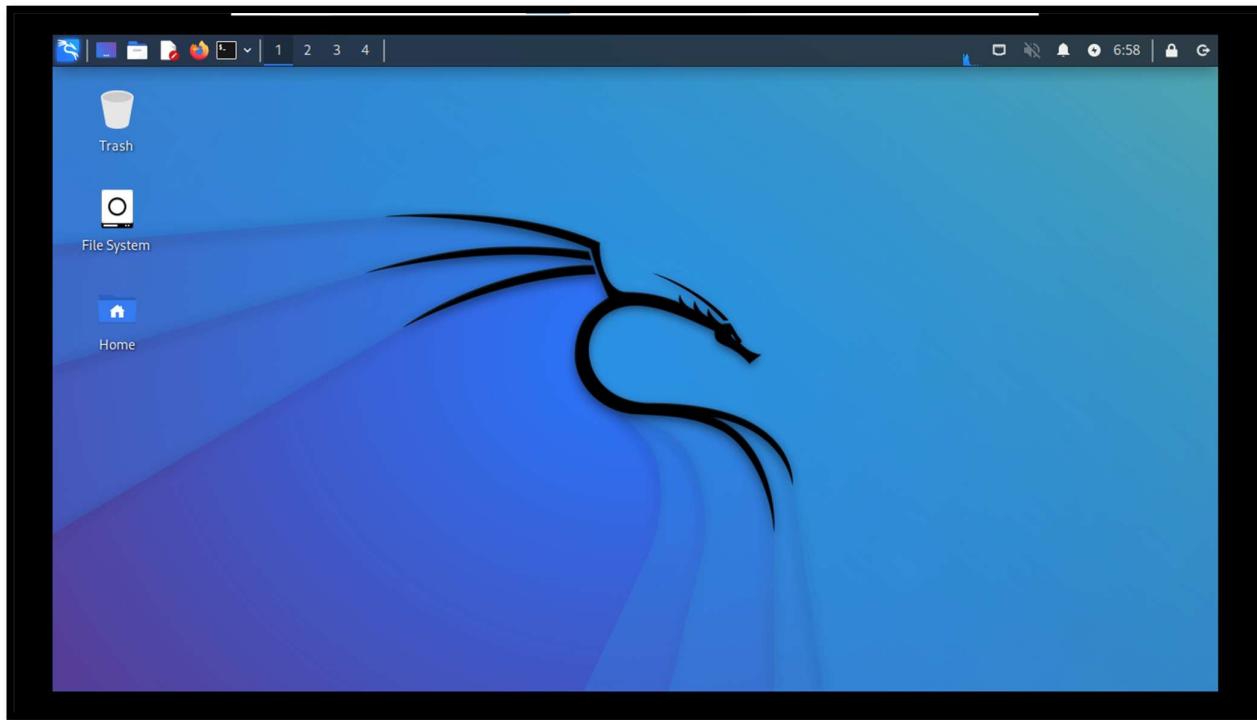
Your OS will Installed Successfully

OUTPUT:









RESULT:

Thus the Linux os is installed in the virtual box successfully.

Ex No:6
Date:

OPEN STACK INSTALLATION

Aim:

To install the open stack and use it as a infrastructure as a service and Use technology own cloud

Procedure:

Steps to install openstack on Ubuntu 18.04 in Virtual box

- 1)First install virtual box <https://www.virtualbox.org/wiki/Downloads>...
- 2)Download Ubuntu 18.04 ISO <https://releases.ubuntu.com/18.04/>
- 3)Create Ubuntu 18.04 VM in Virtual box 4 GB RAM + 2 vCPUs Hard disk capacity of Min 10 GB
- 4)After successful installation of ubuntu ,start Openstack installation

Execute below steps in terminal:

```
sudo snap install microstack -beta  
snap list microstack  
sudo microstack init --auto --control  
sudo apt install net-tools
```

```
ifconfig
```

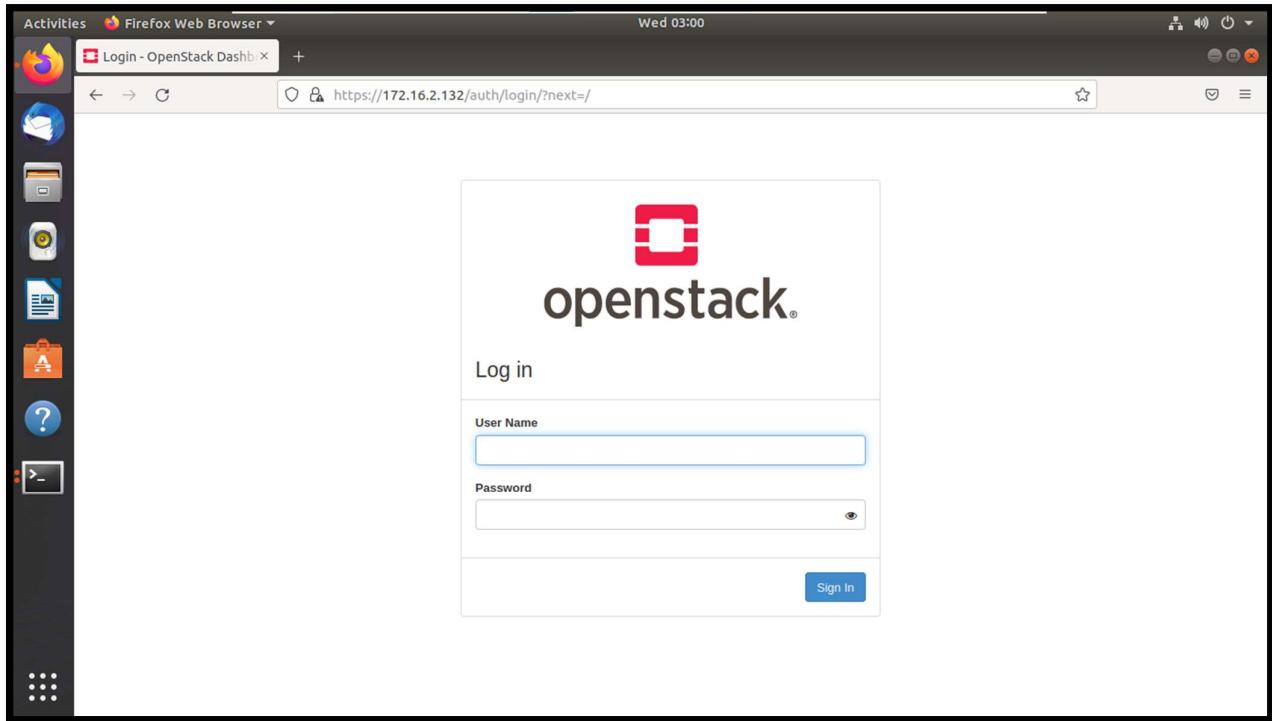
After Successfully installation of Openstack access the Horizon dashboard with below URL:

<http://10.20.20.1/dashboard>

OUTPUT:

```
Activities Terminal Wed 02:23 pvs@ubuntu:~  
File Edit View Search Terminal Help  
pvs@ubuntu:~$ sudo snap install microstack --beta  
[sudo] password for pvs:  
Download snap "microstack" (245) from channel "beta"  
4% 2.87MB/s 2m26s
```

```
Activities Terminal Wed 02:59 pvs@ubuntu:~  
File Edit View Search Terminal Help  
pvs@ubuntu:~$ sudo snap install microstack --beta  
[sudo] password for pvs:  
microstack (beta) ussurি from Canonical✓ installed  
pvs@ubuntu:~$ snap list microstack  
Name Version Rev Tracking Publisher Notes  
microstack ussurি 245 latest/beta canonical✓ -  
pvs@ubuntu:~$ sudo microstack init --auto --control  
2022-09-21 02:34:15,688 - microstack_init - INFO - Configuring clustering ...  
2022-09-21 02:34:15,861 - microstack_init - INFO - Setting up as a control node.  
2022-09-21 02:34:18,671 - microstack_init - INFO - Generating TLS Certificate and Key  
2022-09-21 02:34:19,712 - microstack_init - INFO - Configuring networking ...  
2022-09-21 02:34:25,885 - microstack_init - INFO - Opening horizon dashboard up to *  
2022-09-21 02:34:26,784 - microstack_init - INFO - Waiting for RabbitMQ to start ...  
Waiting for 172.16.2.132:5672  
2022-09-21 02:34:33,185 - microstack_init - INFO - RabbitMQ started!  
2022-09-21 02:34:33,186 - microstack_init - INFO - Configuring RabbitMQ ...  
2022-09-21 02:34:34,097 - microstack_init - INFO - RabbitMQ Configured!  
2022-09-21 02:34:34,133 - microstack_init - INFO - Waiting for MySQL server to start ...  
Waiting for 172.16.2.132:3306  
2022-09-21 02:34:39,541 - microstack_init - INFO - Mysql server started! Creating databases ...  
2022-09-21 02:34:41,261 - microstack_init - INFO - Configuring Keystone Fernet Keys ...  
2022-09-21 02:34:52,661 - microstack_init - INFO - Bootstrapping Keystone ...  
2022-09-21 02:35:04,435 - microstack_init - INFO - Creating service project ...  
2022-09-21 02:35:09,924 - microstack_init - INFO - Keystone configured!  
2022-09-21 02:35:09,953 - microstack_init - INFO - Configuring the Placement service...  
2022-09-21 02:35:30,141 - microstack_init - INFO - Running Placement DB migrations...  
2022-09-21 02:35:33,371 - microstack_init - INFO - Configuring nova control plane services ...  
2022-09-21 02:35:45,778 - microstack_init - INFO - Running Nova API DB migrations (this may take a lot of time)...  
2022-09-21 02:36:08,072 - microstack_init - INFO - Running Nova DB migrations (this may take a lot of time)...  
Waiting for 172.16.2.132:8774  
2022-09-21 02:36:57,338 - microstack_init - INFO - Creating default flavors...  
2022-09-21 02:37:24,152 - microstack_init - INFO - Configuring nova compute hypervisor ...  
2022-09-21 02:37:24,152 - microstack_init - INFO - Checking virtualization extensions presence on the host  
2022-09-21 02:37:24,195 - microstack_init - WARNING - Unable to determine hardware virtualization support by CPU vendor id "GenuineIntel": assuming it is not supported.  
2022-09-21 02:37:24,195 - microstack_init - WARNING - Hardware virtualization is not supported - software emulation will be used for Nova instances  
2022-09-21 02:37:26,845 - microstack_init - INFO - Configuring the Spice HTML5 console service...
```



RESULT:

Thus the open stack is installed successfully in the virtual box

Ex No:7

Date:

CASE STUDY Amazon EC2 and Azure

AIM:

Case study on open source and commercial cloud.

THEORY:

Amazon Web Services (AWS) and Microsoft Azure are the two giants in the world of cloud computing.

While AWS is the largest cloud computing platform, Microsoft Azure is the fastest-growing and second-largest.

Azure is a cloud computing platform and an online portal that allows you to access and manage cloud services and resources provided by Microsoft. These services and resources include storing your data and transforming it, depending on your requirements. To get access to these resources and services, all you need to have is an active internet connection and the ability to connect to the Azure portal.

Azure provides more than 200 services, are divided into 18 categories. These categories include computing, networking, storage, IoT, migration, mobile, analytics, containers, artificial intelligence, and other machine learning, integration, management tools, developer tools, security, databases, DevOps, media identity, and web services.

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) Cloud. Using Amazon EC2 eliminates your need to invest in hardware up front, so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. Amazon EC2 enables you to scale up or down to handle changes in requirements or spikes in popularity, reducing your need to forecast traffic.

Ex No : 08

Date:

IMPLEMENT LED BLINK AND LED PATTERN WITH ARDUINO

Aim:

To Implement LED Blink and LED Pattern With Arduino

Procedure:

STEP 1: Open Arduino Software.

STEP 2: Click Tools -> Board -> Arduino Nano.

STEP 3: In Tools Menu Select Port -> COM.

STEP 4: Click File-> New and Start Write the Code.

STEP 5: And Click tick Icon Button of the file Menu to compile.

STEP 6: Click Upload Icon After Verify.

Program: LED Blink

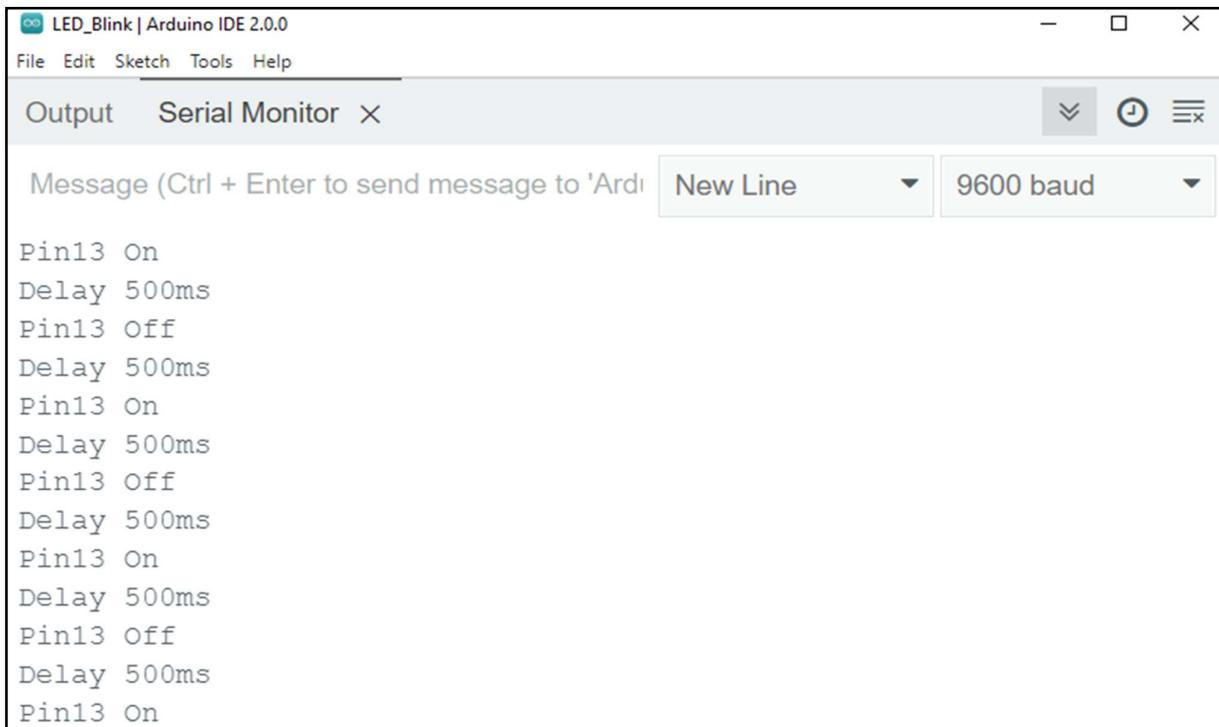
```
#include <LiquidCrystal.h>
LiquidCrystal lcd(A5,A4,A3,A2,A1,A0);
int i;
void setup() {
    // set up the LCD's number of columns and rows:
    lcd.begin(16, 2);
    lcd.print("LED BLINK");
    lcd.setCursor(0,1);
    lcd.print("EXPERIMENT");
    delay(3000);
}
void led_blink()
{
    lcd.clear();
    lcd.print("LED Blinking");
    pinMode(5,OUTPUT);

    for(int i=0;i<10;i++)
    {
        digitalWrite(1,HIGH);
        delay(50);
    }
}
void loop() {
    led_blink();
    delay(100);
}
```

Program: LED Pattern

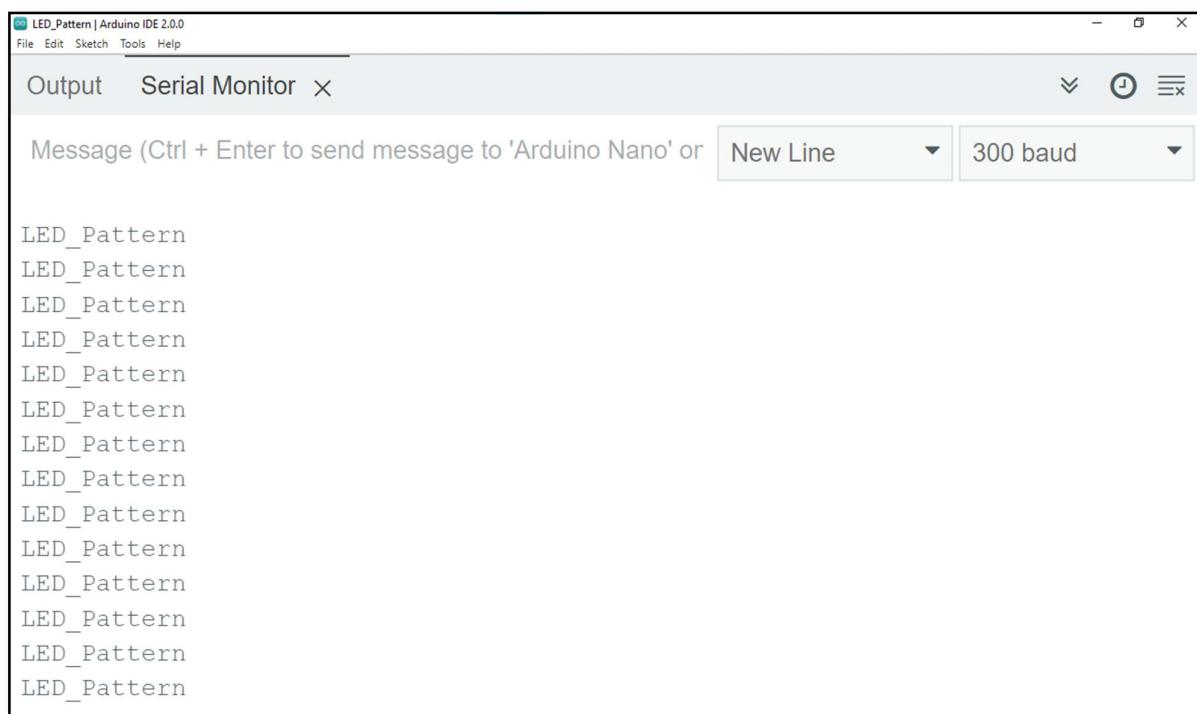
```
#include <LiquidCrystal.h>
LiquidCrystal lcd(A5,A4,A3,A2,A1,A0);
int i;
void setup() {
  lcd.begin(16, 2);
  lcd.print("LED PATTERN ");
  lcd.setCursor(0,1);
  lcd.print("Experimenter");
  delay(3000);
  lcd.clear();
}
void ledPattern()
{
  lcd.clear();
  lcd.print("LED Pattern");
  for (i =2;i<=7;i++)
  {
    pinMode(i, OUTPUT); // turn the LED on (HIGH is the voltage level)
    digitalWrite(i, HIGH);
  }
  for(int j=0;j<5;j++)
  {
    for (i =2;i<=7;i++)
    {
      digitalWrite(i, LOW);
      delay(100);
    }
    for (i =7;i>=2;i--)
    {
      digitalWrite(i, HIGH);
      delay(100);
    }
  }
  for (i =2;i<=7;i++)
  {
    digitalWrite(i, HIGH);
  }
  for (i =2;i<=7;i++)
  {
    pinMode(i, INPUT);
  }
  void loop() {
    ledPattern();
    delay(1000);
  }
}
```

Output:



The screenshot shows the Arduino IDE 2.0.0 interface with the title bar "LED_Blink | Arduino IDE 2.0.0". The menu bar includes File, Edit, Sketch, Tools, and Help. The main window is titled "Output" and displays the serial monitor output. The output text is as follows:

```
Pin13 On
Delay 500ms
Pin13 Off
Delay 500ms
Pin13 On
Delay 500ms
Pin13 Off
Delay 500ms
Pin13 On
Delay 500ms
Pin13 Off
Delay 500ms
Pin13 On
```



The screenshot shows the Arduino IDE 2.0.0 interface with the title bar "LED_Pattern | Arduino IDE 2.0.0". The menu bar includes File, Edit, Sketch, Tools, and Help. The main window is titled "Output" and displays the serial monitor output. The output text is as follows:

```
LED_Pattern
```

Result:

Thus the above LED Blink and LED Pattern with Arduino program was executed and verified successfully

Ex No : 09

Date:

IMPLEMENT LED PATTERN WITH PUSH BUTTON CONTROL WITH ARDUINO

Aim:

Implement LED Pattern with Push Button Control with Arduino

Procedure:

STEP 1: Open Arduino Software.

STEP 2: Click Tools -> Board -> Arduino Nano.

STEP 3: In Tools Menu Select Port -> COM.

STEP 4: Click File-> New and Start Write the Code.

STEP 5: And Click tick Icon Button of the file Menu to compile.

STEP 6: Click Upload Icon After Verify.

Program:

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(A5,A4,A3,A2,A1,A0);
int i;
void setup() {

    lcd.begin(16, 2);
    lcd.print("LED Pattern");
    lcd.setCursor(0,1);
    lcd.print("Experimenter");
    delay(3000);
    lcd.clear();
}

void pushbutton_led()

{
    lcd.clear();
    lcd.print("Push Button");
    lcd.setCursor(0,1);
    lcd.print("Press Key6");

    pinMode(7, INPUT); // turn the LED on (HIGH is the voltage level)
    digitalWrite(7, HIGH);

    for (i =2;i<=6;i++)
    {
        pinMode(i, OUTPUT); // turn the LED on (HIGH is the voltage level)
        digitalWrite(i, HIGH);
    }

    while(digitalRead(7));

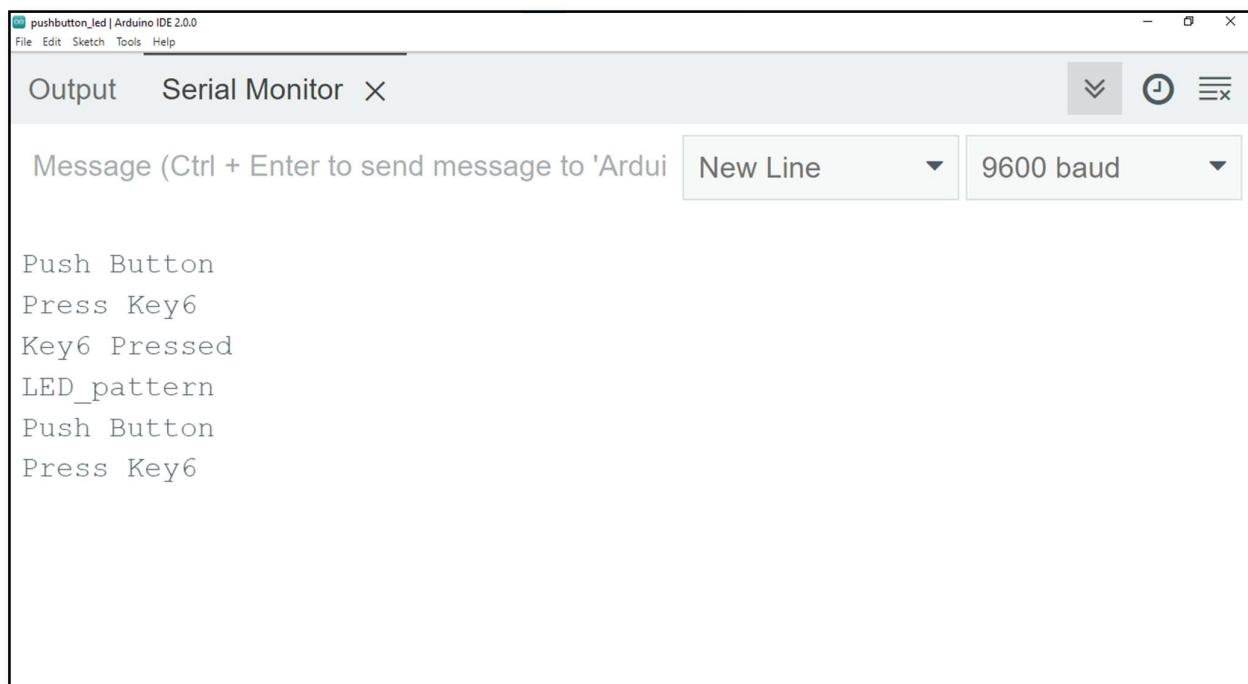
    if(!digitalRead(7))
    {
```

```
lcd.clear();
lcd.print("Key6 Pressed");
for (i =2;i<=6;i++)
{
  digitalWrite(i, LOW);
  delay(100);
}
for (i =6;i>=2;i--)
{
  digitalWrite(i, HIGH);
  delay(100);
}
for (i =2;i<=6;i++)
  digitalWrite(i, HIGH);
}

void loop() {

  pushbutton_led();
  delay(1000);
}
```

Output:



Result:

Thus the above LED Pattern with Push Button Control with Arduino program was executed and verified successfully

Ex No : 10

DISPLAY “HELLO WORLD” IN LCD 16X2 DISPLAY WITH ARDUINO

Aim:

To Display “Hello World” in LCD 16X2 Display with Arduino

Procedure:

STEP 1: Open Arduino Software.

STEP 2: Click Tools -> Board -> Arduino Nano.

STEP 3: In Tools Menu Select Port -> COM.

STEP 4: Click File-> New and Start Write the Code.

STEP 5: And Click tick Icon Button of the file Menu to compile.

STEP 6: Click Upload Icon After Verify.

Program:

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(A5,A4,A3,A2,A1,A0);
int i;
void setup() {
  lcd.begin(16, 2);
  lcd.print("TEXT DISPLAY");
  lcd.setCursor(0,1);
  lcd.print("Experimenter");
  delay(3000);
  lcd.clear();
  lcd.print("HELLO WORLD");
}
void loop() {
```

Output:



The screenshot shows the Arduino IDE interface with the title bar "lcd_hello_world | Arduino IDE 2.0.0". The menu bar includes File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with Output and Serial Monitor buttons. The main area is titled "Message (Ctrl + Enter to send message to 'Arduino Nano' on 'COM4')". The output window displays the text "Hello World" repeated 15 times, indicating successful execution of the program.

Result:

Thus the above Display “Hello World” in LCD 16X2 with Arduino program was executed and verified successfully

Ex No : 11

IMPLEMENT THE SERVO MOTOR CONTROL WITH ARDUINO

Aim:

To Implement the Servo Motor Control with Arduino

Procedure:

STEP 1: Open Arduino Software.

STEP 2: Click Tools -> Board -> Arduino Nano.

STEP 3: In Tools Menu Select Port -> COM.

STEP 4: Click File-> New and Start Write the Code.

STEP 5: And Click tick Icon Button of the file Menu to compile.

STEP 6: Click Upload Icon After Verify.

Program:

```
#include <LiquidCrystal.h>
#include <Servo.h>
Servo myservo;
LiquidCrystal lcd(A5,A4,A3,A2,A1,A0);
int i;
void setup() {
  lcd.begin(16, 2);
  lcd.print("ROTATE SERVO");
  lcd.setCursor(0,1);
  lcd.print("Experimenter");
  delay(3000);
  lcd.clear();
  lcd.print("START ROTATING");
  delay(1000);
  lcd.clear();
}
void servoSweep()
{
  lcd.clear();
  lcd.print("MOTER DEGREE");
  myservo.attach(11);
  int pos;
  for(int i=0; i<2;i++)
  {
    for (pos = 0; pos <= 90; pos += 1) {           // goes from 0 degrees to 180 degrees
      myservo.write(pos);                         // in steps of 1 degree
      delay(15);                                // tell servo to go to position in variable 'pos'
      lcd.setCursor(0,1);                         // waits 15ms for the servo to reach the position
      lcd.print(" ");
      lcd.setCursor(0,1);
      lcd.print(pos);
    }
    for (pos = 130; pos >= 0; pos -= 1) {           // goes from 180 degrees to 0 degrees
      myservo.write(pos);                         // tell servo to go to position in variable 'pos'
      delay(15);
    }
  }
}
```

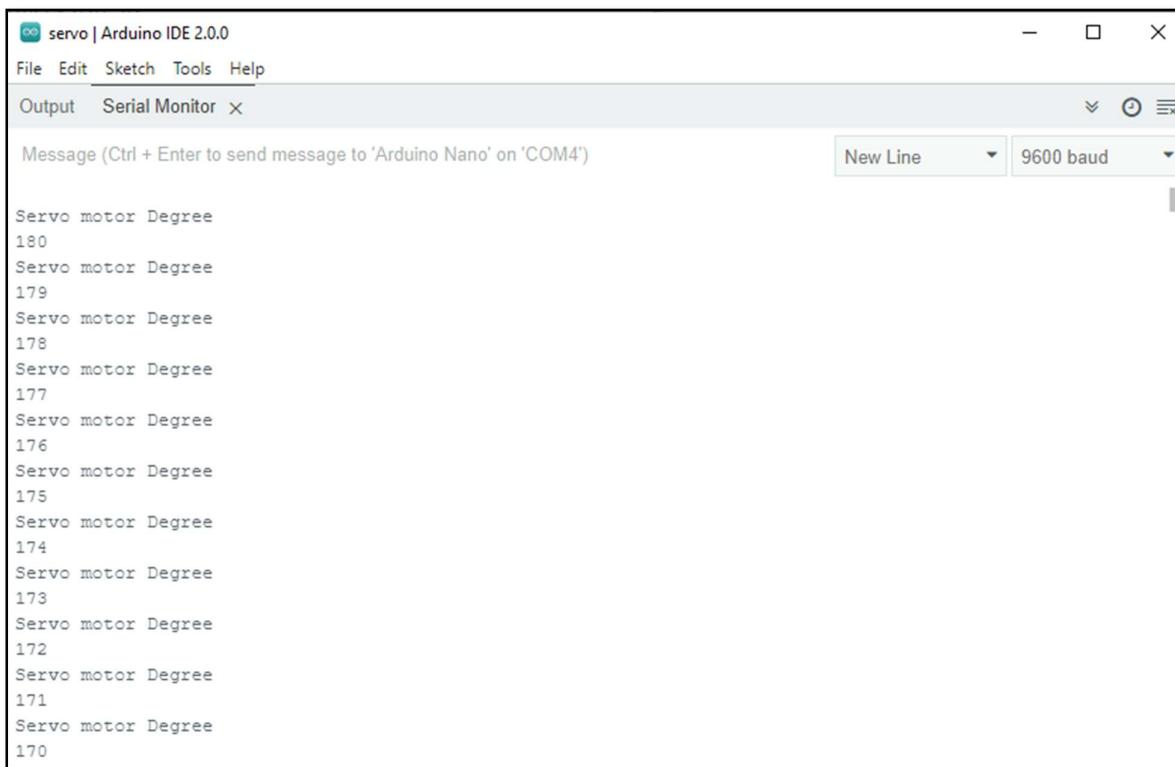
```

    delay(15); // waits 15ms for the servo to reach the
    position
    lcd.setCursor(0,1);
    lcd.print(" ");
    lcd.setCursor(0,1);
    lcd.print(pos);
}
}
}

void loop() {
    servoSweep();
    delay(1000);
}

```

Output:



The screenshot shows the Arduino IDE Serial Monitor window titled "servo | Arduino IDE 2.0.0". The window has a menu bar with File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with Output and Serial Monitor buttons. The main area is labeled "Message (Ctrl + Enter to send message to 'Arduino Nano' on 'COM4')". At the bottom right are buttons for New Line and 9600 baud. The text area displays a series of servo motor degree values starting from 180 and decreasing in increments of 1 down to 170.

```

Servo motor Degree
180
Servo motor Degree
179
Servo motor Degree
178
Servo motor Degree
177
Servo motor Degree
176
Servo motor Degree
175
Servo motor Degree
174
Servo motor Degree
173
Servo motor Degree
172
Servo motor Degree
171
Servo motor Degree
170

```

Result:

Thus the above Implement the Servo Motor Control with Arduino program was executed and verified successfully

Ex No : 12

Date:

**IMPLEMENT AND MONITOR THE LM35 TEMPERATURE
SENSOR AND ULTRASONIC DISTANCE MEASUREMENT
WITH ARDUINO**

Aim:

To Implement and Monitor the LM35 Temperature Sensor and Ultrasonic Distance Measurement With Arduino

Procedure:

STEP 1: Open Arduino Software.

STEP 2: Click Tools -> Board -> Arduino Nano.

STEP 3: In Tools Menu Select Port -> COM.

STEP 4: Click File-> New and Start Write the Code.

STEP 5: And Click tick Icon Button of the file Menu to compile.

STEP 6: Click Upload Icon After Verify.

Program for LM35:

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(A5,A4,A3,A2,A1,A0);
int i;

void setup() {
  lcd.begin(16, 2);
  lcd.print("Find Temperature");
  lcd.setCursor(0,1);
  lcd.print("Experimenter");
  delay(3000);
  lcd.clear();
}

void temperature()
{
  int val;
  float mv;
  float cel;
  float farh;

  for(int i=0;i<20;i++)
  {
    lcd.clear();
    lcd.print("Temperature");
    val = analogRead(A6);
    mv = ( val/1024.0)*5000;
    cel = mv/10;
    farh = (cel*9)/5 + 32;
    lcd.setCursor(0,1);
    lcd.print(cel);
    lcd.print("*C");
    delay(1000);
  }
}
```

```
}
```

```
void loop() {
```

```
    temperature();
```

```
    delay(1000);
```

```
}
```

Program for Ultrasonic (SR04):

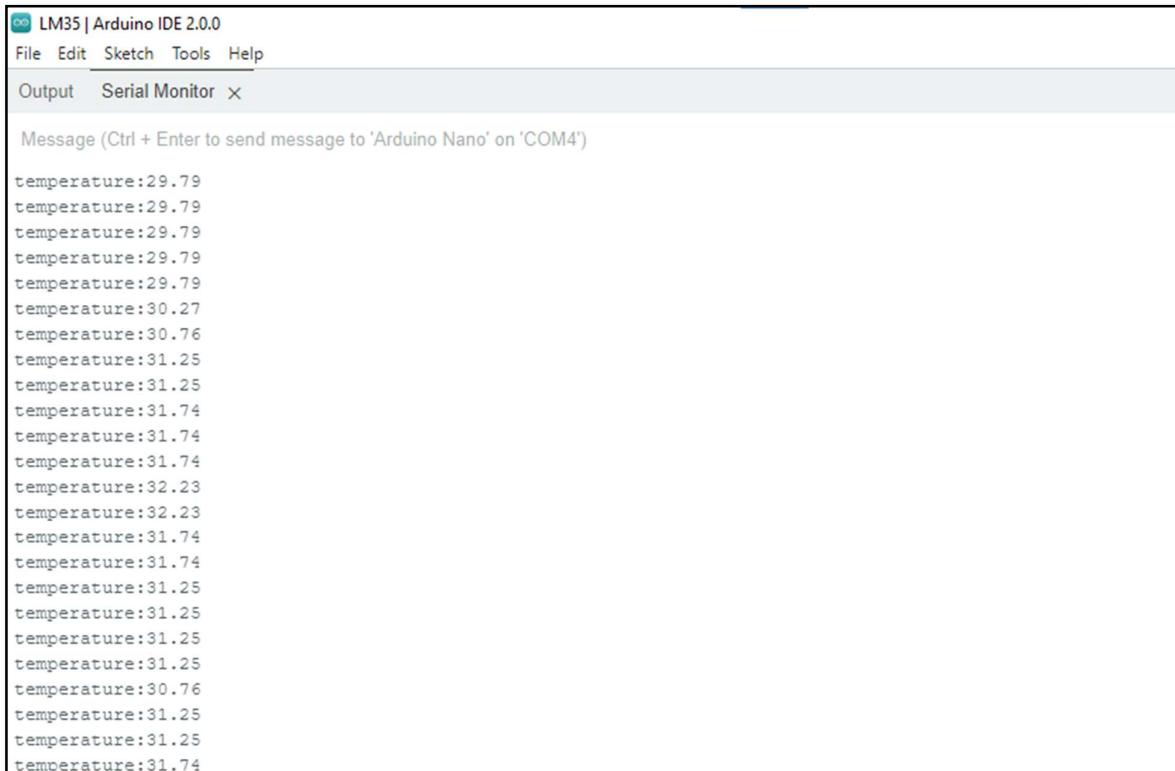
```
#include <LiquidCrystal.h>
LiquidCrystal lcd(A5,A4,A3,A2,A1,A0);
int i;

void setup() {
    lcd.begin(16, 2);
    lcd.print("Finding Distance");
    lcd.setCursor(0,1);
    lcd.print("Experimenter");
    delay(3000);
    lcd.clear();
}

void sr04()
{
    pinMode(13, OUTPUT);
    pinMode(12, INPUT);
    for(int i=0;i<100;i++)
    {
        lcd.clear();
        lcd.print("UltraSonic");
        lcd.setCursor(0,1);
        digitalWrite(13, LOW);
        delayMicroseconds(2);
        digitalWrite(13, HIGH);
        delayMicroseconds(10);
        digitalWrite(13, LOW);
        long duration = pulseIn(12, HIGH);
        int distance = duration * 0.034 / 2; // Speed of sound wave divided by 2 (go and back)
        lcd.print("Distance:");
        lcd.print(distance);
        lcd.print("cm");
        delay(100);
    }
}

void loop() {
    sr04();
    delay(1000);
}
```

Output:



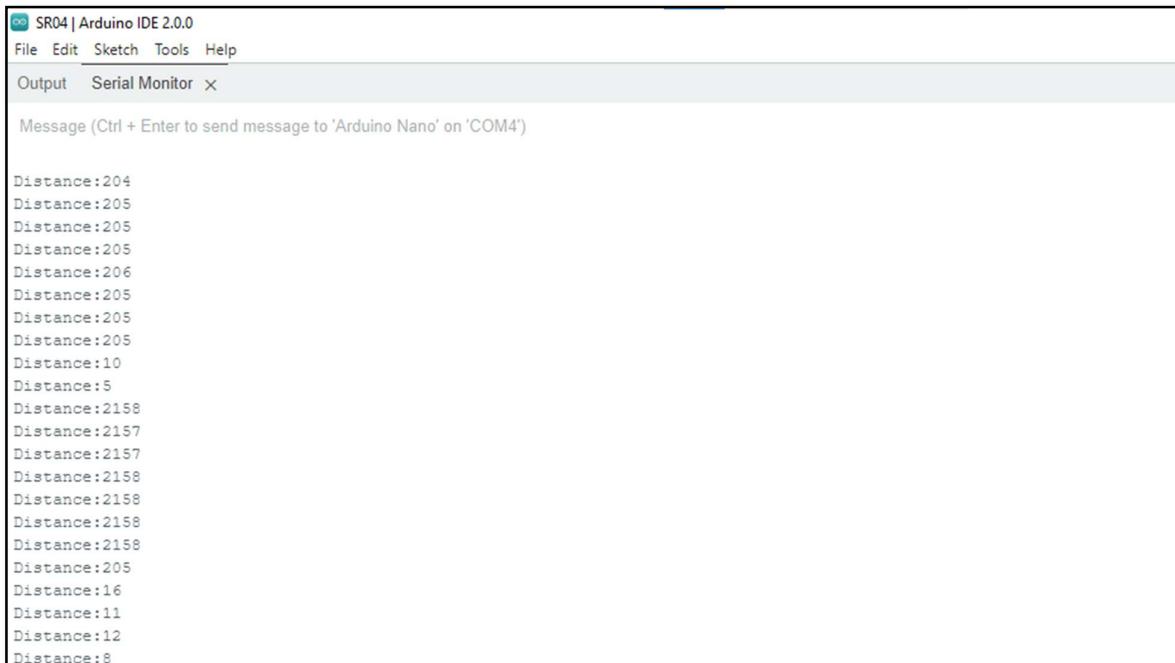
LM35 | Arduino IDE 2.0.0

File Edit Sketch Tools Help

Output Serial Monitor ×

Message (Ctrl + Enter to send message to 'Arduino Nano' on 'COM4')

```
temperature:29.79
temperature:29.79
temperature:29.79
temperature:29.79
temperature:29.79
temperature:30.27
temperature:30.76
temperature:31.25
temperature:31.25
temperature:31.74
temperature:31.74
temperature:31.74
temperature:32.23
temperature:32.23
temperature:31.74
temperature:31.74
temperature:31.25
temperature:31.25
temperature:31.25
temperature:30.76
temperature:31.25
temperature:31.25
temperature:31.74
```



SR04 | Arduino IDE 2.0.0

File Edit Sketch Tools Help

Output Serial Monitor ×

Message (Ctrl + Enter to send message to 'Arduino Nano' on 'COM4')

```
Distance:204
Distance:205
Distance:205
Distance:205
Distance:206
Distance:205
Distance:205
Distance:205
Distance:205
Distance:10
Distance:5
Distance:2158
Distance:2157
Distance:2157
Distance:2158
Distance:2158
Distance:2158
Distance:2158
Distance:205
Distance:16
Distance:11
Distance:12
Distance:8
```

Result:

Thus the above Implement and Monitor the LM35 Temperature Sensor and Ultrasonic Distance Measurement with Arduino Program was executed and verified successfully

Ex No : 13

Date:

IMPLEMENT THE IR SENSOR ANALOG INPUT WITH ARDUINO

Aim:

To Implement the IR Sensor Analog Input with Arduino

Procedure:

STEP 1: Open Arduino Software.

STEP 2: Click Tools -> Board -> Arduino Nano.

STEP 3: In Tools Menu Select Port -> COM.

STEP 4: Click File-> New and Start Write the Code.

STEP 5: And Click tick Icon Button of the file Menu to compile.

STEP 6: Click Upload Icon After Verify.

Program:

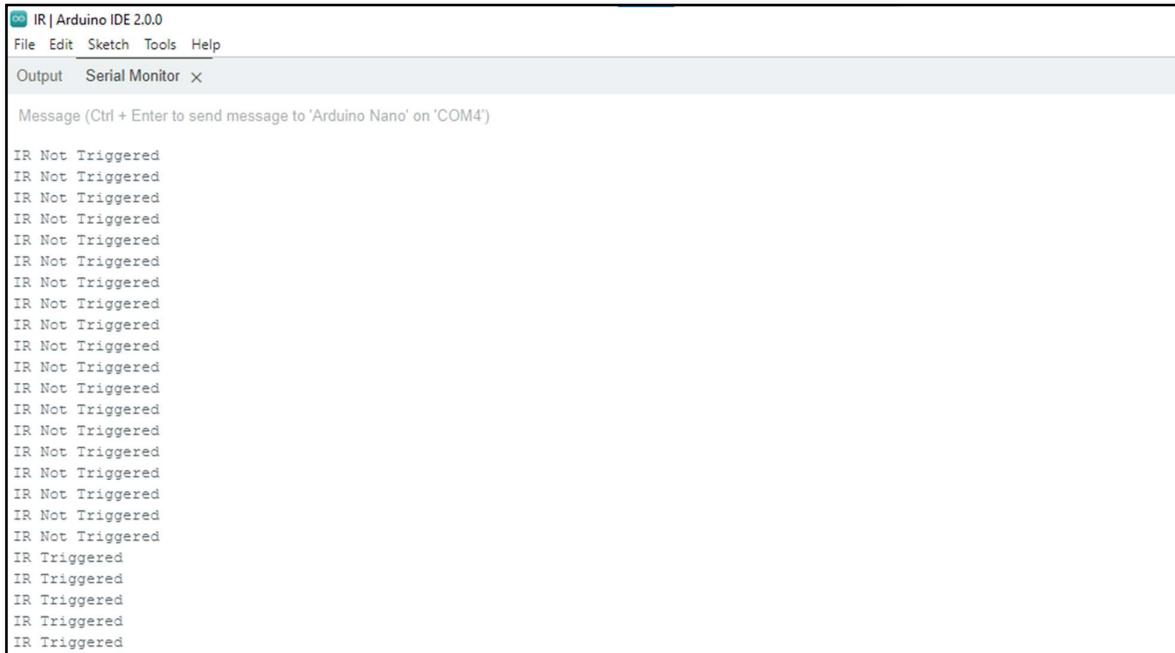
```
#include <LiquidCrystal.h>
LiquidCrystal lcd(A5,A4,A3,A2,A1,A0);
int i;

void setup() {
  lcd.begin(16, 2);
  lcd.print("IR INPUT");
  lcd.setCursor(0,1);
  lcd.print("Experimenter");
  delay(3000);
  lcd.clear();
}
void ir_sensor()
{
  pinMode(10,INPUT);
  pinMode(2,OUTPUT);
  digitalWrite(2,HIGH);

  for(int i=0;i<100;i++)
  {
    if(!digitalRead(10))
    {
      lcd.clear();
      lcd.print("IR Triggered");
      digitalWrite(2,LOW);
    }
    else
    {
      lcd.clear();
      lcd.print("IR Not Triggered");
      digitalWrite(2,HIGH);
    }
    delay(100);
  }
}
void loop()
```

```
    ir_sensor();
    delay(1000);
}
```

Output:



The screenshot shows the Arduino IDE 2.0.0 interface with the 'Serial Monitor' tab selected. The window title is 'IR | Arduino IDE 2.0.0'. The menu bar includes File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with Output and Serial Monitor buttons. A message at the top of the serial monitor window reads 'Message (Ctrl + Enter to send message to 'Arduino Nano' on 'COM4')'. The main text area displays a series of messages from the Arduino sketch, alternating between 'IR Not Triggered' and 'IR Triggered', indicating the state of the infrared sensor over time.

```
IR Not Triggered
IR Triggered
IR Triggered
IR Triggered
IR Triggered
```

Result:

Thus the above Implement the IR Sensor Analog Input with Arduino program was executed and verified successfully

Ex No : 14

Date:

USING THINKSPEAK CLOUD READING TEMPERATURE SENSOR MONITORING WITH NODEMCU

Aim:

To Using ThinkSpeak Cloud Reading Temperature Sensor Monitoring with NodeMCU

Procedure:

STEP 1: Open Arduino Software.

STEP 2: Click Tools -> Board -> NodeMcu.

STEP 3: In Tools Menu Select Port -> COM.

STEP 4: Click File-> New and Start Write the Code.

STEP 5: Login Your Think speak Cloud and check API key and Channel Number.

STEP 6: Open secrets.h Header file.

STEP 7: Edit SSID &Password of the WIFI and add Your API key and Channel Number of Think speak cloud

STEP 8: And Click tick Icon Button of the file Menu to compile.

STEP 9: Click Upload Icon After Verify.

STEP 10: Now, Check you think speak cloud temperature will sense as Graph

Program:

Secrets.h

```
#define SECRET_SSID "MySSID"          // replace MySSID with your WiFi network name
#define SECRET_PASS "MyPassword"        // replace MyPassword with your WiFi password
#define SECRET_CH_ID 0000000           // replace 0000000 with your channel number
#define SECRET_WRITE_APIKEY "XYZ"       // replace XYZ with your channel write API Key
```

NodeMcu.ino

```
#include <ESP8266WiFi.h>
#include "secrets.h"
#include "ThingSpeak.h" // always include thingspeak header file after other header files and
                      // custom macros
#include <SoftwareSerial.h>
SoftwareSerial swSer(D6, D7);
char ssid[] = SECRET_SSID; // your network SSID (name)
char pass[] = SECRET_PASS; // your network password
int keyIndex = 0;          // your network key Index number (needed only for WEP)
WiFiClient client;

unsigned long myChannelNumber = SECRET_CH_ID;
const char * myWriteAPIKey = SECRET_WRITE_APIKEY;

// Initialize our values
float number1 = 0;
float number2 = 0;
float number3 = 0;
float number4 = 0;
```

```
float number5 = 0;
float number6 = 0;
float number7 = 0;
float number8 = 0;
String myStatus = "";
String response;
int ESPwait(String stopstr, int timeout_secs)
{
    bool found = false;
    char c;
    long timer_init;
    long timer;
    response="";
    timer_init = millis();
    while (!found) {
        timer = millis();
        if (((timer - timer_init) / 1000) > timeout_secs) { // Timeout?
            Serial.println("!Timeout!");
            return 0; // timeout
        }
        if (swSer.available()) {
            c = swSer.read();
            //Serial.print(c);
            response += c;
            if (response.endsWith(stopstr)) {
                found = true;
                delay(10);
                swSer.flush();
                Serial.flush();
                Serial.println();
            }
        } // end Serial1_available()
    } // end while (!found)
    return 1;
}
int ESPwait1(String stopstr, int timeout_secs)
{
    bool found = false;
    char c;
    long timer_init;
    long timer;
    response="";
    timer_init = millis();
    while (!found) {
        timer = millis();
        if (((timer - timer_init) / 1000) > timeout_secs) { // Timeout?
            Serial.println("!Timeout!");
            return 0; // timeout
        }
        if (Serial.available()) {
            c = Serial.read();
            Serial.print(c);
            response += c;
        }
    }
}
```

```

if(response.endsWith(stopstr)) {
    found = true;
    delay(10);
    Serial.flush();
    Serial.println();
}
} // end Serial1.available()
} // end while (!found)
return 1;
}
void setup() {
    Serial.begin(9600); // Initialize serial
    swSer.begin(9600);
    pinMode(LED_BUILTIN,OUTPUT);

    while (!Serial) {
        ; // wait for serial port to connect. Needed for Leonardo native USB port only
    }
    WiFi.mode(WIFI_STA);
    ThingSpeak.begin(client); // Initialize ThingSpeak
}
void loop() {
    char c;
    // Connect or reconnect to WiFi
    if(WiFi.status() != WL_CONNECTED){
        Serial.print("Attempting to connect to SSID: ");
        Serial.println(SECRET_SSID);
        while(WiFi.status() != WL_CONNECTED){
            WiFi.begin(ssid, pass); // Connect to WPA/WPA2 network. Change this line if using open or
            WEP network
            Serial.print(".");
            delay(5000);
        }
        Serial.println("\nConnected.");
    }

    for(int i=0;i<5;i++)
    {
        digitalWrite(LED_BUILTIN,LOW);
        delay(100);
        digitalWrite(LED_BUILTIN,HIGH);
        delay(100);
    }
}

if(swSer.available())
{
    c=swSer.read();
    swSer.println(c);
    if(c=='*')
    {
        if(ESPwait("#",3))
        {
            char * strtokIdx;
            response.remove(response.length()-1);
        }
    }
}

```

```

strtokIdx = strtok(const_cast<char*>(response.c_str()),","); // get the first part - the string
number1 = atof(strtokIdx);
Serial.println(response);
response="";

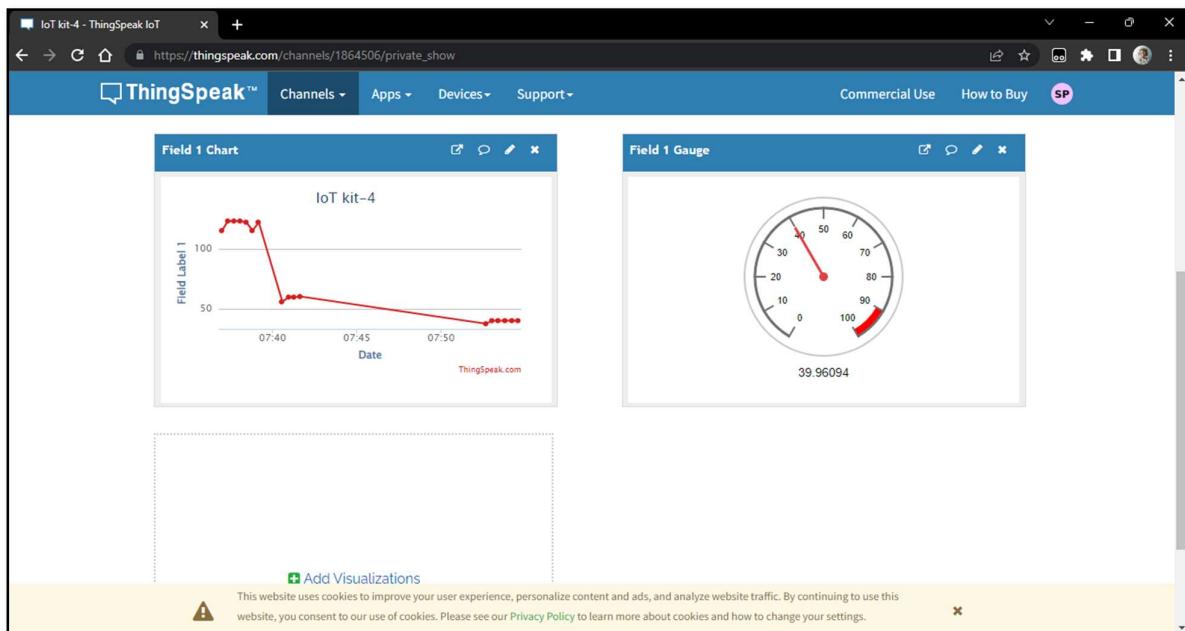
ThingSpeak.setField(1, number1);
int x = ThingSpeak.writeField(myChannelNumber, 1, number1, myWriteAPIKey);
if(x == 200)
{
    Serial.println("Channel update successful.");
    digitalWrite(LED_BUILTIN,LOW);
}

else{
    Serial.println("Problem updating channel. HTTP error code " + String(x));
    digitalWrite(LED_BUILTIN,HIGH);
}
Serial.println("Sent");

}
}
}
}

```

Output:



Result:

Thus the above ThinkSpeak Cloud Reading Temperature Sensor Monitoring with NodeMCU was executed successfully