

UNIT NO	List of Practical	Date	Signature
<u>1</u>	Write a program for implementing Client Server communication model using TCP		
<u>2</u>	Write a program for implementing the Client Server communication model using UDP.		
<u>3</u>	Write a program to show the object communication using RMI.		
<u>4</u>	Show the implementation of web services.		
<u>5</u>	Study and implementation of Infrastructure as a Service. and study of cloud computing Architecture		
<u>6</u>	Installation and Configuration of virtualization using KVM.		
<u>7</u>	Study and implementation of Infrastructure as a Service		
<u>8</u>	Study and implementation of Storage as a Service		
<u>9</u>	Study and implementation of Identity Management		
<u>10</u>	Study Cloud Security management		
<u>11</u>	Write a program for the web feed.		
<u>12</u>	Study and implementation of Single-Sing-On.		
<u>13</u>	User Management in the Cloud		
<u>14</u>	Case study on Amazon EC2/Microsoft Azure/Google Cloud Platform		

BSCS602

Pratik Patil

Roll No: 58

KERALEEYA SAMAJAM(REGD.) DOMBIVLI'S
MODEL COLLEGE
EMPOWERED AUTONOMOUS

PRACTICAL 1

Date: 05-12-24

Aim: Write a program for implementing a client server communication model using TCP.

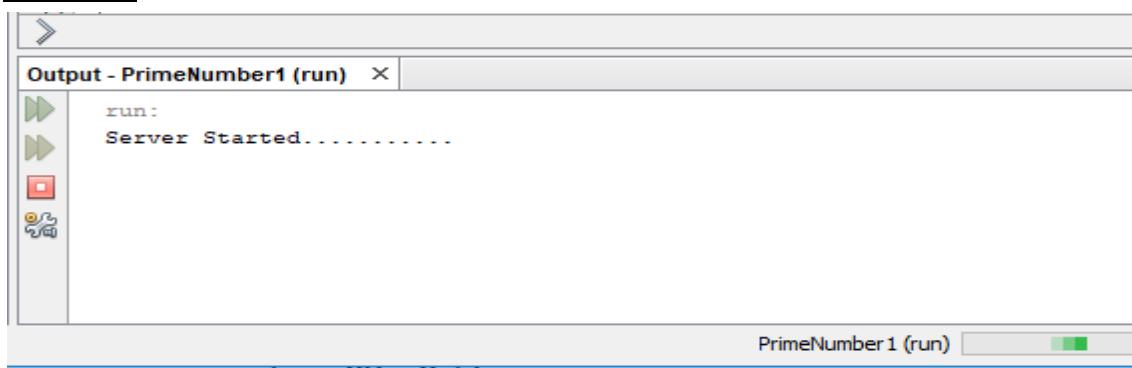
a) A client-server based program using TCP to find if the number entered is prime or not. Server Code:

```

package primenumber;
import java.net.*;
import java.io.*;

public class PrimeNumber{
    public static void main(String[] args){
        try{
            ServerSocket ss = new ServerSocket(8001);
            System.out.println("Server Started.....");
            Socket s = ss.accept();
            DataInputStream in = new DataInputStream(s.getInputStream()); int x
            = in.readInt();
            DataOutputStream otc = new DataOutputStream(s.getOutputStream()); boolean
            isPrime = true;
            if (x <= 1) {
                isPrime = false;
            } else {
                for (int i = 2; i <= x / 2; i++) {
                    if (x % i == 0) {
                        isPrime = false;
                        break;
                    }
                }
            }
            if (isPrime) {
                otc.writeUTF(x + " is prime");
            } else {
                otc.writeUTF(x + " is not prime");
            }
            s.close();
            ss.close();
        } catch (Exception e) {
            System.out.println(e.toString());
        }
    }
}

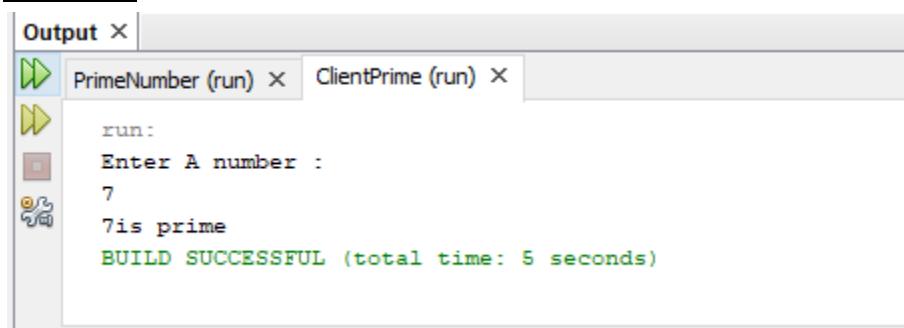
```

OUTPUT:


The screenshot shows the Eclipse IDE's Output window titled "Output - PrimeNumber1 (run)". It contains the message "run: Server Started.....". The window has standard Eclipse UI elements like a toolbar on the left and a progress bar at the bottom.

Client-code:

```
package clientprime;
import java.io.*;
import java.net.*;
public class ClientPrime {
    public static void main(String[] args) { try{
        Socket cs = new Socket("LocalHost",8001);
        BufferedReader infu = new BufferedReader(new InputStreamReader(System.in));
        System.out.println("Enter A number :");
        int a = Integer.parseInt(infu.readLine());
        DataOutputStream out = new DataOutputStream(cs.getOutputStream());
        out.writeInt(a);
        DataInputStream in = new DataInputStream(cs.getInputStream());
        System.out.print(in.readUTF());
        cs.close();
    }
    catch(Exception e){
        System.out.println(e.toString());
    }
}
}
```

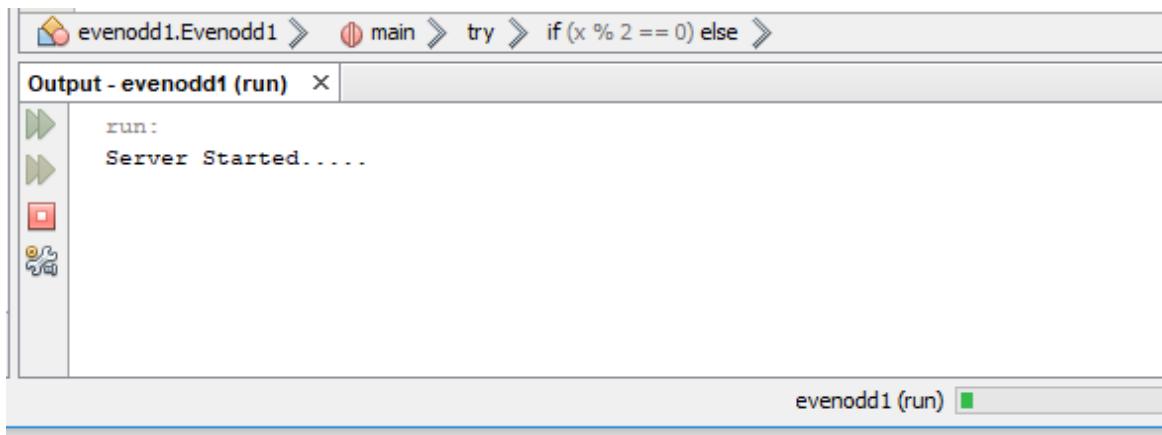
OUTPUT:


The screenshot shows the Eclipse IDE's Output window with two tabs: "Output" and "ClientPrime (run)". The "ClientPrime (run)" tab is active and displays the following interaction:
 run:
 Enter A number :
 7
 7is prime
 BUILD SUCCESSFUL (total time: 5 seconds)

b) A client-server based program using TCP to find if the number entered is Even or Odd. Server code:

```
package evenodd1;
import java.net.*;
import java.io.*;
public class Evenodd1 {
    public static void main(String[] args) throws IOException {
        try{
            ServerSocket ss = new ServerSocket(8002);
            System.out.println("Server Started .......");
            Socket s = ss.accept();
            DataInputStream in = new DataInputStream(s.getInputStream()); int x
            = in.readInt();
            DataOutputStream otc = new DataOutputStream(s.getOutputStream()); if(x %
            2==0)
            {
                otc.writeUTF(x + "is Even");
            }
            else{
                otc.writeUTF(x + "is odd");
            }
        }
        catch(Exception e){
            System.out.println(e.toString());
        }
    }
}
```

OUTPUT:



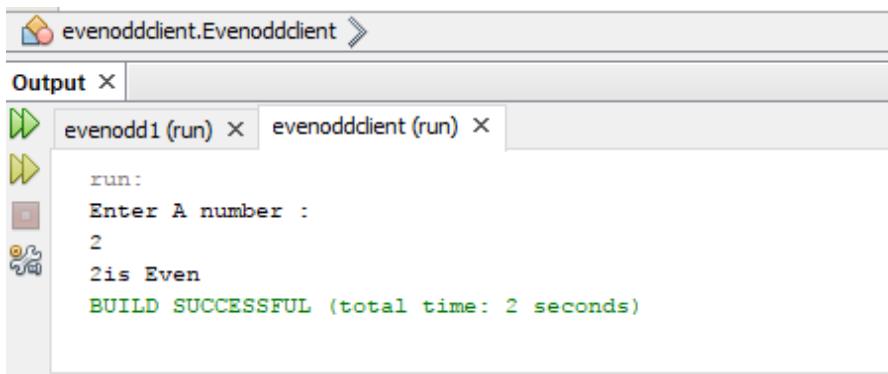
The screenshot shows the Java IDE interface with the following details:

- Project Path:** evenodd1.Evenodd1 > main > try > if (x % 2 == 0) else >
- Output Window:** Output - evenodd1 (run) X
- Output Content:**

```
run:
Server Started.....
```
- Toolbar:** Includes icons for Run, Stop, Refresh, and others.
- Status Bar:** evenodd1 (run) [green icon]

Client-code:

```
package evenoddclient;
import java.net.*; import
java.io.*;
public class Evenoddclient {
    public static void main(String[] args) { try{
        Socket cs = new Socket("LocalHost",8002);
        BufferedReader infu = new BufferedReader(new InputStreamReader(System.in));
        System.out.println("Enter A number :");
        int a = Integer.parseInt(infu.readLine());
        DataOutputStream out = new DataOutputStream(cs.getOutputStream());
        out.writeInt(a);
        DataInputStream in = new DataInputStream(cs.getInputStream());
        System.out.println(in.readUTF());
        cs.close();
    }
    catch(Exception e){
        System.out.println(e.toString());
    }
}
}
```

OUTPUT:

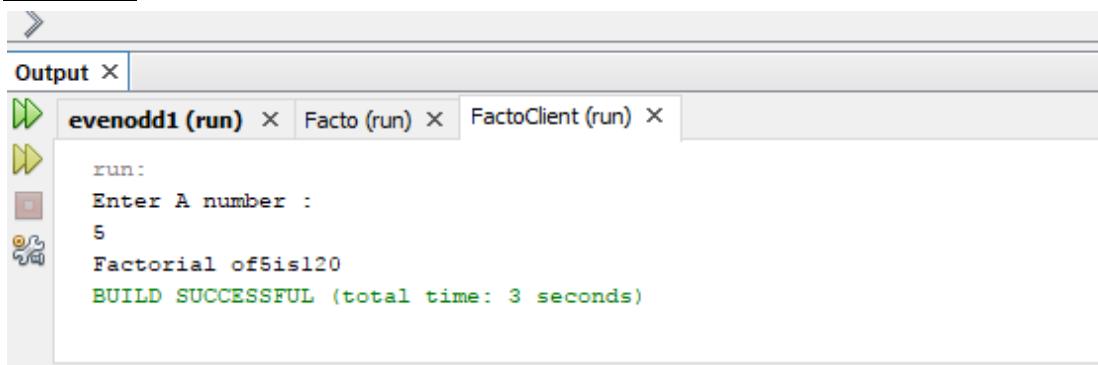
```
evenoddclient.Evenoddclient >
Output ×
evenodd1 (run) × evenoddclient (run) ×
run:
Enter A number :
2
2 is Even
BUILD SUCCESSFUL (total time: 2 seconds)
```

c) A client-server based program using TCP to find the factorial of the number entered. Server code:

```
package facto; import  
java.io.*; import  
java.net.*;  
public class  
Facto {  
    public static void main(String[] args) { try{  
        ServerSocket ss = new ServerSocket(8003);  
        System.out.println("Server Started.....");  
        Socket s = ss.accept();  
        DataInputStream in = new DataInputStream(s.getInputStream()); int x  
        = in.readInt();  
        DataOutputStream otc = new DataOutputStream(s.getOutputStream()); long  
        factorial=1;  
        for(int i =1;i<=x;i++){  
            factorial *= i;  
        }  
        otc.writeUTF("Factorial of"+ x+"is"+factorial);  
    }  
    catch(Exception e){  
        System.out.println(e.toString());  
    }  
}
```

```
Client-code package  
factoclient; import  
java.io.*; import  
java.net.*;  
public class FactoClient {  
    public static void main(String[] args) { try{  
        Socket cs = new Socket("LocalHost",8003);  
        BufferedReader infu = new BufferedReader(new InputStreamReader(System.in));  
        System.out.println("Enter A number :");  
        int a = Integer.parseInt(infu.readLine());  
        DataOutputStream out = new DataOutputStream(cs.getOutputStream());  
        out.writeInt(a);  
        DataInputStream in = new DataInputStream(cs.getInputStream());  
        System.out.println(in.readUTF());  
        cs.close();  
    }  
    catch(Exception e){
```

```
        System.out.println(e.toString());
    }
}
}
```

OUTPUT:

```
evenodd1 (run) X Facto (run) X FactoClient (run) X
run:
Enter A number :
5
Factorial of 5 is 120
BUILD SUCCESSFUL (total time: 3 seconds)
```

PRACTICAL 2

Date: 12-12-24

Aim: Write a program for implementing the Client-Server communication model using UDP.

- a) Client-Server based program UDP to find the number entered is even or odd.

Server Code:

```

package evenoddserver;
import java.io.*;
import java.net.*;
public class EvenOddServer {
    public static void main(String[] args) { try
    {
        DatagramSocket ds = new DatagramSocket(2000); byte
        b[] = new byte[1024];
        DatagramPacket dp = new DatagramPacket(b,b.length); ds.receive(dp);
        String str = new String(dp.getData(),0,dp.getLength());
        System.out.println(str);
        int a = Integer.parseInt(str);
        String s = new String();
        if(a%2==0)
            s= "Number is even"; else
            s= "Number is odd"; byte
        b1[] = new byte[1024]; b1 =
        s.getBytes();
        DatagramPacket dp1 = new
        DatagramPacket(b1,b1.length,InetAddress.getLocalHost(),1000); ds.send(dp1);
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
}

```

Client Code:

```

package evenoddclient1;
import java.io.*;
import java.net.*;
public class EvenOddClient1 {
    public static void main(String[] args) { try
    {
        DatagramSocket ds = new DatagramSocket(1000);
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        System.out.println("Enter a number :");
        String num = br.readLine();
        byte b[] = new byte[1024];
        b= num.getBytes();
        DatagramPacket dp = new
        DatagramPacket(b,b.length,InetAddress.getLocalHost(),2000); ds.send(dp);
        byte b1[] = new byte[1024];
        DatagramPacket dp1 =new DatagramPacket(b1,b1.length);
        ds.receive(dp1);
        String str = new String(dp1.getData(),0,dp1.getLength());
        System.out.println(str);
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
}
}

```

OUTPUT:

```

Output x
updServerEO (run) x udpClientEO (run) x
run:
Enter a number:
7
Number is odd
BUILD SUCCESSFUL (total time: 2 seconds)
|
```

```

Output ×
updServerEO (run) × udpClientEO (run) ×

run:
7
BUILD SUCCESSFUL (total time: 5 seconds)
|
```

- b)** Client-server based program using UDP to find factorial of entered number.

Server Code package

```

serverfacto; import
java.io.*; import
java.net.*;
public class ServerFacto {
    public static void main(String[] args) { try{
        DatagramSocket ds = new DatagramSocket(2000); byte b
        [] = new byte[1024];
        DatagramPacket dp = new DatagramPacket(b,b.length); ds.receive(dp);
        String str = new String(dp.getData(),0,dp.getLength());
        System.out.println(str);
        int num = Integer.parseInt(str);
        long factorial = calculateFactorial(num);
        String result = "Factorial of" + num + "is" + factorial; byte
        b1[] = result.getBytes();
        DatagramPacket dp1 = new DatagramPacket(b1,b1.length,InetAddress.getLocalHost(),1000);
        ds.send(dp1);
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
public static long calculateFactorial(int num)
{
    long factorial =1; for(int
    i=1;i<=num;i++){
        factorial*=i;
    }
    return factorial;
```

```

    }
}
}
```

Client Code package

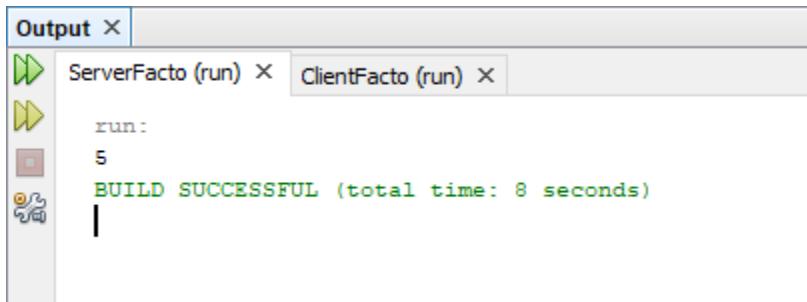
```

clientfacto; import
java.io.*; import
java.net.*;
public class ClientFacto {
    public static void main(String[] args) { try
    {
        DatagramSocket ds=new DatagramSocket(1000);
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        System.out.println("Enter a number: ");
        String num=br.readLine(); byte[]
        b=num.getBytes();
        DatagramPacket dp=new
        DatagramPacket(b,b.length,InetAddress.getLocalHost(),2000); ds.send(dp);
        byte[] b1=new byte[1024];
        DatagramPacket dp1=new DatagramPacket(b1,b1.length);
        ds.receive(dp1);
        String result=new String(dp1.getData(),0,dp1.getLength());
        System.out.println(result);
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
}
```

OUTPUT:

```

Output X
ServerFacto (run) X ClientFacto (run) X
run:
Enter a number:
5
Factorial of5is120
BUILD SUCCESSFUL (total time: 2 seconds)
|
```



The screenshot shows the 'Output' tab of an IDE. It displays the results of a build process. The log entries are:

- ServerFacto (run) × ClientFacto (run) ×
- run:
- 5
- BUILD SUCCESSFUL (total time: 8 seconds)

- c) Client-server based program to implement simple calculator operations addition, subtraction, multiplication and division.

Server Code:

```
package calserver;
import java.io.*;
import java.net.*;
public class CalServer {
    public static void main(String[] args) { try
    {
        DatagramSocket ds = new DatagramSocket(2000); byte[]
        b = new byte[1024];
        DatagramPacket dp = new DatagramPacket(b, b.length); ds.receive(dp);
        String str = new String(dp.getData(), 0, dp.getLength());
        System.out.println("Received data: " + str);

        // Parse the received data (e.g., "5 + 3") String[]
        parts = str.split(" ");
        double num1 = Double.parseDouble(parts[0]); String
        operation = parts[1];
        double num2 = Double.parseDouble(parts[2]);

        // Perform the operation
        double result = 0; switch
        (operation) {
            case "+":
                result = num1 + num2;
                break;
            case "-":
                result = num1 - num2;
                break;
            case "*":
                result = num1 * num2;
    
```

```

        break;

    case "/":
        if (num2 != 0) {
            result = num1 / num2;
        } else {
            throw new ArithmeticException("Division by zero");
        }
        break;
    default:
        throw new IllegalArgumentException("Invalid operation: " + operation);
    }

    // Send the result back to the client String
    response = "Result: " + result; byte[] b1 =
    response.getBytes();
    DatagramPacket dp1 = new DatagramPacket(b1, b1.length, dp.getAddress(), dp.getPort());
    ds.send(dp1);
} catch (Exception e) {
    e.printStackTrace();
}
}
}
}

```

Client Code:

```

package calclient;
import java.io.*;
import java.net.*;
public class CalClient {
    public static void main(String[] args) { try
    {
        DatagramSocket ds = new DatagramSocket(1000);
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

        // Input two numbers and an operation from the user
        System.out.println("Enter the first number: "); String
        num1 = br.readLine(); System.out.println("Enter the
        operation (+, -, *, /): "); String operation =
        br.readLine(); System.out.println("Enter the second
        number: "); String num2 = br.readLine();

        // Combine input into a single string
        String data = num1 + " " + operation + " " + num2; byte[]
    }
}
}
}
}

```

```
b = data.getBytes();

// Send the data to the server
DatagramPacket dp = new DatagramPacket(b, b.length,
InetAddress.getLocalHost(), 2000);
ds.send(dp);

// Receive the result from the server
byte[] b1 = new byte[1024];
DatagramPacket dp1 = new DatagramPacket(b1, b1.length);
ds.receive(dp1);
String response = new String(dp1.getData(), 0, dp1.getLength());
System.out.println(response);

} catch (Exception e) {
    e.printStackTrace();
}

}
```

OUTPUT:

Addition:

```
Output x | CalServer (run) x CalClient (run) x |  
run:  
Enter the first number:  
5  
Enter the operation (+, -, *, /):  
+  
Enter the second number:  
2  
Result: 7.0  
BUILD SUCCESSFUL (total time: 18 seconds)
```

```
Output X CalServer (run) X CalClient (run) X

run:
Received data: 5 + 2
BUILD SUCCESSFUL (total time: 21 seconds)
```

Subtraction:

```
Output ×
CalServer (run) × CalClient (run) ×

run:
Enter the first number:
4
Enter the operation (+, -, *, /):
-
Enter the second number:
3
Result: 1.0
BUILD SUCCESSFUL (total time: 6 seconds)
```

```
Output ×
CalServer (run) × CalClient (run) ×

run:
Enter the first number:
5
Enter the operation (+, -, *, /):
+
Enter the second number:
2
Result: 7.0
```

Multiplication:

```
Output ×
CalServer (run) × CalClient (run) ×

run:
Enter the first number:
2
Enter the operation (+, -, *, /):
*
Enter the second number:
3
Result: 6.0
BUILD SUCCESSFUL (total time: 7 seconds)
```

```
Output ×
CalServer (run) × CalClient (run) ×

run:
Received data: 2 * 3
BUILD SUCCESSFUL (total time: 11 seconds)
```

Division:

```
Output ×
CalServer (run) × CalClient (run) ×

run:
Enter the first number:
4
Enter the operation (+, -, *, /):
/
Enter the second number:
2
Result: 2.0
BUILD SUCCESSFUL (total time: 10 seconds)
```

```
Output ×
CalServer (run) × CalClient (run) ×

run:
Received data: 4 / 2
BUILD SUCCESSFUL (total time: 13 seconds)
```

- d) Client-server based program to find square, square root, cube and cube root of the entered numbers.

Server code:

```
package udpserveroperations;
import java.io.*;
import java.net.*;
import java.math.*;
public class UdpServerOperations { public
static void main (String args []){ try {
System.out.print("Server is running..."); DatagramSocket ds
= new DatagramSocket(2000);
```

```

byte b[]=new byte[1024];
DatagramPacket dp = new DatagramPacket(b,b.length); ds.receive(dp);
String str=new String (dp.getData(),0,dp.getLength());
System.out.println(str);
int a=Integer.parseInt(str);
String s= new String();
s="The Square: "+ a*a+" \nSquare _Root: "+Math.sqrt(a)
+" \nCube:"+a*a*a+"\nCube_Root: "+Math.cbrt(a);
byte b1[]= new byte[1024];
b1=s.getBytes(); DatagramPacket
dp1= new
DatagramPacket(b1,b1.length,InetAddress.getLocalHost(),1000); ds.send(dp1);
}
catch (Exception e){
e.printStackTrace();
}
}
}
}

```

Client code:

```

package udpclientoperations;
import java.io.*;
import java.net.*;
public class UdpClientOperations { public
    static void main (String args[]){
        try {
            System.out.println("Client is running...");
            BufferedReader br = new BufferedReader (new InputStreamReader(System.in));
            System.out.println("Enter Number :");
            String num= br.readLine(); byte
            b[]= new byte[1024];
            b=num.getBytes();
            DatagramSocket ds= new DatagramSocket (1000); DatagramPacket dp
            = new
            DatagramPacket(b,b.length,InetAddress.getLocalHost(),2000); ds.send(dp);
            byte b1[]= new byte[1024];
            DatagramPacket dp1= new DatagramPacket(b1,b1.length);
            ds.receive(dp1);
            String r= new String (dp1.getData(),0,dp1.getLength());
            System.out.println(r);
        }
    }
}

```

```
        catch (Exception e){ e.printStackTrace();
    }
}
}
```

OUTPUT:

```
Client is running...
Enter Number :
8
The Square: 64
Square _Root: 2.8284271247461903
Cube:512
Cube_Root: 2.0
BUILD SUCCESSFUL (total time: 2 seconds)
```

BSCS602

Pratik Patil

Roll No: 58

KERALEEYA SAMAJAM(REGD.) DOMBIVLI'S
MODEL COLLEGE
EMPOWERED AUTONOMOUS

PRACTICAL 3

Date: 02-01-25

Aim: Write a program to show the object communication using RMI.

- a) RMI-based application program to display current date and time.

```

import java.rmi.*;
public interface InterDate extends Remote {
    public String display() throws Exception;
}
//Server code
import java.rmi.*;
import java.rmi.server.*;
import java.util.*;
public class ServerDate extends UnicastRemoteObject implements InterDate{ public
ServerDate() throws Exception
{
}
public String display() throws Exception
{
String str="";
Date d=new Date();
str=d.toString(); return
str;
}
public static void main(String args[]) throws Exception {
ServerDate s1=new ServerDate();
Naming.bind("DS", s1);
System.out.println("Object registered.....");
}
}

```

Client Code

```

import java.rmi.*;
import java.io.*;
public class ClientDate {
public static void main(String args[]) throws Exception { String
s1;
InterDate h1=(InterDate)Naming.lookup("DS");

```

```
s1=h1.display();
System.out.println(s1);
}
}

OUTPUT:
```

The image contains two vertically stacked screenshots of the Eclipse IDE's Output view. Both screenshots show three tabs at the top: 'RMI (run) X', 'RMI (run) #2 X', and 'RMI (run) #3 X'. The left screenshot shows the following log entries:

```
run:
Object registered.....
```

The right screenshot shows the following log entries:

```
run:
Sat Jan 04 11:13:15 IST 2025
BUILD SUCCESSFUL (total time: 0 seconds)
```

- b)** RMI-based application program that converts digits to words. Eg 123 will be converted to ONE TWO THREE.

```
//interface.java
import java.rmi.*;
public interface InterConvert extends Remote{
public String convertDigit(String no) throws Exception;
}

//server code
import java.rmi.*;
import java.rmi.server.*;
public class ServerConvert extends UnicastRemoteObject implements InterConvert{ public
ServerConvert() throws Exception
{
}
public String convertDigit(String no) throws Exception
{
String str="";
for(int i=0;i<no.length();i++)
{
int p=no.charAt(i);
if(p==48)
```

```
{  
str+="zero";  
}  
if(p==49)  
{  
str+="one";  
}  
if(p==50)  
{  
str+="two";  
}  
if(p==51)  
{  
str+="three";  
}  
if(p==52)  
{  
str+="four";  
}  
if(p==53)  
{  
str+="five";  
}  
if(p==54)  
{  
str+="six";  
}  
if(p==55)  
{  
str+="seven";  
}  
if(p==56)  
{  
str+="eight";  
}  
if(p==57)  
{  
str+="nine";  
}  
}  
return str;  
}  
public static void main(String args[]) throws Exception {
```

```

ServerConvert s1=new ServerConvert(); Naming.bind("Wrd",
s1);
System.out.println("Object registered ..... ");
}
}

Client code import
java.rmi.*; import
java.io.*;
public class ClientConvert {
public static void main(String args[]) throws Exception
{
InterConvert h1=(InterConvert)Naming.lookup("Wrd");
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
System.out.println("Enter a number : \t");
String no=br.readLine();
String ans=h1.convertDigit(no);
System.out.println("The word representation of the entered digit is : "+ans);
}
}

```

OUTPUT:

The image displays three separate screenshots of the Eclipse IDE's Output view, each showing a series of log entries. The tabs at the top of each view are labeled 'Output', 'RMI (run) #2', 'RMI1 (run)', and 'RMI1 (run) #2'. The first screenshot shows the initial registration of the object. The second and third screenshots show the interaction between the client and the server, including the user input and the resulting word representation of the digit.

```

Output X
RMI (run) #2 × RMI1 (run) × RMI1 (run) #2 × RMI1 (run) #3 ×
run:
Object registered.....|
```



```

Output X
RMI (run) #2 × RMI1 (run) × RMI1 (run) #2 × RMI1 (run) #3 ×
run:
Enter a number :
5
The word representation of the entered digit is : five
BUILD SUCCESSFUL (total time: 6 seconds)|
```



```

Output X
RMI (run) #2 × RMI1 (run) × RMI1 (run) #2 × RMI1 (run) #3 ×
run:
Enter a number :
246
The word representation of the entered digit is : twofoursix
BUILD SUCCESSFUL (total time: 13 seconds)|
```

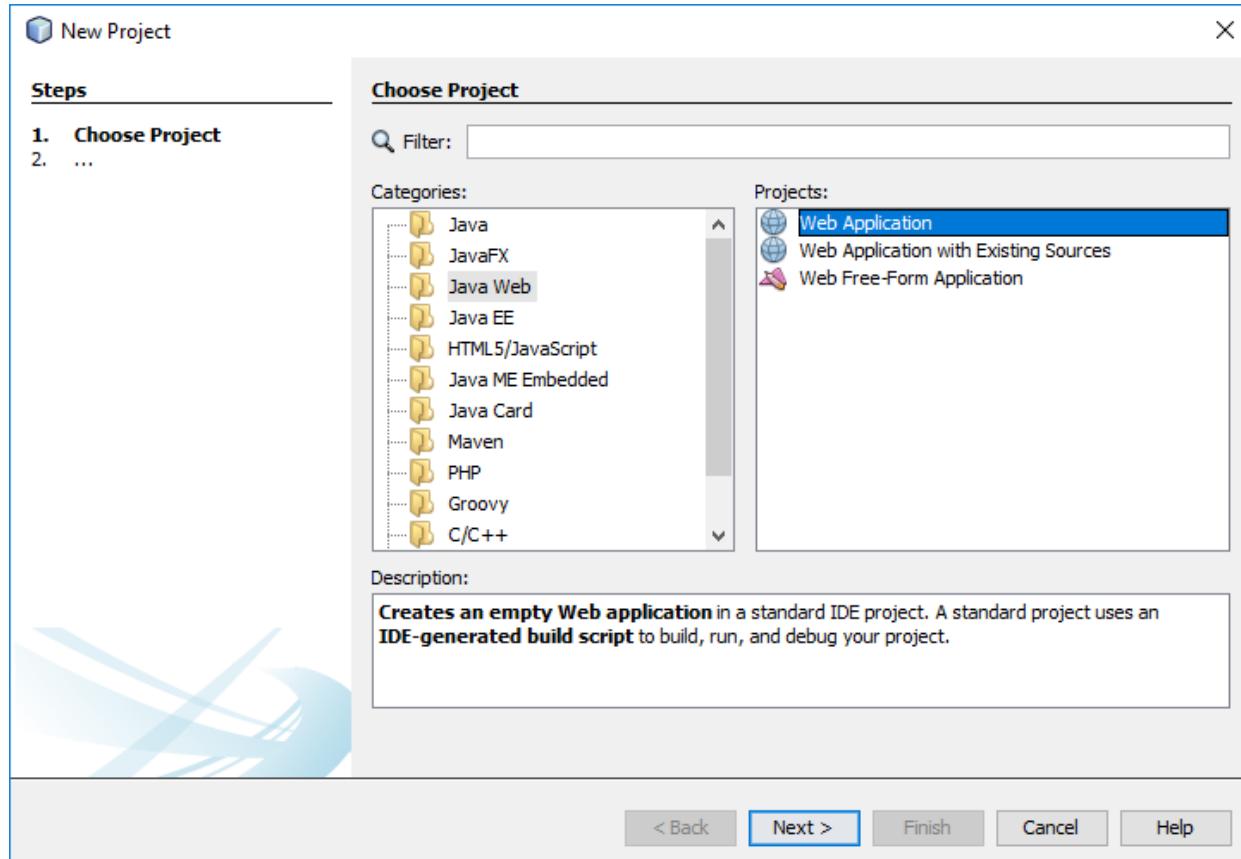
PRACTICAL 4

Date: 16-01-25

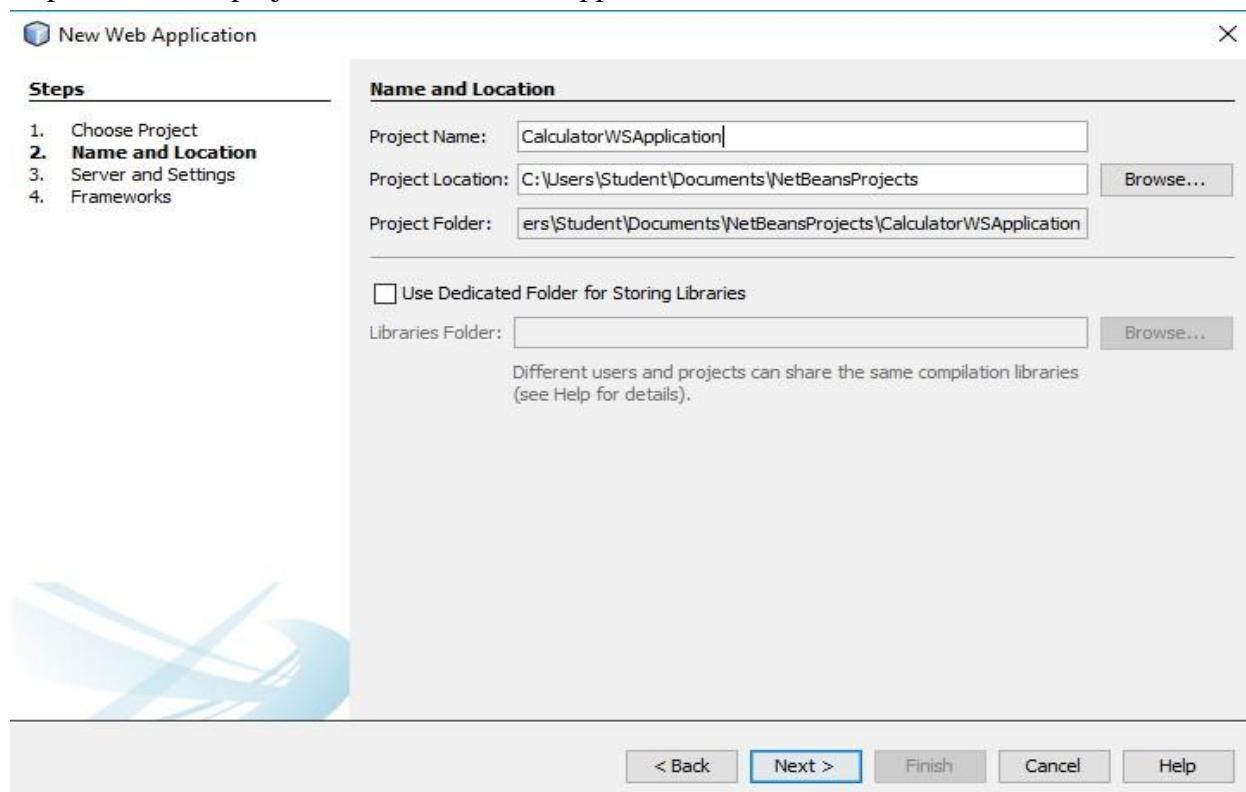
Aim: Show the implementation of webservices.

- a) Implement Big web services.

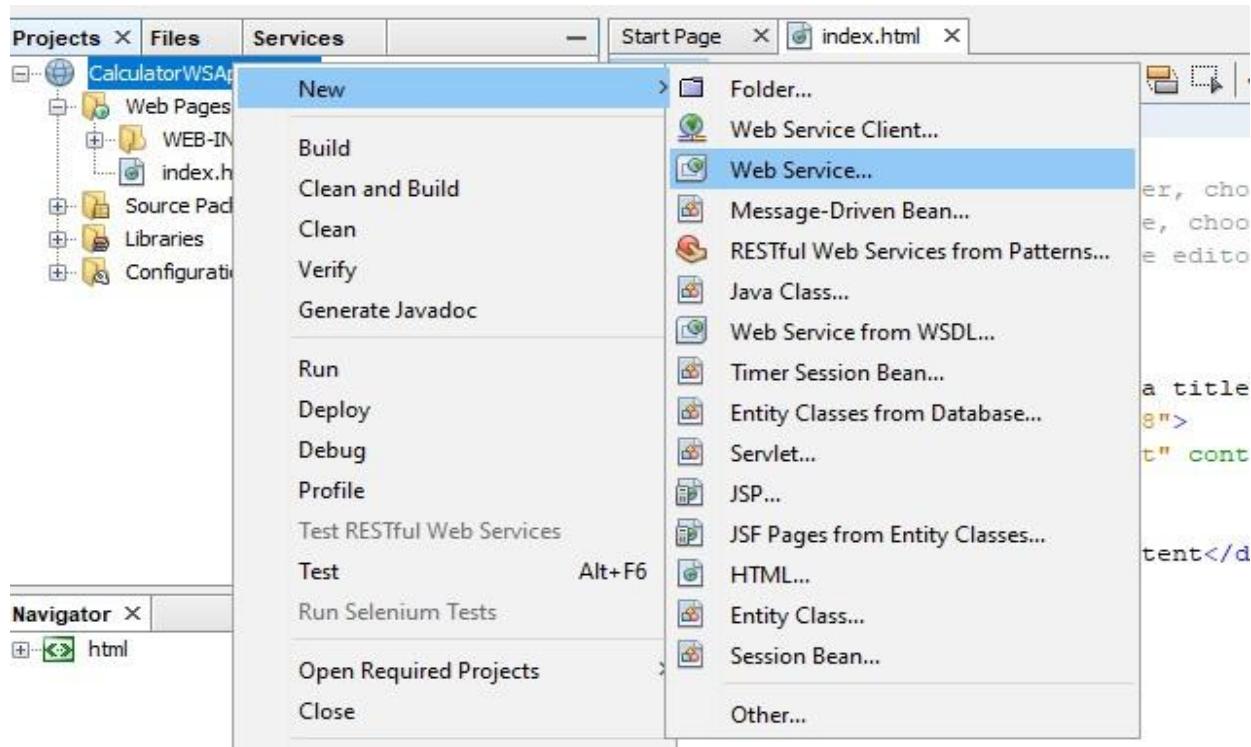
Step 1: Select File → New Project → Java Web → Web Application.



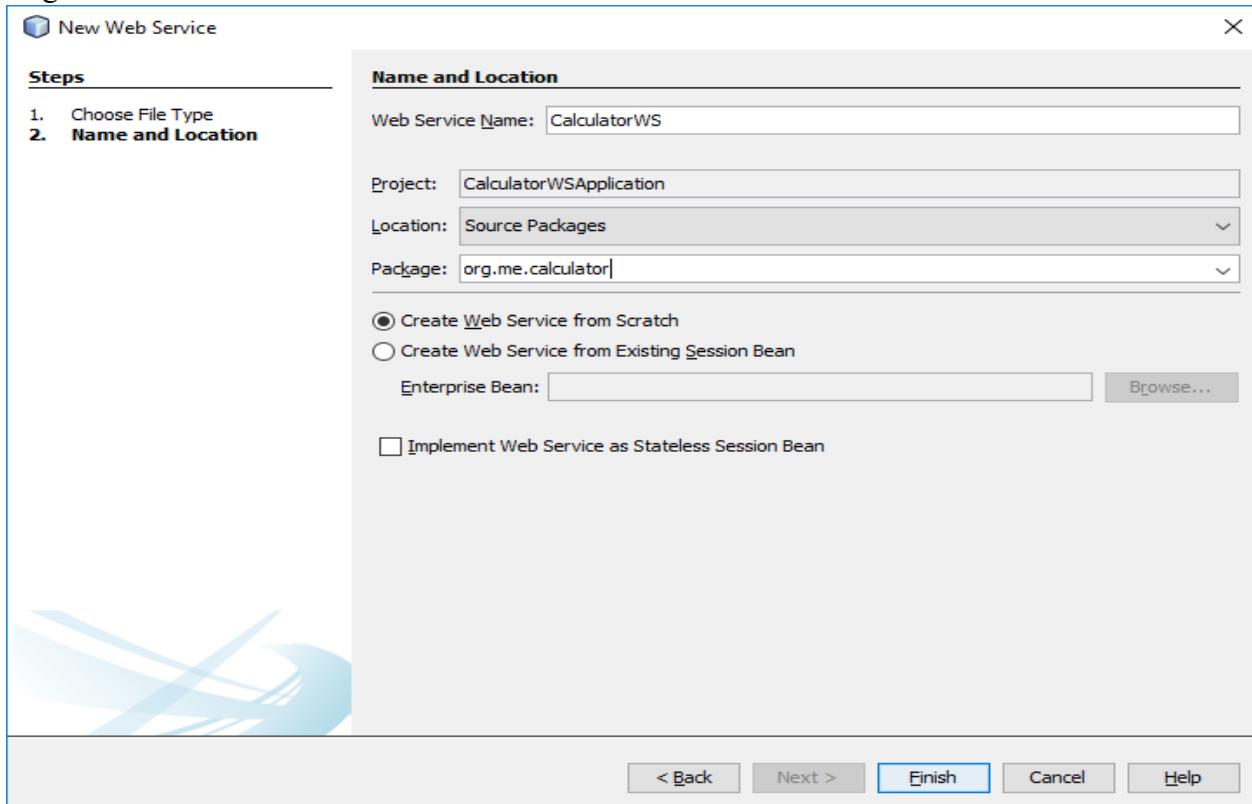
Step 2: Name the project as “CalculatorWSApplication”. Click on Next.



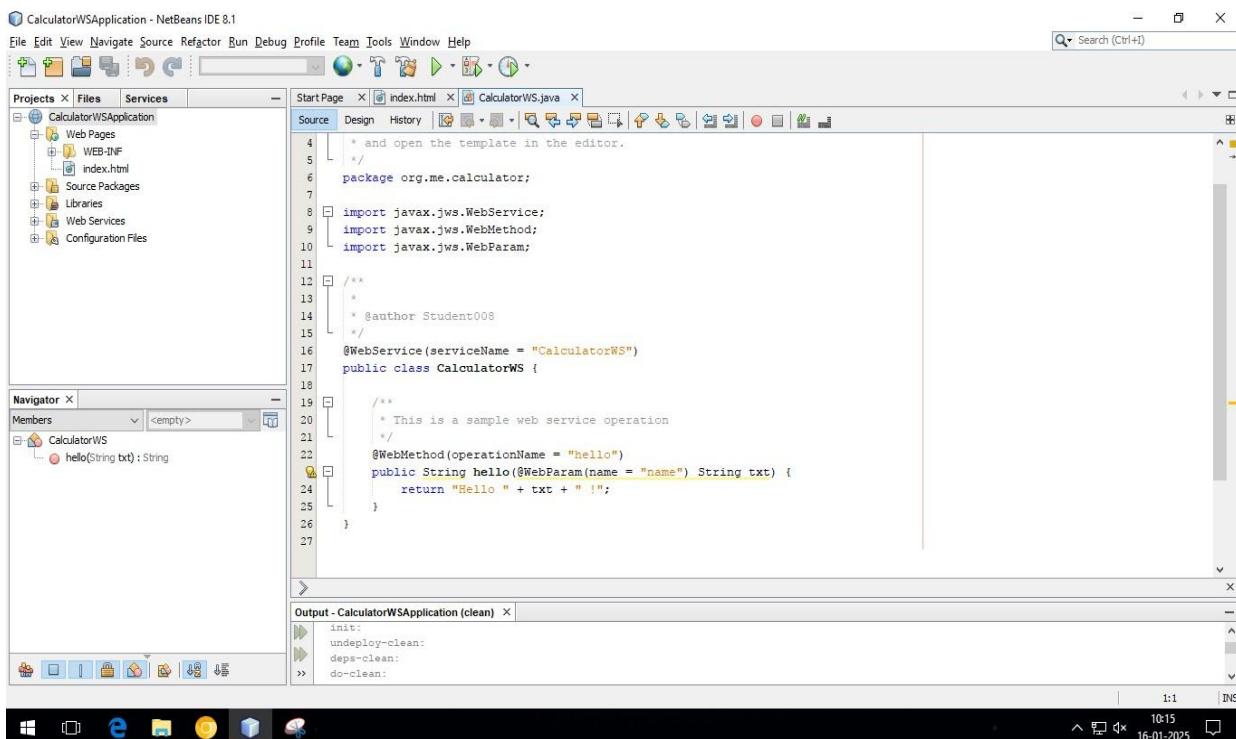
Step 3: A window like this will appear right click on the project that we created New → Web Service.



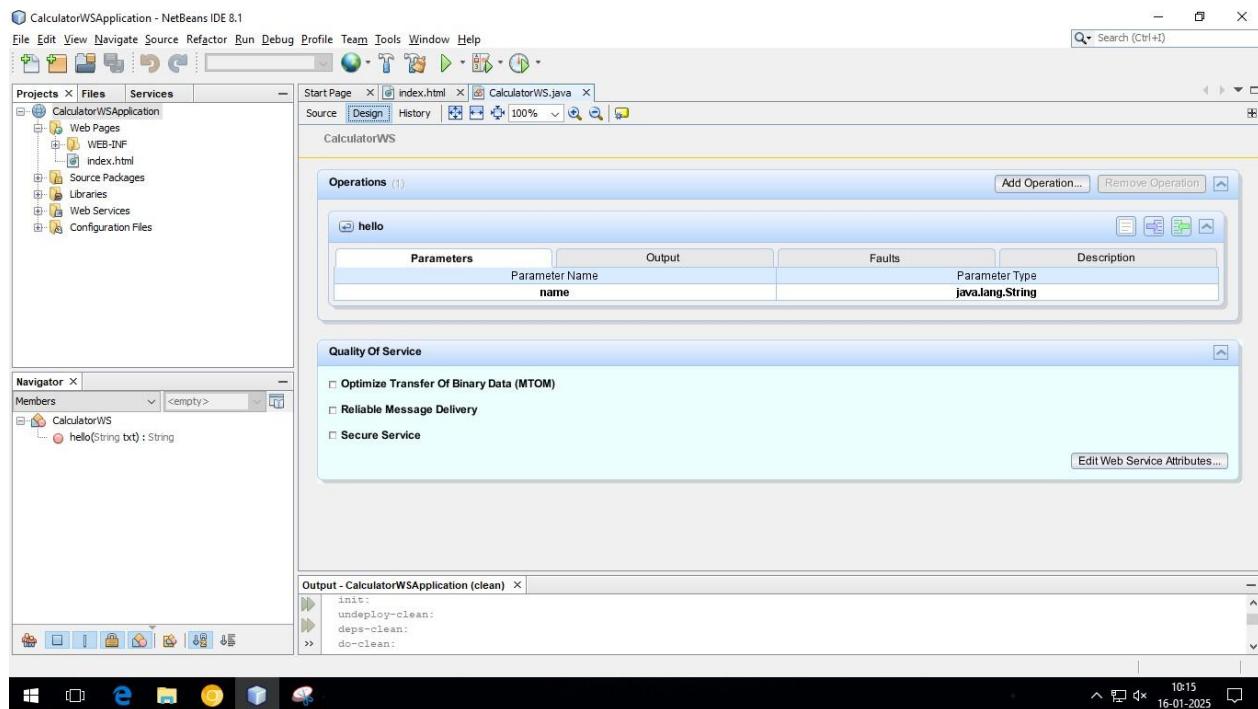
Step 4: Give the name as “CalculatorWS” and also provide a package name as “org.me.calculator”. Click on Finish.



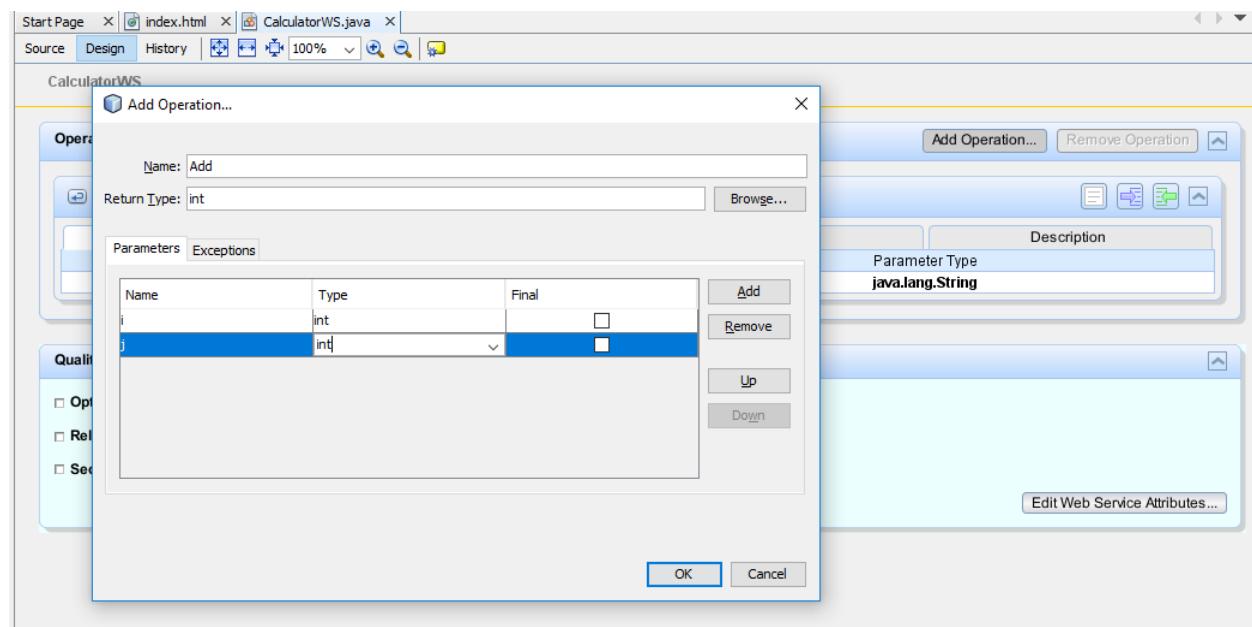
Step 5: Now a window like this will appear along with the source code as below.



Step 6: Go to the Design tab as shown below.



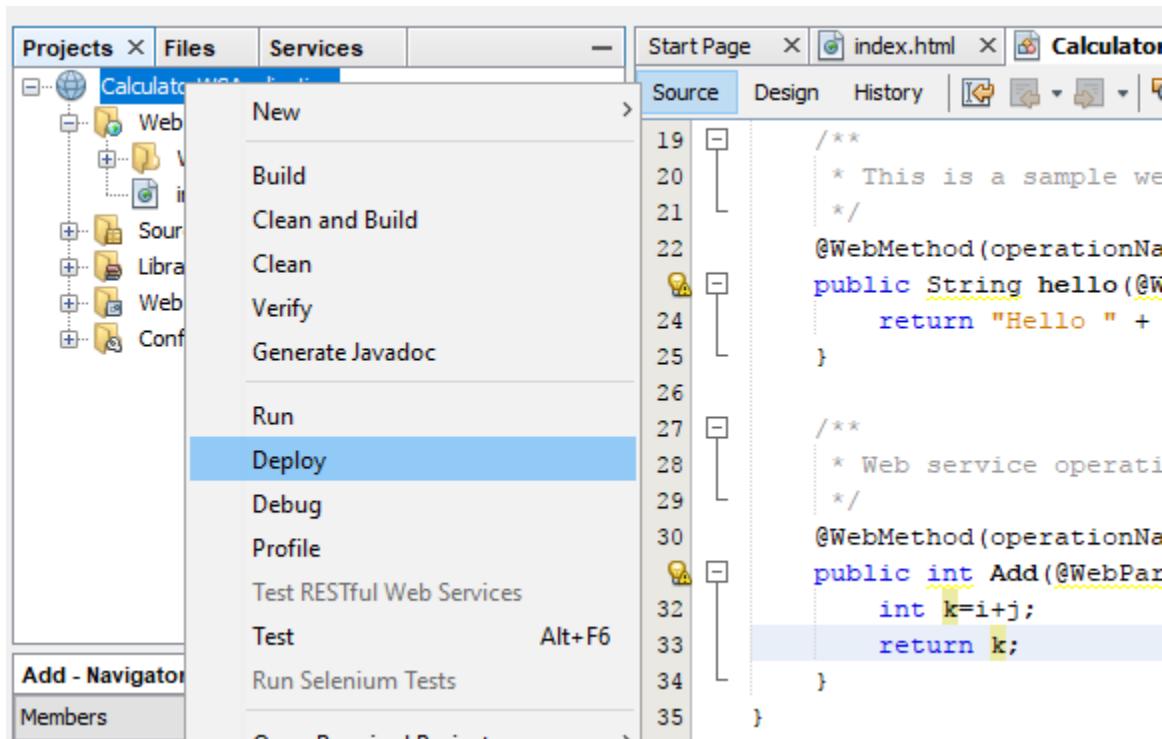
Step 7: In the first dialog box give the Name as “Add” and Return Type as “int”. Then click on the add button to add “i” and “j” parameters and provide the type as “int”. Click on Ok.



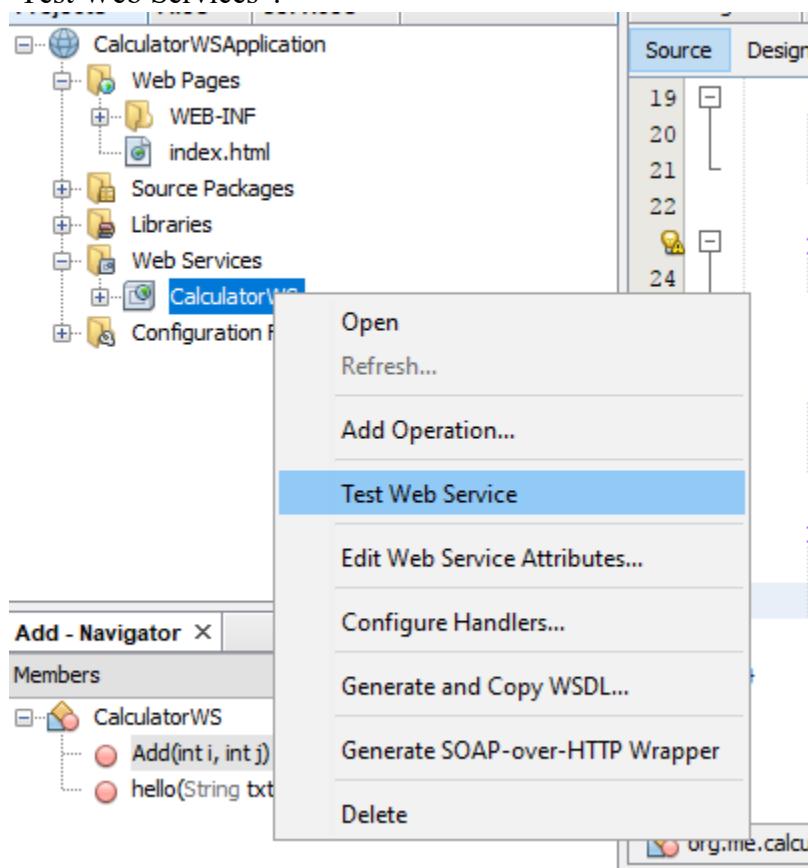
Step 8: The following code will be visible in the source code .

```
* Web service operation
*/
@WebMethod(operationName = "Add")
public int Add(@WebParam(name = "i") int i, @WebParam(name = "j") int j) {
    int k=i+j;
    return k;
}
```

Step 9: Now Right Click on the project and select “Deploy”.



Step 10: Now go to the project tab→web services→CalculatorWS. Right click on it and select “Test Web Services”.



Step 11: A window like this will appear provide the numbers as “3” and “4” click on the add button.

CalculatorWS Web Service Tester

This form will allow you to test your web service implementation ([WSDL File](#))

To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.

Methods :

public abstract int org.me.calculator.CalculatorWS.add(int,int)

(,)

public abstract java.lang.String org.me.calculator.CalculatorWS.hello(java.lang.String)

()

Step 12: Here we can see the method returned as “int 7”.

add Method invocation

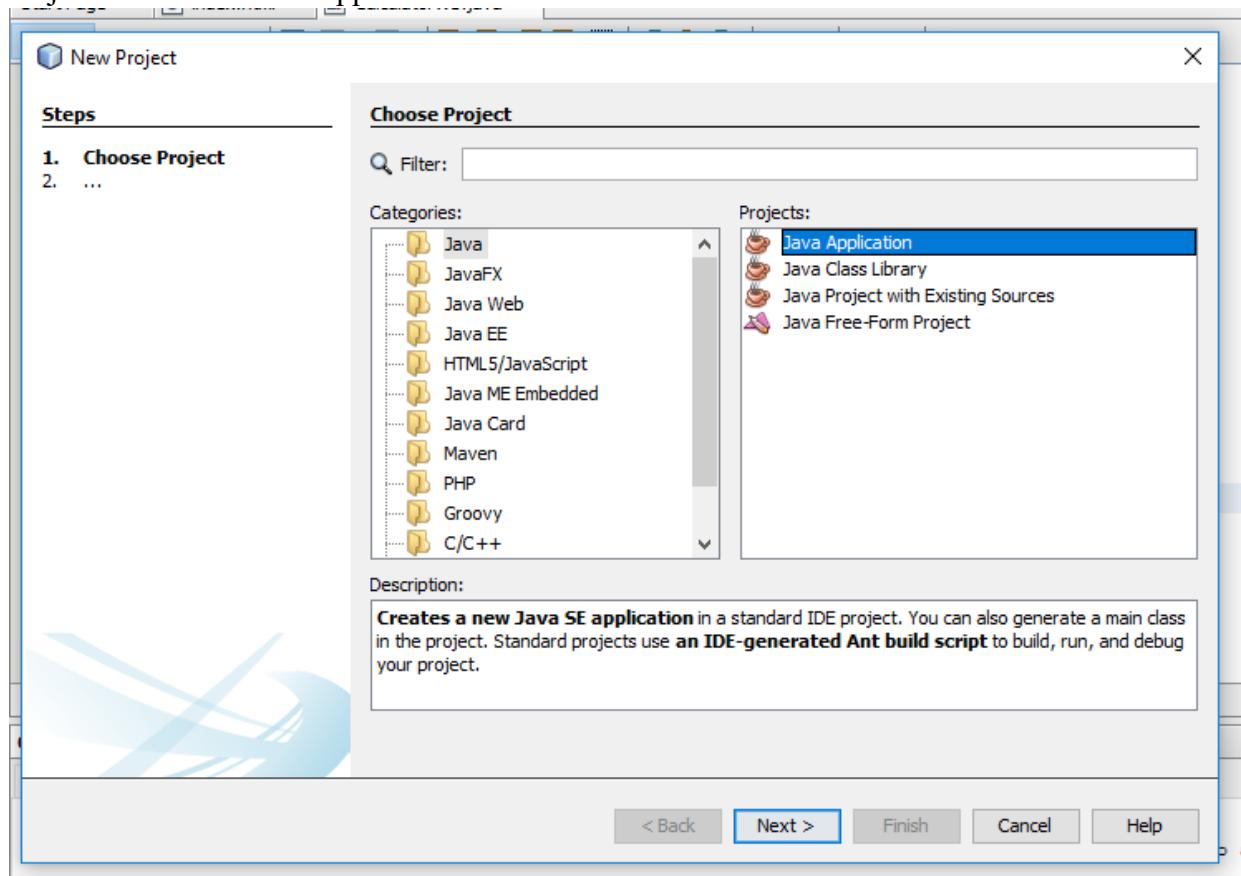
Method parameter(s)

Type	Value
int	3
int	4

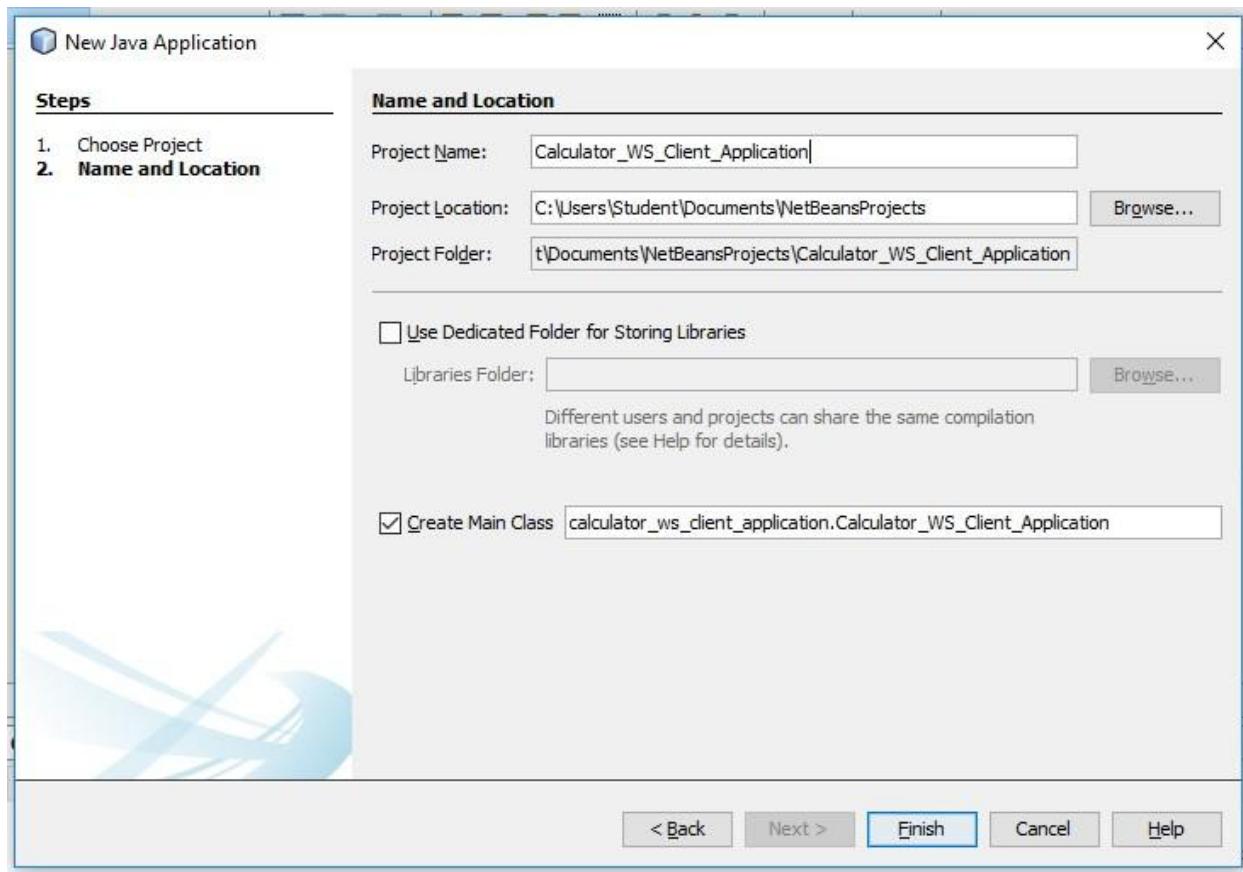
Method returned

int : "7"

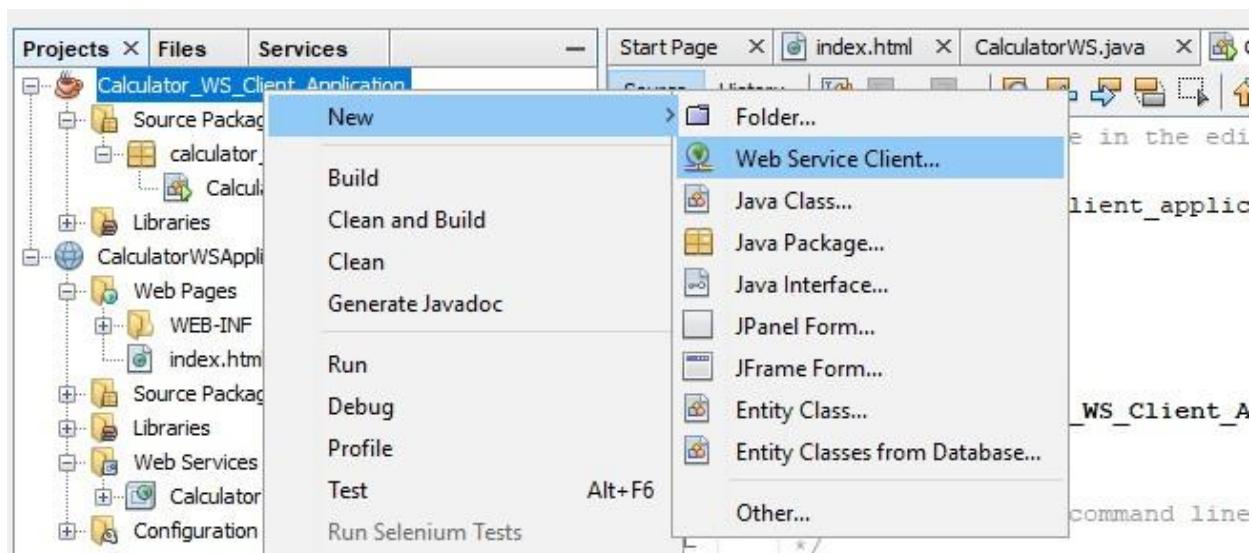
Step 13: Now we have to create a client for consuming the web services . Click on File → New Project → Java → Java Application.



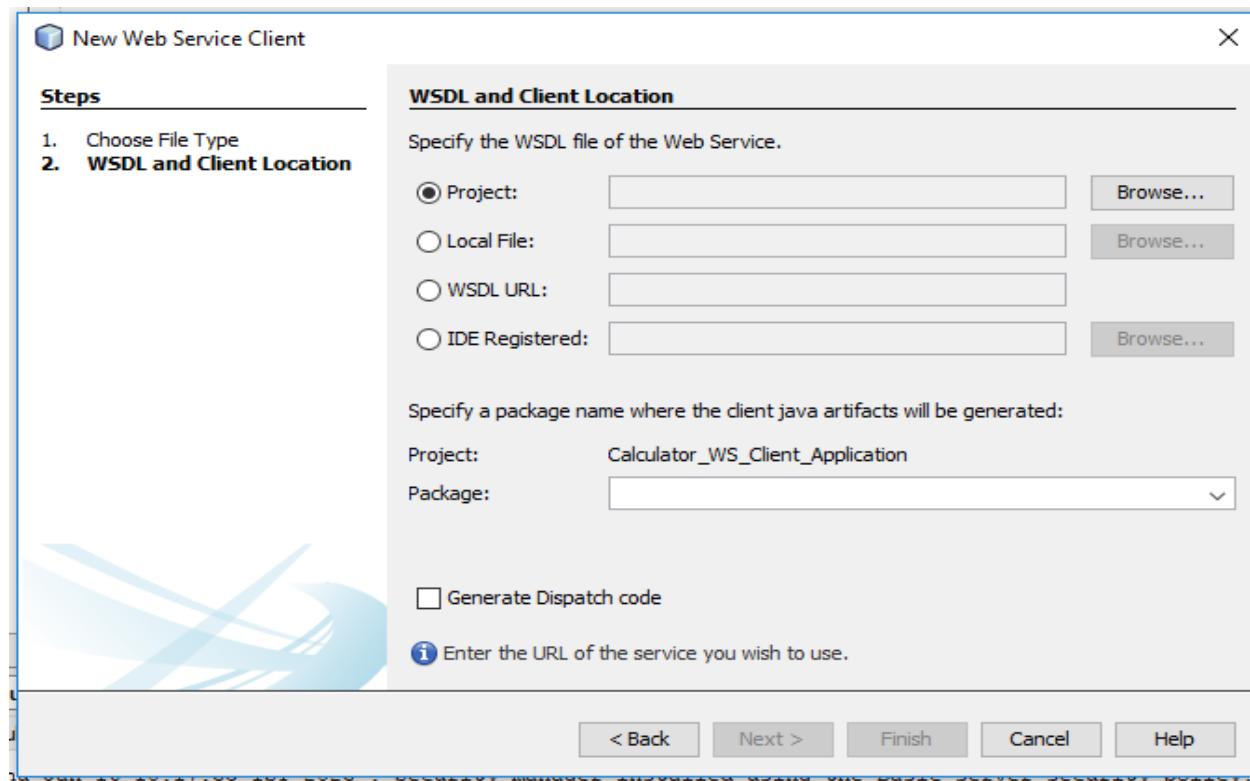
Step 14: Give the Name as “Calculator_WS_Client_Application”. Click on Finish.



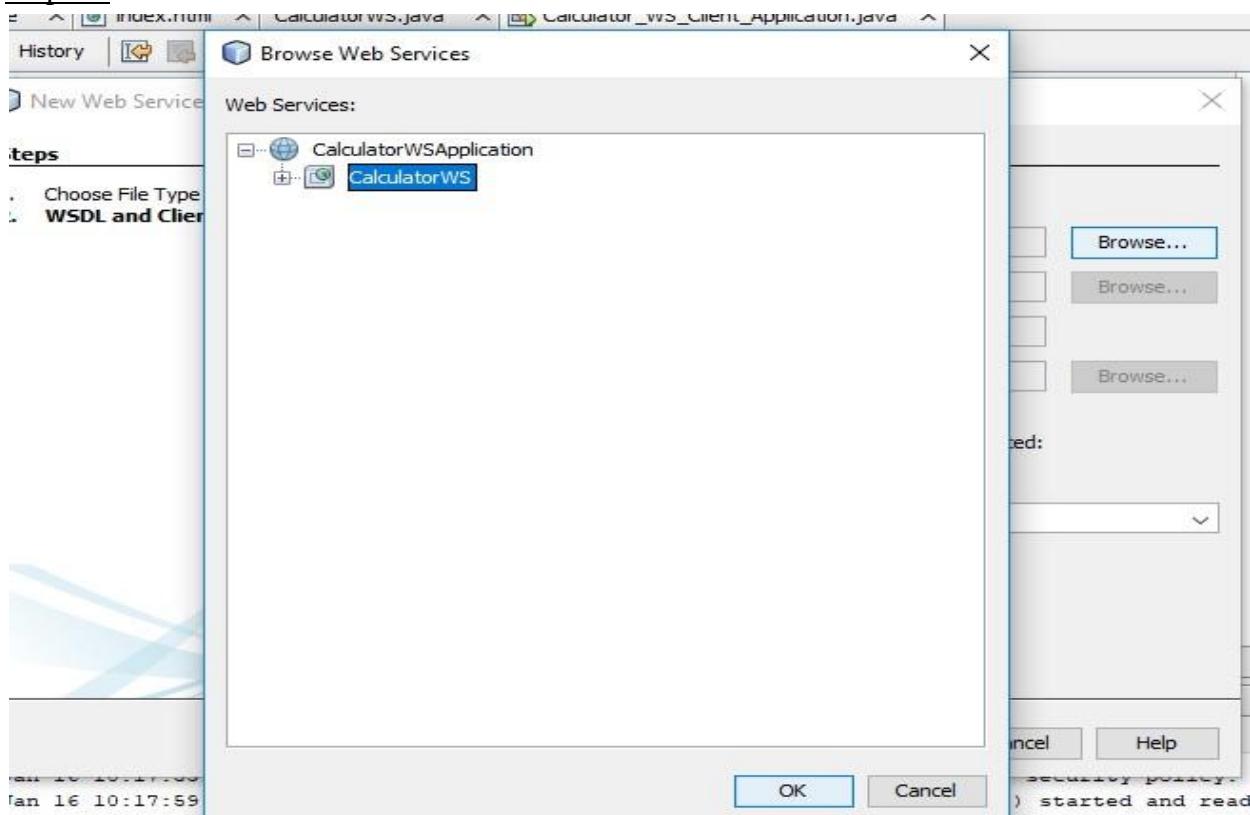
Step 15: Right click on the project New → Web Service → Client.



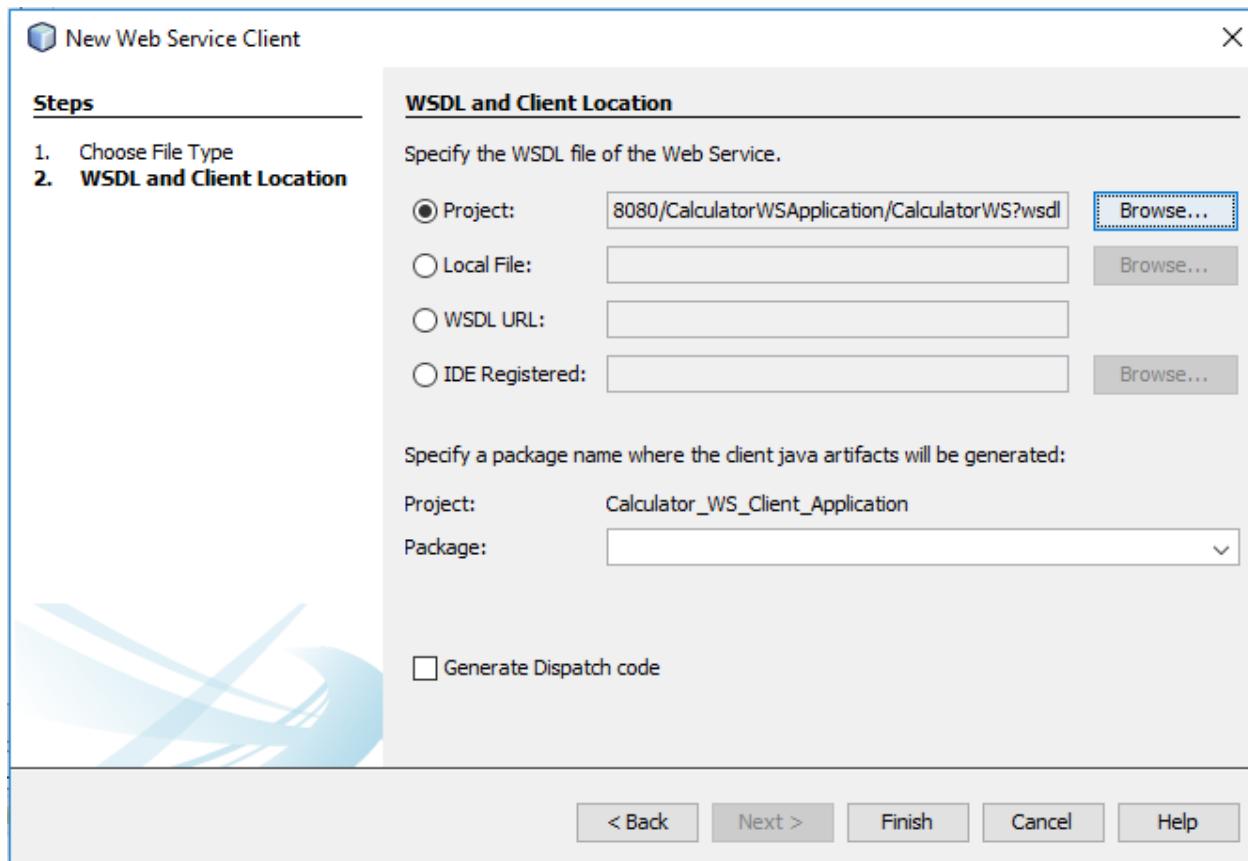
Step 16: Select project as the WSDL File of the Web Service and click on Browse.



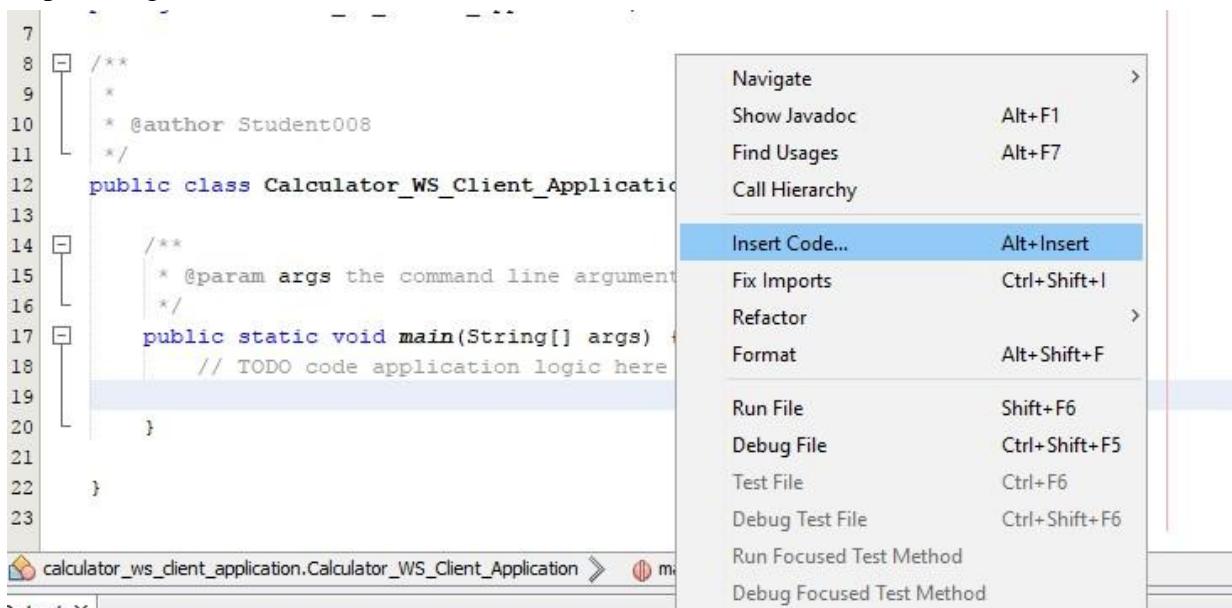
Step 17: Select CalculatorWS. Click on Ok.



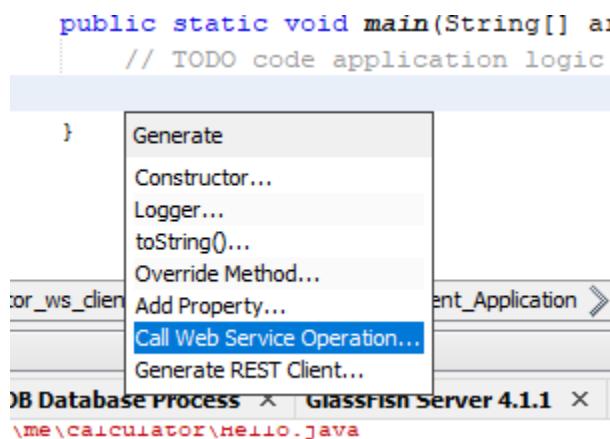
Step 18: A window like this will appear .



Step 17: Right click on the main method area and select “Insert Code”.

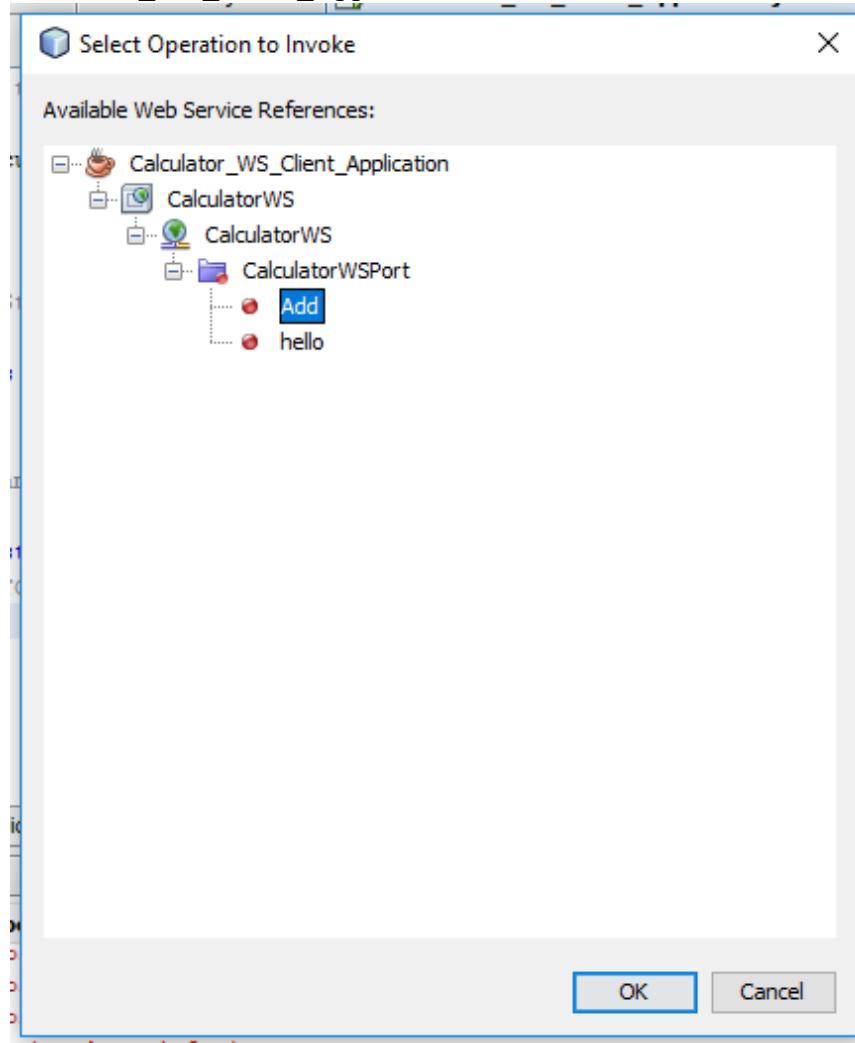


Step 18: Select on the “Call Web Service Operation”.

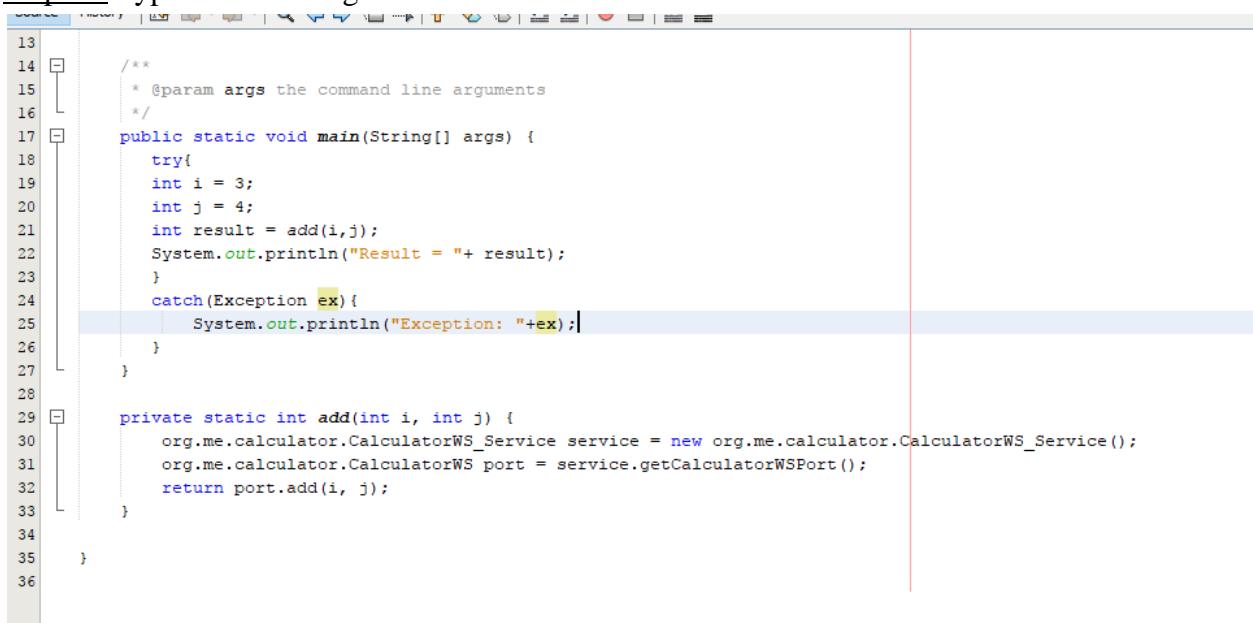


Step 19: Select

Calculator_WS_Client_Application → CalculatorWS → CalculatorWSPort → Add. Click on Ok.



Step 20: Type the following code as shown below.

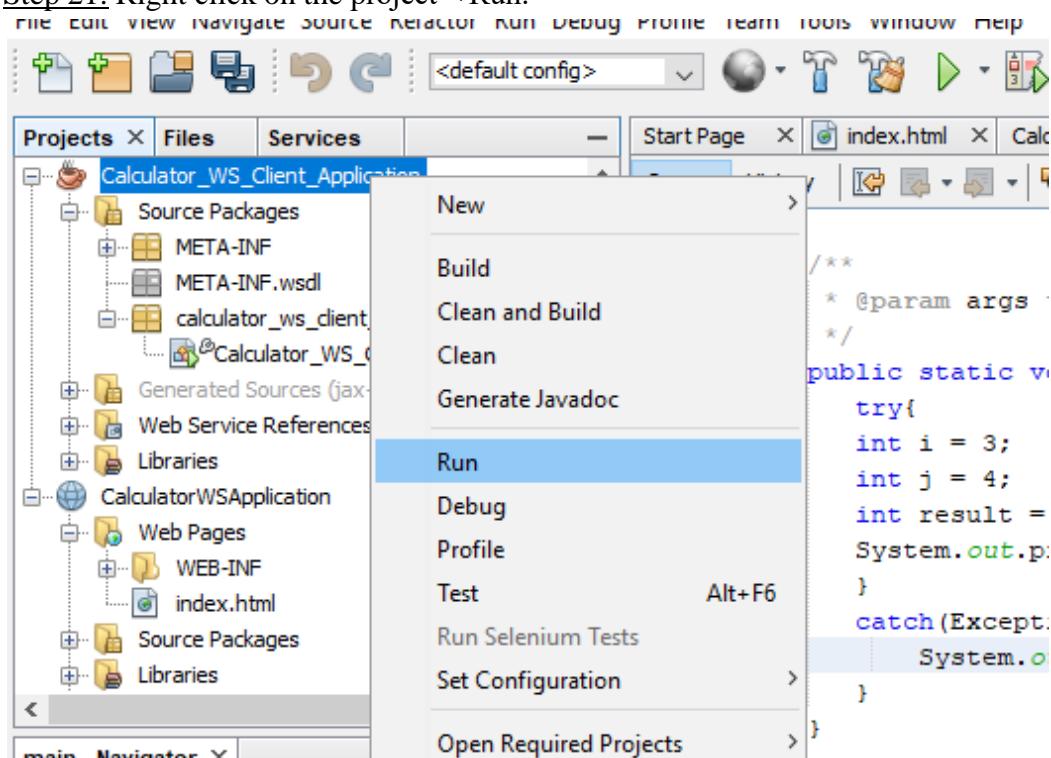


```

13
14     /**
15      * @param args the command line arguments
16     */
17    public static void main(String[] args) {
18        try{
19            int i = 3;
20            int j = 4;
21            int result = add(i,j);
22            System.out.println("Result = "+ result);
23        }
24        catch(Exception ex){
25            System.out.println("Exception: "+ex);
26        }
27    }
28
29    private static int add(int i, int j) {
30        org.me.calculator.CalculatorWS_Service service = new org.me.calculator.CalculatorWS_Service();
31        org.me.calculator.CalculatorWS port = service.getCalculatorWSSPort();
32        return port.add(i, j);
33    }
34
35}
36

```

Step 21: Right click on the project→Run.



OUTPUT:

The screenshot shows the NetBeans IDE's Output window. It has tabs for 'Output' (selected), 'Java DB Database Process', 'GlassFish Server 4.1.1', 'Retriever Output', and 'Calculator_WS_Clie'. The 'Output' tab displays the following build logs:

```
Created dir: C:\Users\Student\Documents\NetBeansProjects\Calculator_WS_Clien
Created dir: C:\Users\Student\Documents\NetBeansProjects\Calculator_WS_Clien
Created dir: C:\Users\Student\Documents\NetBeansProjects\Calculator_WS_Clien
Compiling 9 source files to C:\Users\Student\Documents\NetBeansProjects\Calc
Copying 1 file to C:\Users\Student\Documents\NetBeansProjects\Calculator_WS_
Copied 2 empty directories to 1 empty directory under C:\Users\Student\Docum
compile:
run:
Result = 7
BUILD SUCCESSFUL (total time: 1 second)
```

BSCS602

Pratik Patil

Roll No:58

KERALEEYA SAMAJAM(REGD.) DOMBIVLI'S
MODEL COLLEGE
EMPOWERED AUTONOMOUS

PRACTICAL 6

Date:

Aim: Installation and configuration of virtualization using KVM.**Objectives:** From this experiment, the student will be able to,

- Understand the concept of virtualization.
- Understand KVM architecture and its configuration.

Outcomes: The learner will be able to

- Analyze user models and develop user centric interfaces.
- To analyze local and global impact of computing on individuals, organizations and society.
- To engage in life-long learning development and higher studies.
- To understand, identify, analyze and design the problem, implement and validate the solutions including both hardware and software.

Hardware / Software Required: Ubuntu operating system, open-source software KVM, Internet.**Theory:**

Virtualization is software that separates physical infrastructures to create various dedicated resources. It is the fundamental technology that powers cloud computing. The technology behind virtualization is known as a virtual machine monitor(VMM) or virtual manager, which separates compute environments from the actual physical infrastructure.

Virtualizations make servers, workstations, storage and other systems independent of the physical hardware layer. This is done by installing a Hypervisor on top of the hardware layer, where the systems are then installed.

There are three areas of IT where virtualization is making headboards , network virtualization, storage virtualization and server virtualization:

- Network virtualization is a method of combining the available resources in a network by splitting up the available bandwidth into channels , each of which is independent from the others, and each of which can be assigned(or reassigned) to a particular server or device in real time. The idea is that virtualization disguises the true complexity of the network by separating it into manageable parts , much like your partitioned hard drive makes it easier to manage your files.

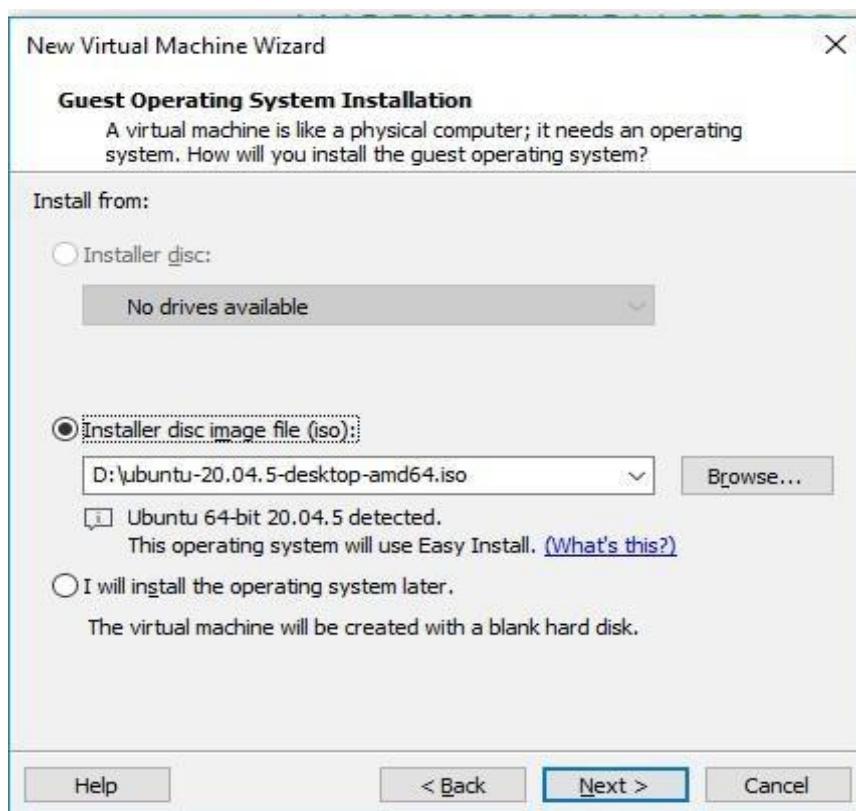
Step 1: Open VMware to create a new virtual machine and select the option “Create New Virtual Machine”.



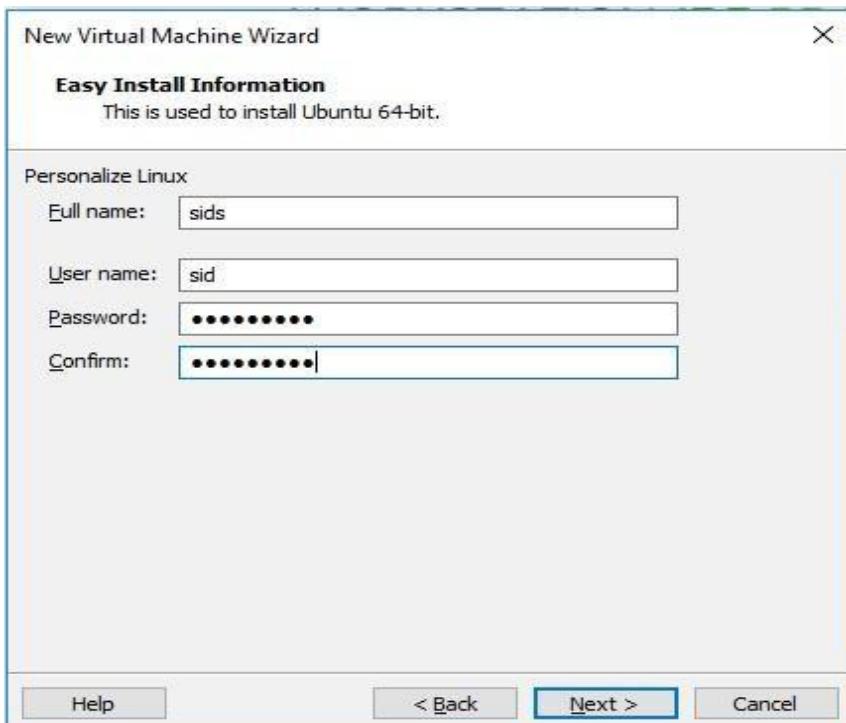
Step 2: Select the “Typical(recommended)” option and click on Next.



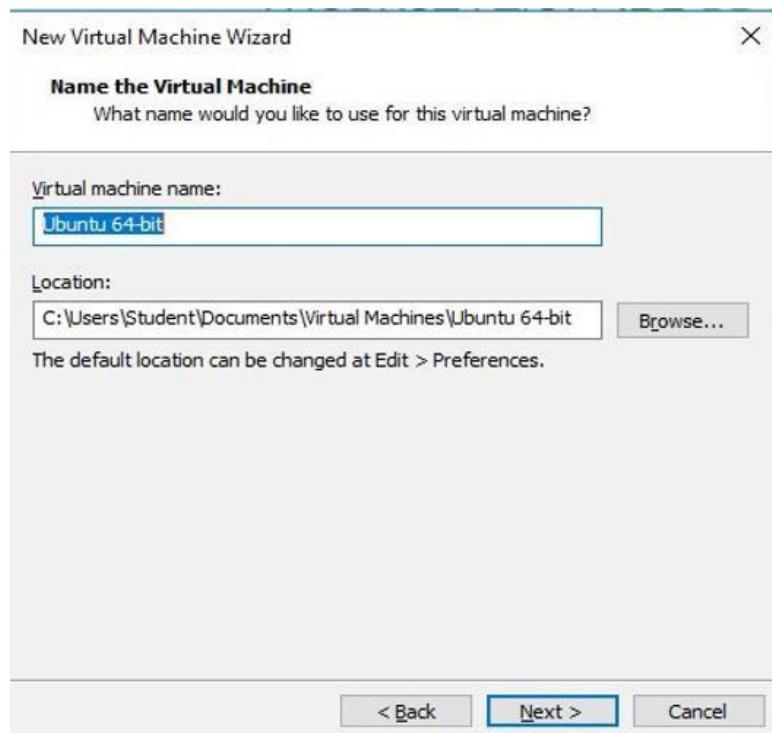
Step 3: The following window will appear click on Next and proceed.



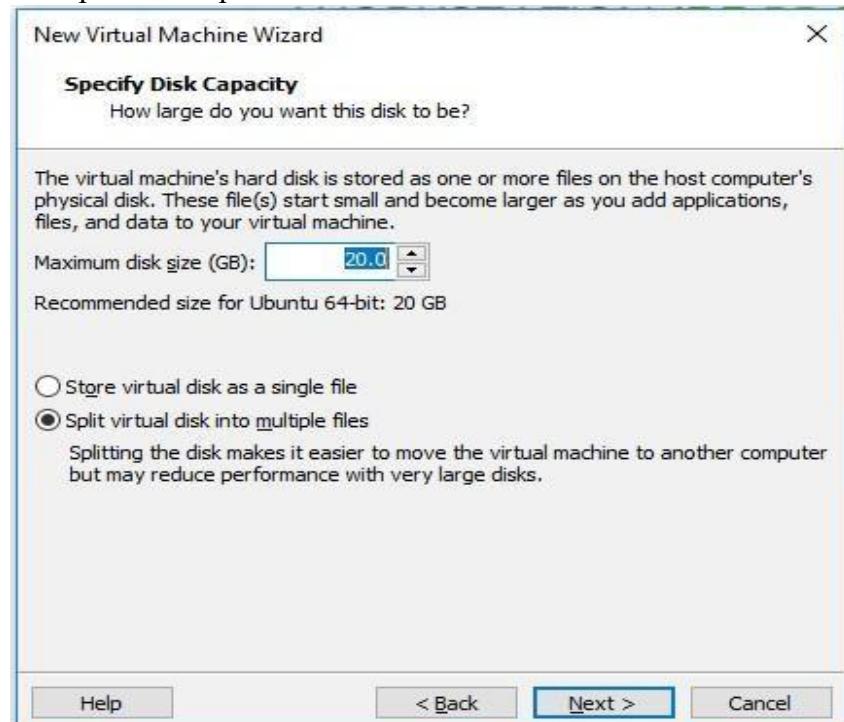
Step 4: Fill the following fields such as the Full name, Username and Password . Click on Next.



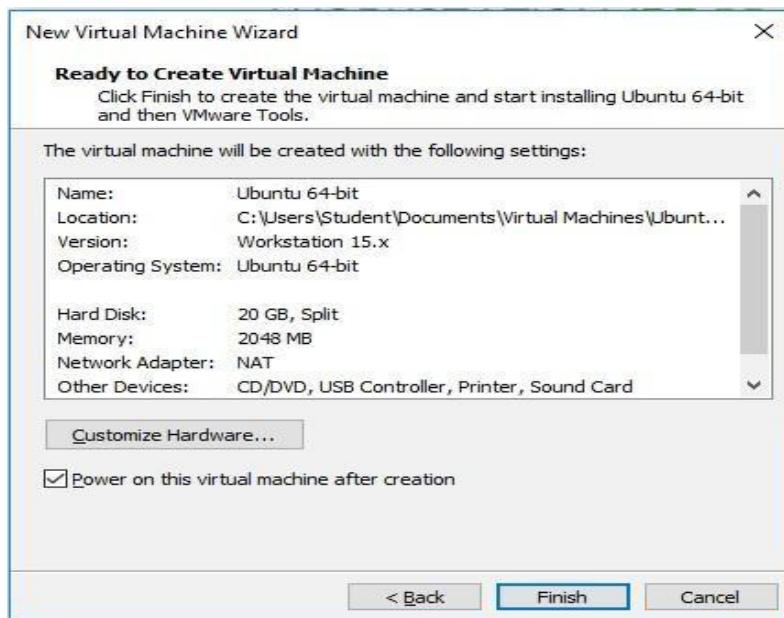
Step 5: The following fields show the Name and Location where the Virtual Machine Is Stored



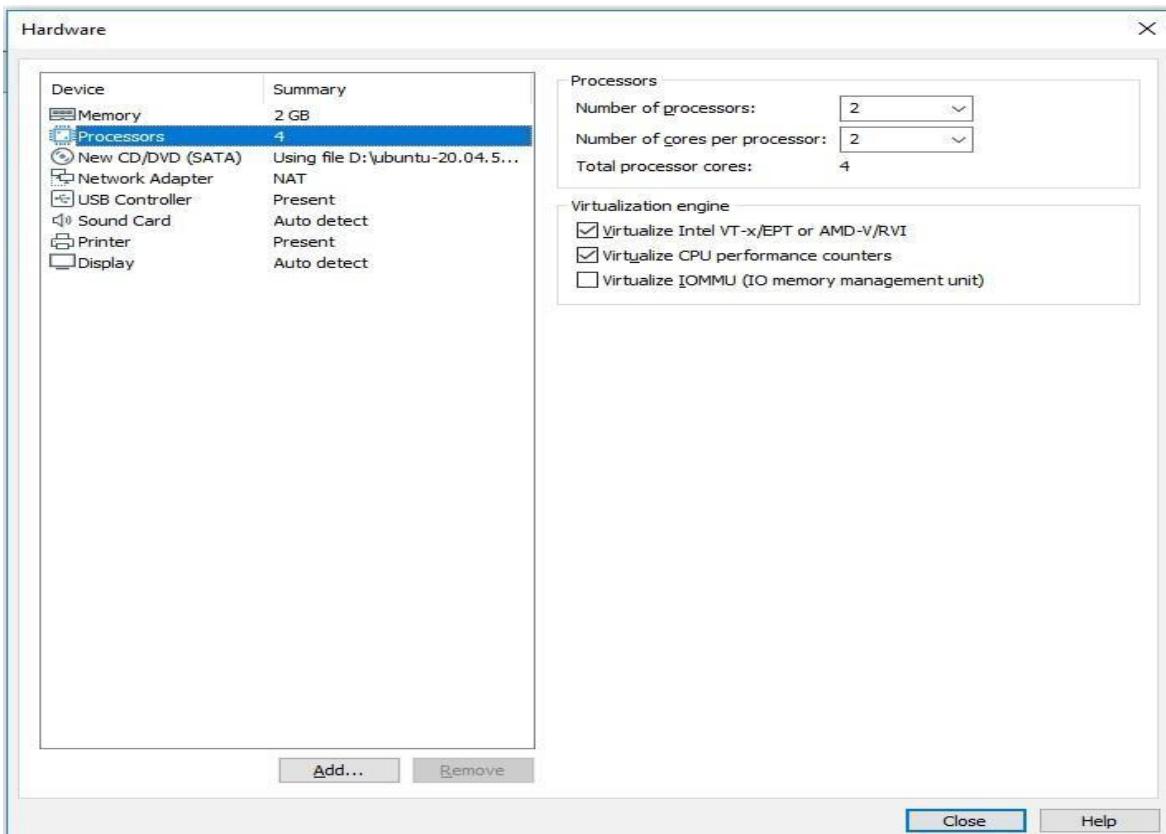
Step 6: Keep the maximum disk size as 20.0 GB and select the “Split virtual disk into multiple files” option.



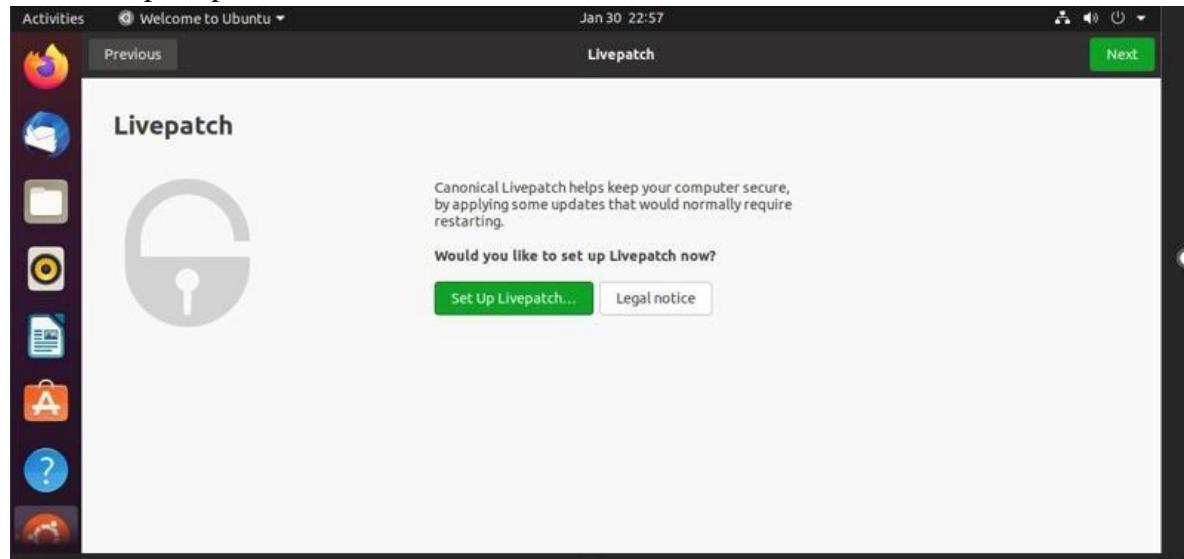
Step 7: Click on the option of “Customize Hardware”.



Step 8: Click on processors and keep the value of Number of processors and Number of core processors as 2. And make sure that the below two checkboxes are ticked . Click on close and then on finish.



Step 9: The following window will appear once the virtual machine starts, go to the command prompt section



Step 10: Update Ubuntu by executing following commands

- Sudo apt update
 - Sudo apt upgrade

```
sid@ubuntu: ~/Desktop$ sudo apt update
[sudo] password for sid:
Hit:1 http://us.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://security.ubuntu.com/ubuntu focal-security InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
425 packages can be upgraded. Run 'apt list --upgradable' to see them.
sid@ubuntu: ~/Desktop$
```

```
sid@ubuntu:~/Desktop$ sudo apt upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following package was automatically installed and is no longer required:
  gir1.2-goa-1.0
Use 'sudo apt autoremove' to remove it.
The following NEW packages will be installed:
  libatomic1 libxmlb2 ubuntu-adantage-desktop-daemon ubuntu-pro-client
  ubuntu-pro-client-l10n
```

Step 11: Check the virtualization support for Ubuntu by executing the following command. This command returns a numeric value ,if it returns 0 it means that our processor is not capable of running KVM. Any other value indicates its KVM.

```
sid@ubuntu:~/Desktop$ egrep -c '(vmx|svm)' /proc/cpuinfo
8
```

Step 12: Check if the system is accelerated by using the command “sudo kvm-ok”. If kvm-ok returns an error then install cpu-checker to resolve the issue .Once the installation is done rerun the command “sudo kvm-ok”.

```
sid@ubuntu:~/Desktop$ sudo kvm-ok
[sudo] password for sid:
sudo: kvm-ok: command not found
sid@ubuntu:~/Desktop$ sudo apt install cpu-checker
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  gir1.2-goa-1.0
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  msr-tools
The following NEW packages will be installed:
  cpu-checker msr-tools
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 16.9 kB of archives.
After this operation, 66.6 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 msr-tools amd64 1.3-3 [10.0 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal/main amd64 cpu-checker amd64 0.7-1.1 [6,936
```

```
sid@ubuntu:~/Desktop$ sudo kvm-ok
INFO: /dev/kvm exists
KVM acceleration can be used
```

Step 13: Install KVM package by executing the following command.

```
sid@ubuntu:~/Desktop$ sudo apt install qemu-kvm libvirt-daemon-system libvirt-clients bridge-utils -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  gir1.2-goa-1.0
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  dmeventd ibverbs-providers ipxe-qemu ipxe-qemu-256k-compat-efi-roms libaio1 libcacard0 libdevmapper-event1.02.
  libfdt1 libibverbs1 libiscsi7 liblvm2cmd2.03 libnss-myhostname libpmem1 librados2 librbd1 librdmacm1 libreadline5
  libslirp0 libspice-server1 libusbsredis-parser1 libvirt-daemon libvirt-daemon-driver-qemu
```

Step 14: Only members of libvirt and KVM usergroups can run virtual machine . If you want a specific user to run a virtual machine then we have to add that user group with the following command.

```
sid@ubuntu:~/Desktop$ sudo adduser 'sid' libvirt
The user 'sid' is already a member of 'libvirt'.
```

Step 15: Do the same thing with the KVM group.

```
sid@ubuntu:~/Desktop$ sudo adduser 'sid' kvm
Adding user `sid' to group `kvm' ...
Adding user sid to group kvm
Done.
```

Step 16: Verify the installation. Confirm the KVM installation with virsh command .

The command will list all active and inactive virtual machines on the system.

```
sid@ubuntu:~/Desktop$ sudo virsh list --all
 Id  Name   State
```

Step 17: Use systemctl command to check the status of libvирtd, the daemon that provides the backend services for libvirt virtualization management system. If the virtualization is not active then activate it with following command

“sudo systemctl enable --now libvирtd”

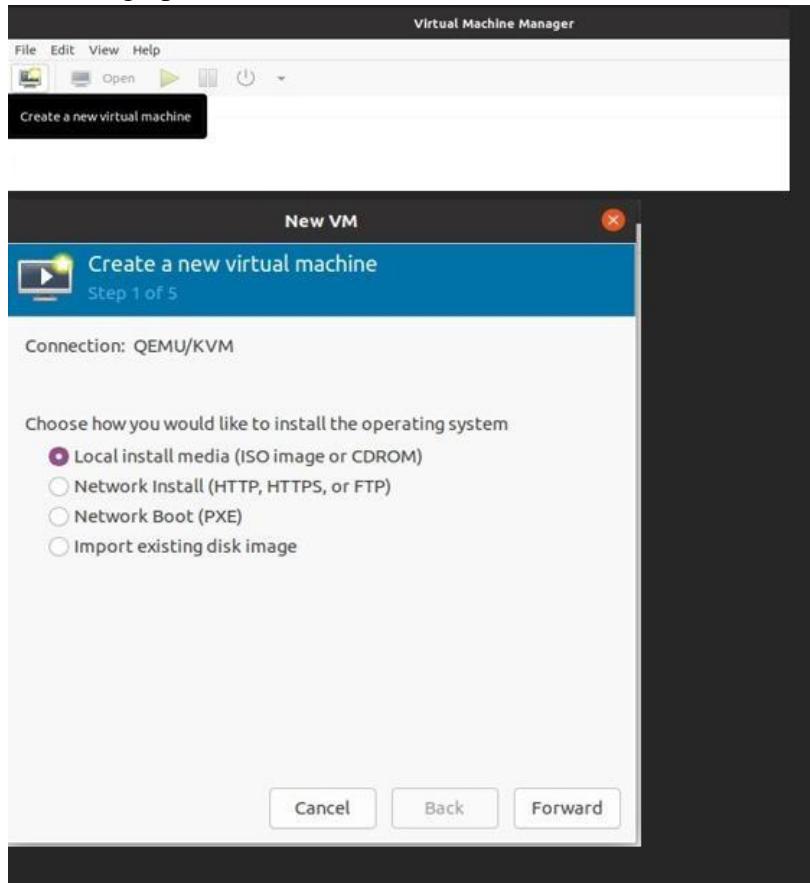
```
sid@ubuntu:~/Desktop$ sudo systemctl status libvирtd
● libvирtd.service - Virtualization daemon
   Loaded: loaded (/lib/systemd/system/libvирtd.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2025-01-30 23:22:05 PST; 35min ago
     TriggeredBy: ● libvирtd-admin.socket
                  ● libvирtd-ro.socket
                  ● libvирtd.socket
   Docs: man:libvирtd(8)
         https://libvirt.org
 Main PID: 54966 (libvирtd)
    Tasks: 19 (limit: 32768)
   Memory: 15.9M
      CGroup: /system.slice/libvирtd.service
```

Step 18: Create a virtual machine on Ubuntu. For creating a virtual machine install virt-manager, it is a tool for creating and managing virtual machines.

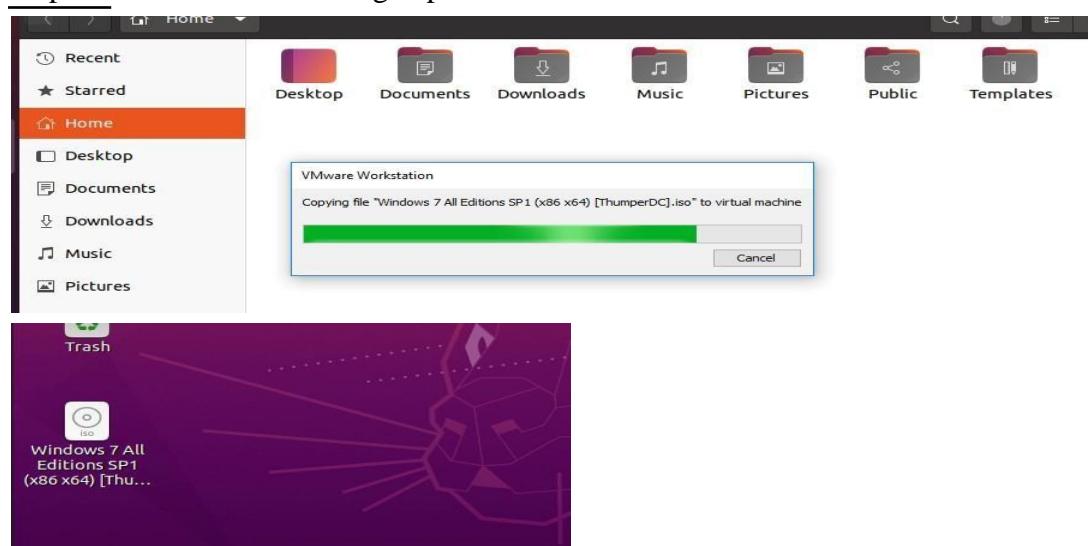
```
sid@ubuntu:~/Desktop$ sudo apt install virt-manager -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  gir1.2-goa-1.0
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  gir1.2-appindicator3-0.1 gir1.2-gtk-vnc-2.0 gir1.2-libosinfo-1.0 gir1.2-libvirt-glib-1.0 gir1.2-
  gir1.2-spiceclientgtk-3.0 i965-va-driver intel-media-va-driver libgovirt-common libgovirt2 libgt
  libgvnc-1.0-0 libigdgmm11 libosinfo-1.0-0 libphodav-2.0-0 libphodav-2.0-common libspice-client-g
  libspice-client-gtk-3.0-5 libusbredirhost1 libva-x11-2 libva2 libvirt-glib-1.0-0 mesa-va-drivers
  python3-distutils python3-libvirt python3-libxml2 spice-client-glib-usb-acl-helper va-driver-all
  virtinst
Suggested packages:
```

```
sid@ubuntu:~/Desktop$ sudo virt-manager
```

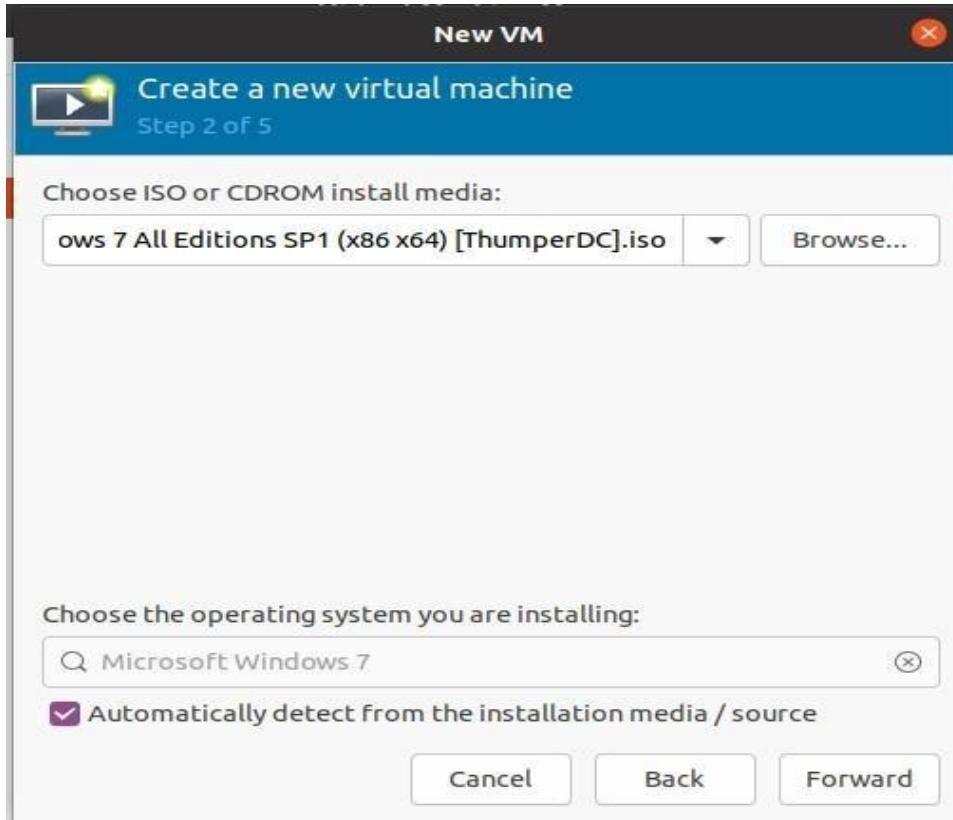
Step 19: Copy the Ubuntu ISO file to the desktop of the virtual machine and click on the Following option.



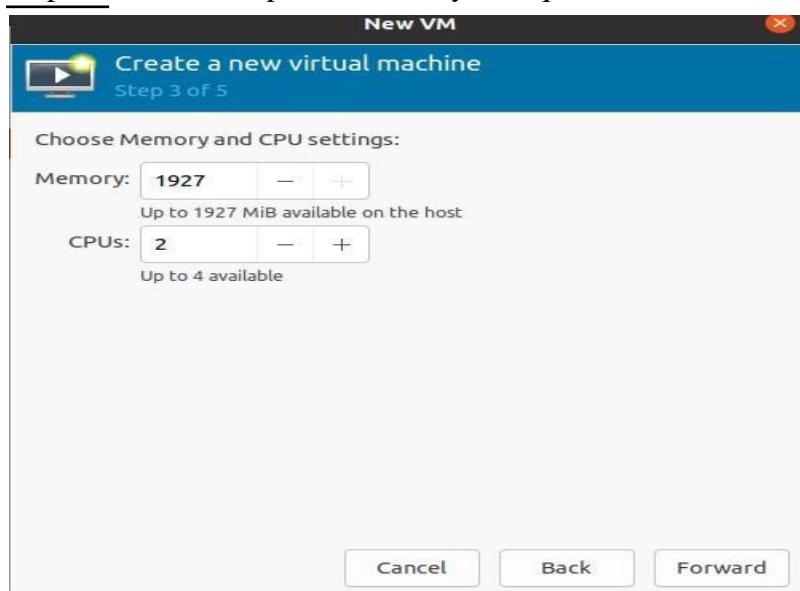
Step 20: Here the file is being copied.



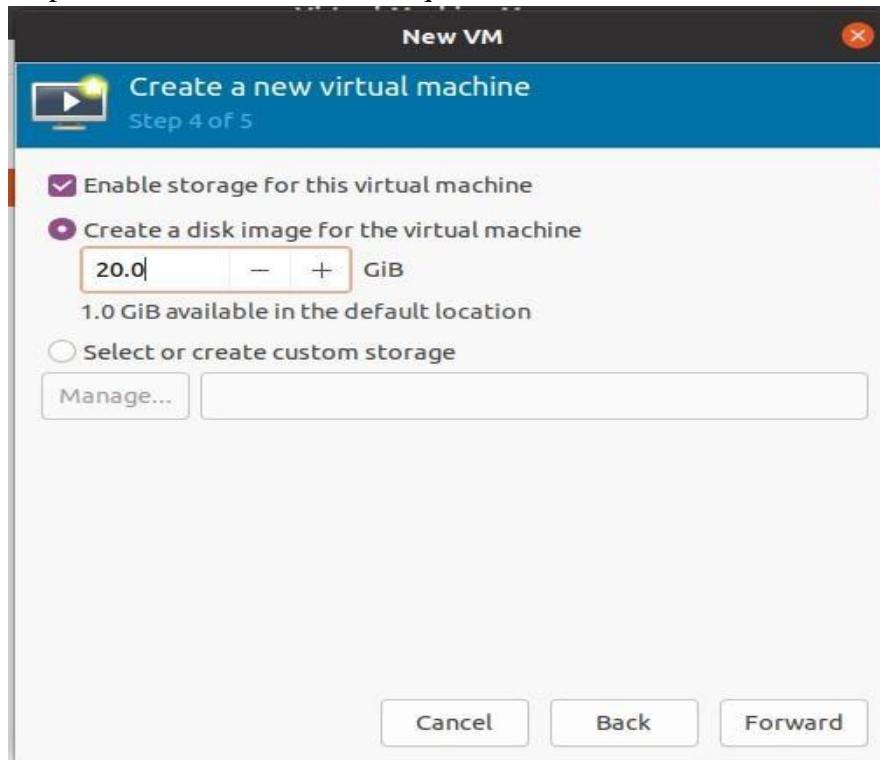
Step 21: Browse the ISO file and tick the checkbox as follows. Click on forward.



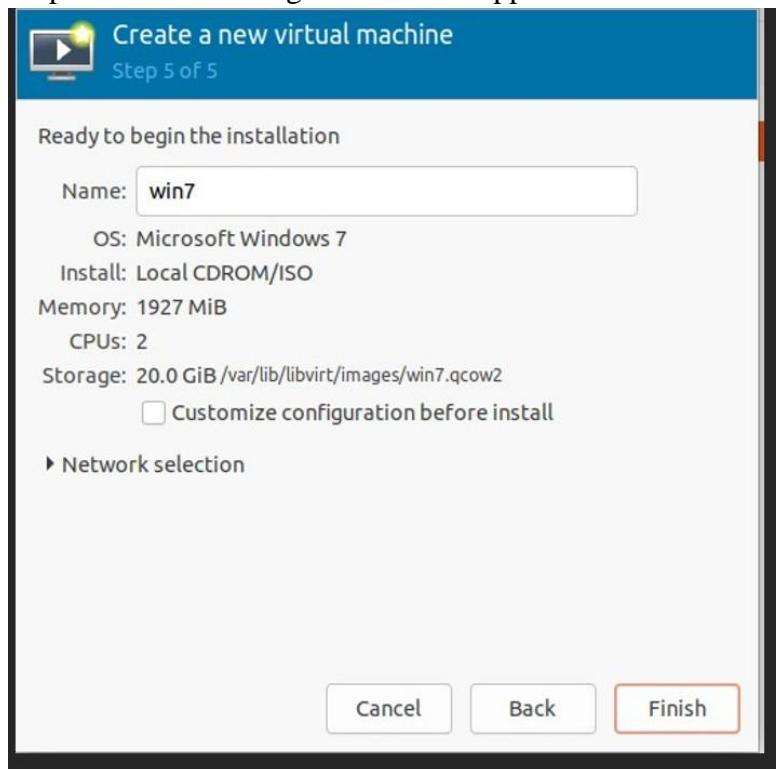
Step 22: Select the cpu and memory as required. Click on forward.



Step 23: Give the disk size as required in that field. Click on forward.



Step 24: The following window will appear which is the final stage of the installation



BSCS602

Pratik Patil

Roll No:58

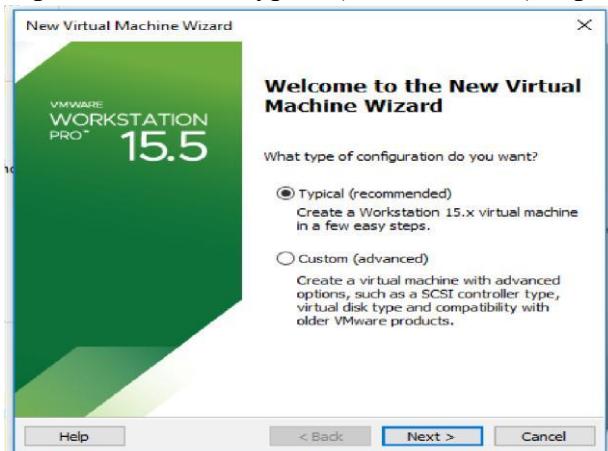
KERALEEYA SAMAJAM(REGD.) DOMBIVLI'S
MODEL COLLEGE
EMPOWERED AUTONOMOUS

Practical 7**Aim : Study and implementation of Infrastructure as a Service.**

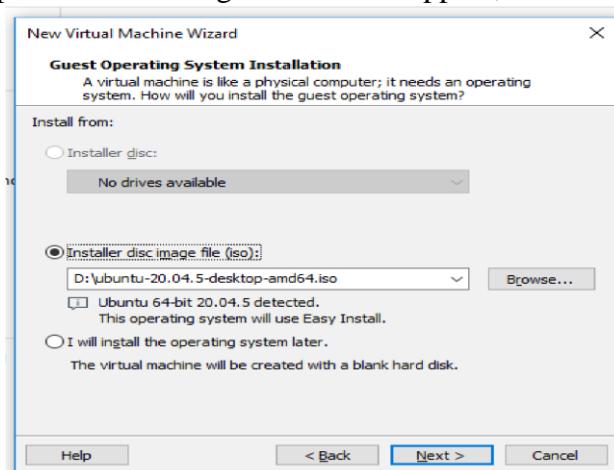
Step 1: Open VMware to create a new virtual machine and select the option “Create a New Virtual Machine”.



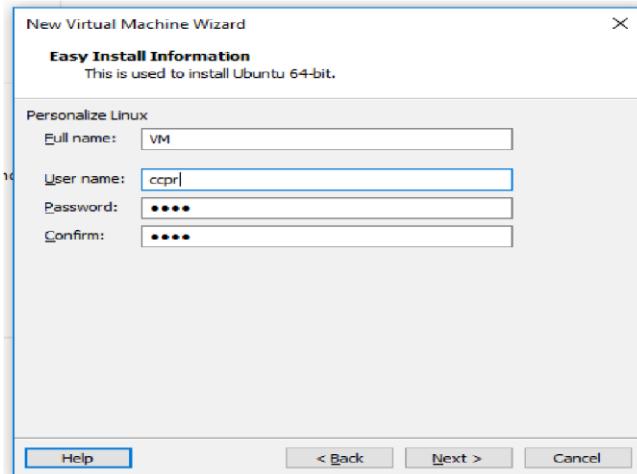
Step 2: Select the “Typical(recommended)” option and click on Next.



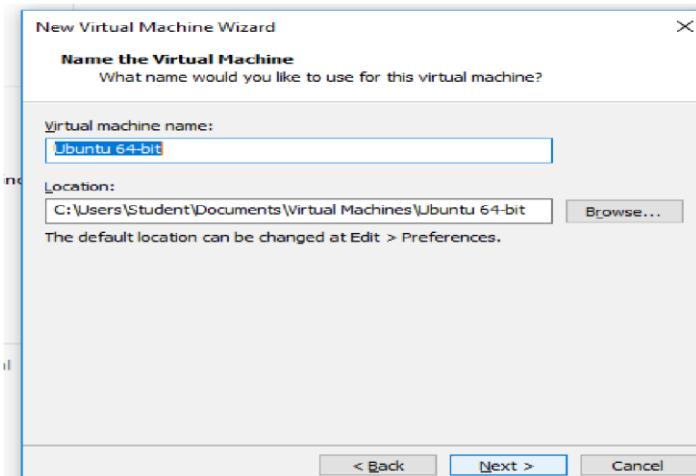
Step 3: The following window will appear, click on Next and proceed.



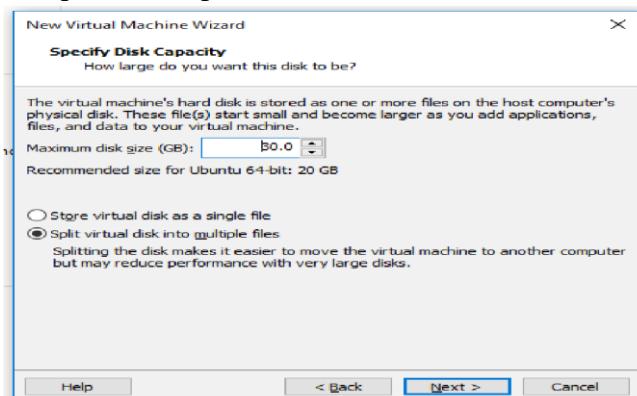
Step 4: Fill the following fields such as the Full name, Username and Password . Click on Next. (Password-ccpr)



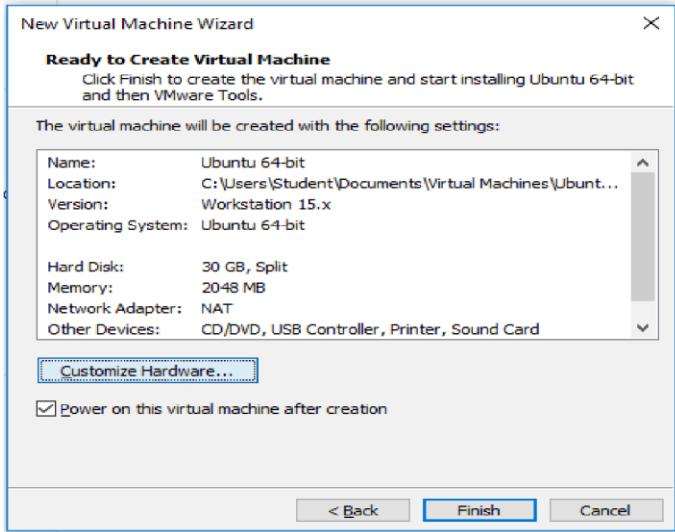
Step 5: The following fields show the Name, Location where the Virtual Machine is stored.



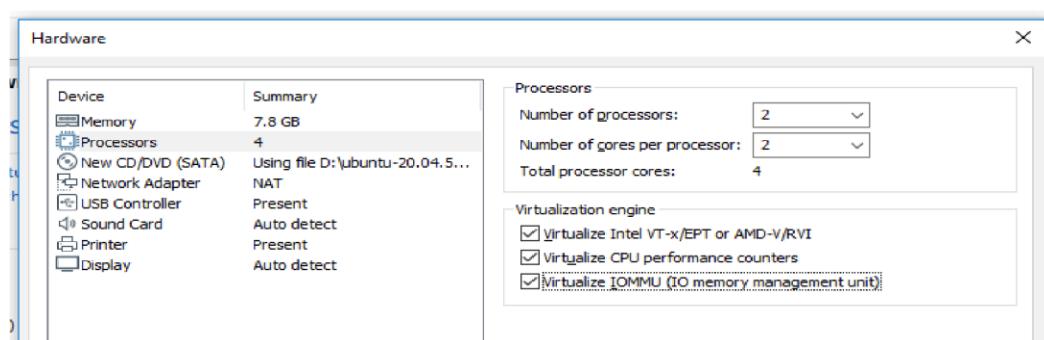
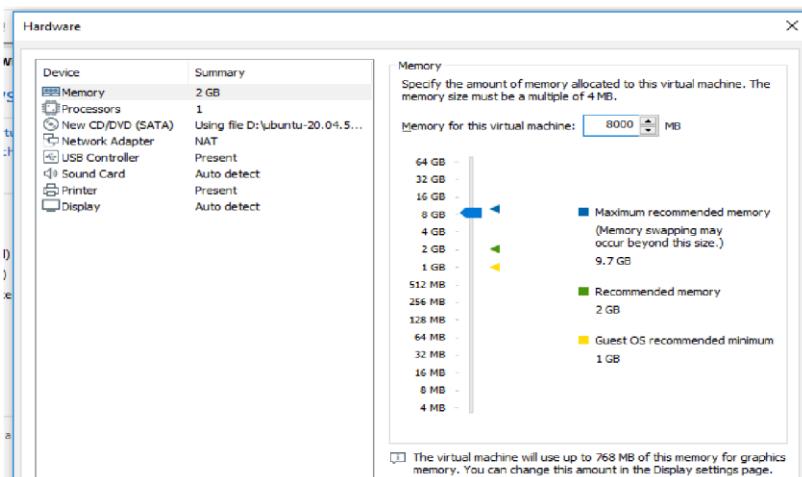
Step 6: Keep the maximum disk size as 30.0 GB and select the “Split virtual disk into multiple files” option. Click on Next.

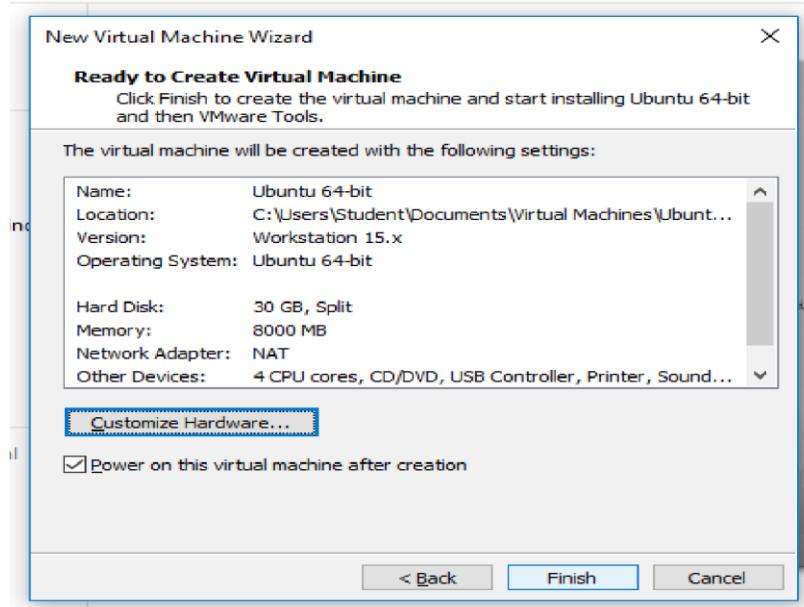
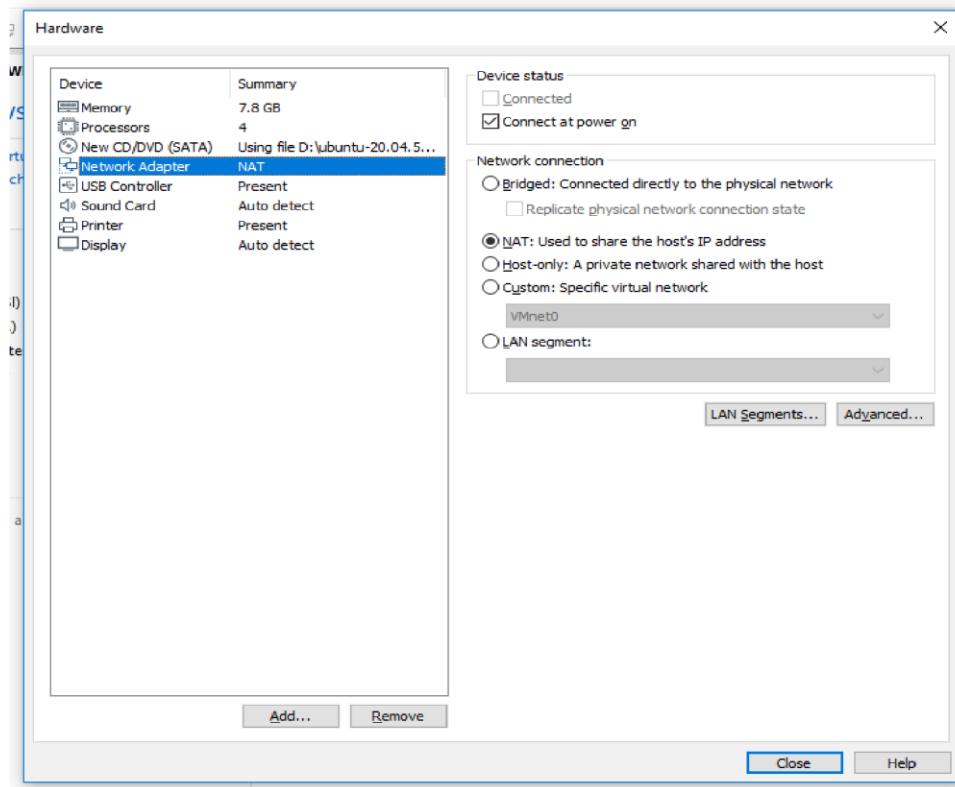


Step 7: Click on the option of “Customize Hardware”.

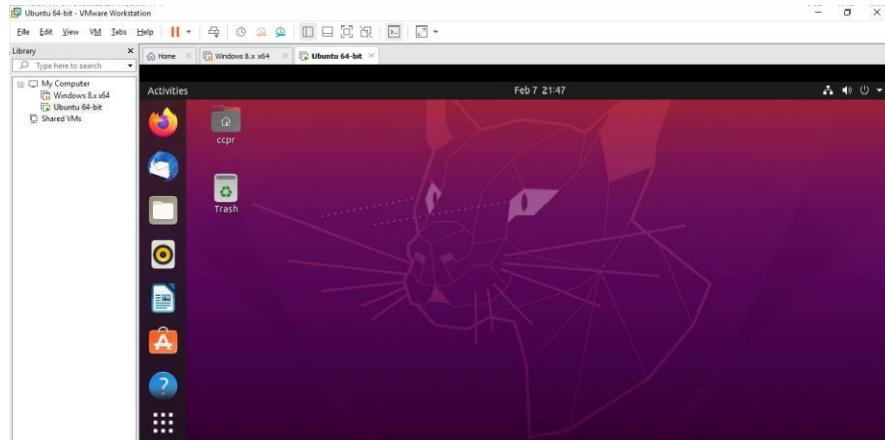


Step 8: Click on Memory and set the memory as 8000 MB and then click on processors and keep the value of Number of processors and Number of core processors as 2. And make sure that the below two checkboxes are ticked . Click on close and then on finish.





Step 9: Now the following window will appear then open command prompt.



Step 10: Update Ubuntu by executing following commands

- Sudo apt update
- Sudo apt upgrade

```
ccpr@ubuntu:~/Desktop
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

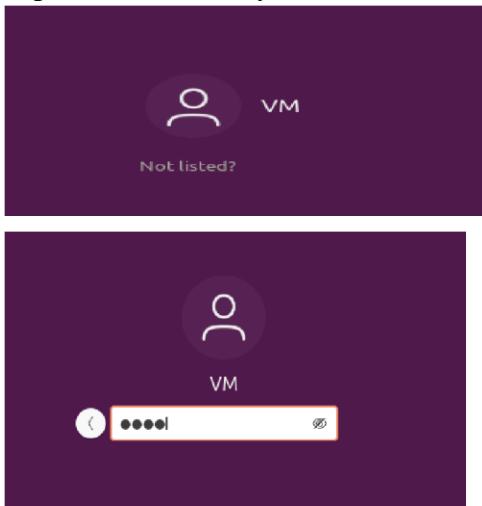
ccpr@ubuntu:~/Desktop$ sudo apt-get update
[sudo] password for ccpr:
Hit:1 http://us.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu focal-security InRelease
Reading package lists... Done
ccpr@ubuntu:~/Desktop$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages have been kept back:
  fwupd gir1.2-javascriptcoregtk-4.0 gir1.2-webkit2-4.0 libfwupd2
  libfwupdplugin5 libjavascriptcoregtk-4.0-18 libwebkit2gtk-4.0-37
  python3-software-properties python3-update-manager
  software-properties-common software-properties-gtk ubuntu-adantage-tools
  update-manager update-manager-core
The following packages will be upgraded:
  accountsservice amd64-microcode apparmor apport apport-gtk apt apt-utils
  avahi-autoipd avahi-daemon avahi-utils base-files bind9-dnsutils bind9-host

Processing triggers for initramfs-tools (0.136ubuntu6.7) ...
update-initramfs: Generating /boot/initrd.img-5.15.0-131-generic
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for gnome-menus (3.36.0-1ubuntu1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.17) ...
Processing triggers for rsyslog (8.2001.0-1ubuntu1.3) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for cracklib-runtime (2.9.6-3.2) ...
Processing triggers for plymouth-theme-ubuntu-text (0.9.4git20200323-0ubuntu6.2)
...
update-initramfs: deferring update (trigger activated)
Processing triggers for shared-mime-info (1.15-1) ...
Processing triggers for install-info (6.7.0.dfsg.2-5) ...
Processing triggers for fontconfig (2.13.1-2ubuntu3) ...
Processing triggers for desktop-file-utils (0.24-1ubuntu3) ...
Processing triggers for ca-certificates (20240203-20.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
Processing triggers for libgdk-pixbuf2.0-0:amd64 (2.40.0+dfsg-3ubuntu0.5) ...
Processing triggers for initramfs-tools (0.136ubuntu6.7) ...
update-initramfs: Generating /boot/initrd.img-5.15.0-131-generic
```

Step 11: Now reboot the system by the ‘sudo reboot’ or ‘init 6’ command.

```
ccpr@ubuntu:~/Desktop$ sudo reboot
```

Step 12: Now select your created machine and enter the respective password.



Step 13: Create new user called stack

```
ccpr@ubuntu:~/Desktop$ sudo adduser stack
[sudo] password for ccpr:
Adding user `stack' ...
Adding new group `stack' (1001) ...
Adding new user `stack' (1001) with group `stack' ...
Creating home directory `/home/stack' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
passwd: password unchanged
Try again? [y/N] y
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for stack
Enter the new value, or press ENTER for the default
```

Step 14: Log in as root.

```
ccpr@ubuntu:~/Desktop$ sudo -i
root@ubuntu:~# 
```

Step 15: Assign the sudo privileges to stack user as follows (#echo “stack ALL=(ALL) NOPASSWD:ALL” | sudo tee /etc/sudoers.d/stack)

```
root@ubuntu:~# echo "stack ALL=(ALL) NOPASSWD:ALL" | sudo tee /etc/sudoers.d/stack
stack ALL=(ALL) NOPASSWD:ALL
root@ubuntu:~# 
```

Step 16: Switch to the stack user, by following command “sudo su - stack”.

```
root@ubuntu:~# sudo su - stack
stack@ubuntu:~$
```

Step 17: Install the Git using the command “sudo apt install git -y”.

```
stack@ubuntu:~$ sudo apt install git -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk
  gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 14 not upgraded.
Need to get 5,525 kB of archives.
After this operation, 38.8 MB of additional disk space will be used.
0% [Working]
```

Step 18: Download OpenStack

Once Git is installed, download the DevStack from github by following command “git clone https://git.openstack.org/openstack-dev/devstack”.

```
stack@ubuntu:~$ git clone https://git.openstack.org/openstack-dev/devstack
Cloning into 'devstack'...
warning: redirecting to https://opendev.org/openstack/devstack/
remote: Enumerating objects: 51481, done.
remote: Counting objects: 100% (31304/31304), done.
remote: Compressing objects: 100% (10571/10571), done.
remote: Total 51481 (delta 30539), reused 20733 (delta 20733), pack-reused 20177
Receiving objects: 100% (51481/51481), 9.67 MiB | 175.00 KiB/s, done.
Resolving deltas: 100% (36543/36543), done.
stack@ubuntu:~$
```

Step 19: Go to DevStack directory and look for local.conf file.

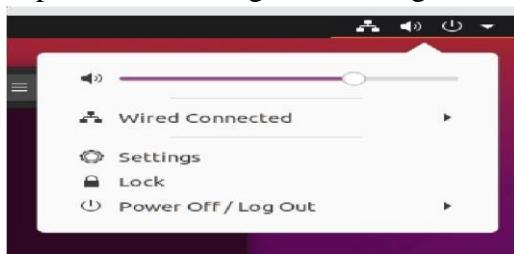
```
stack@ubuntu:~$ ls
devstack
stack@ubuntu:~$ cd devstack
stack@ubuntu:~/devstack$ ls
clean.sh      functions      lib       roles      tools
CONTRIBUTING.rst  functions-common LICENSE  run_tests.sh tox.ini
data          FUTURE.rst    Makefile   samples    unstack.sh
doc           gate          openrc    stackrc
extras.d      HACKING.rst  playbooks  stack.sh
files          inc          README.rst tests
stack@ubuntu:~/devstack$ cd samples
stack@ubuntu:~/devstack/samples$ ls
local.conf  local.sh
stack@ubuntu:~/devstack/samples$ cp local.conf ../
stack@ubuntu:~/devstack/samples$ cd ..
stack@ubuntu:~/devstack$ ls
clean.sh      functions      lib       README.rst  tests
CONTRIBUTING.rst  functions-common LICENSE  roles      tools
data          FUTURE.rst    local.conf run_tests.sh tox.ini
doc           gate          Makefile   samples    unstack.sh
extras.d      HACKING.rst  openrc    stackrc
files          inc          playbooks  stack.sh
stack@ubuntu:~/devstack$
```

Step 20: Install vim using the command “sudo apt-get install vim”.

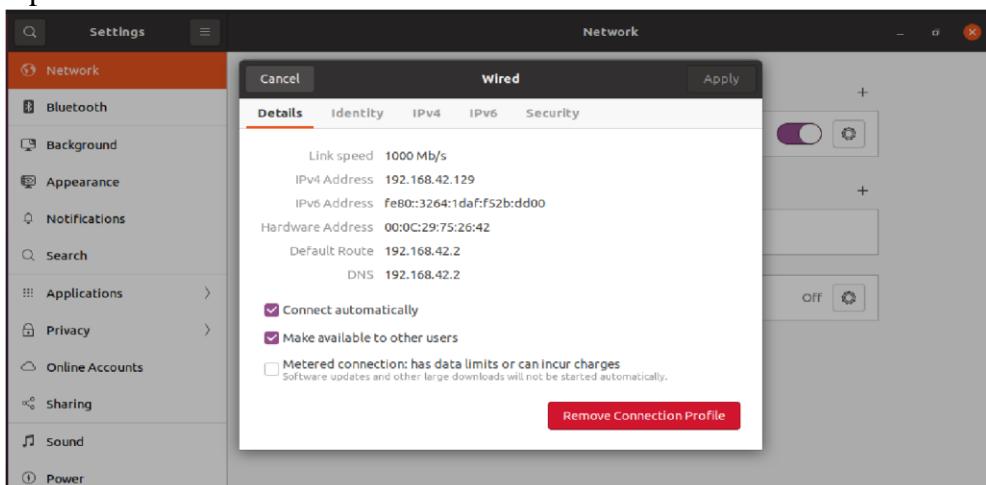
```
stack@ubuntu:~/devstack$ sudo apt-get install vim
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  vim-runtime
Suggested packages:
  ctags vim-doc vim-scripts
The following NEW packages will be installed:
  vim vim-runtime
0 upgraded, 2 newly installed, 0 to remove and 14 not upgraded.
Need to get 7,117 kB of archives.
After this operation, 34.6 MB of additional disk space will be used.
Do you want to continue? [Y/n]

Adding 'diversion of /usr/share/vim/vim81/doc/tags to /usr/share/vim/vim81/doc/tags.vim-tiny by vim-runtime'
Unpacking vim-runtime (2:8.1.2269-1ubuntu5.30) ...
Selecting previously unselected package vim.
Preparing to unpack .../vim_2%3a8.1.2269-1ubuntu5.30_amd64.deb ...
Unpacking vim (2:8.1.2269-1ubuntu5.30) ...
Setting up vim-runtime (2:8.1.2269-1ubuntu5.30) ...
Setting up vim (2:8.1.2269-1ubuntu5.30) ...
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vim (vim) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vimdiff (vimdiff) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/rvim (rvim) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/rview (rview) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vi (vi) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/view (view) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/ex (ex) in auto mode
Processing triggers for man-db (2.9.1-1) ...
```

Step 21: For fetching IP address go to ‘Wired Connected’ option.



Step 22: Now note down the IPv4 Address. Here “192.168.42.129”



Step 23: Now open the local.conf file using command “vim local.conf”.

```
stack@ubuntu:~/devstack$ vim local.conf
```

Step 24: Once the file is opened go to insert mode and then after the device_password field write the system IP address that we copied from settings above “HOST_IP=192.168.42.129”

```
# Note that if ``localrc`` is present it will be used in favor of this section.
[[local|localrc]]

# Minimal Contents
# ----

# While ``stack.sh`` is happy to run without ``localrc``, devlife is better when
# there are a few minimal variables set:

# If the ``*_PASSWORD`` variables are not set here you will be prompted to enter
# values for them by ``stack.sh`` and they will be added to ``local.conf``.
ADMIN_PASSWORD=p1
DATABASE_PASSWORD=p1
RABBIT_PASSWORD=p1
SERVICE_PASSWORD=p1

# ``HOST_IP`` and ``HOST_IPV6`` should be set manually for best results if
# the NIC configuration of the host is unusual i.e. ``eth1`` has the default
HOSTIP=192.168.42.129

# The ``localrc`` section replaces the old ``localrc`` configuration file.
# Note that if ``localrc`` is present it will be used in favor of this section.
[[local|localrc]]

# Minimal Contents
# ----

# While ``stack.sh`` is happy to run without ``localrc``, devlife is better when
# there are a few minimal variables set:

# If the ``*_PASSWORD`` variables are not set here you will be prompted to enter
# values for them by ``stack.sh`` and they will be added to ``local.conf``.
ADMIN_PASSWORD=p1
DATABASE_PASSWORD=p1
RABBIT_PASSWORD=p1
SERVICE_PASSWORD=p1
HOST_IP=192.168.56.1

# ``HOST_IP`` and ``HOST_IPV6`` should be set manually for best results if
# the NIC configuration of the host is unusual, i.e. ``eth1`` has the default
# route but ``eth0`` is the public interface. They are auto-detected in
# ``stack.sh`` but often is indeterminate on later runs due to the IP moving
# from an Ethernet interface to a bridge on the host. Setting it here also
-- INSERT --                                32,21      19%
```

Step 26: To install and run the OpenStack, execute the following commands “./stack.sh”

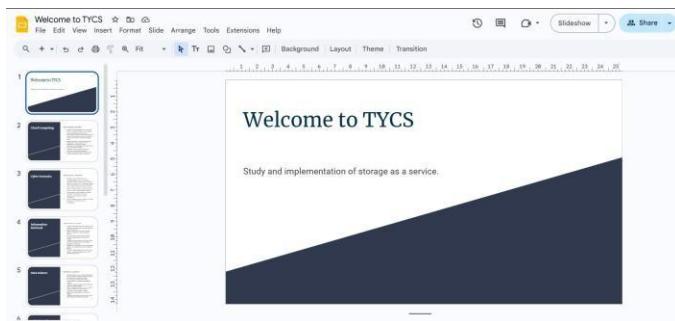
```
stack@ubuntu:~/devstack$ ./stack.sh
+ unset GREP_OPTIONS
+ unset LANG
+ unset LANGUAGE
+ LC_ALL=en_US.UTF-8
+ export LC_ALL
++ env
++ grep -E '^OS_'
++ cut -d = -f 1
+ unset
+ umask 022
+ PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/usr/local/bin:/usr/local/sbin:/usr/sbin:/sbin
+++ dirname ./stack.sh
++ cd .
++ pwd
+ TOP_DIR=/home/stack/devstack
+ NOUNSET=
+ [[ -n '' ]]
++ date +%
+ DEVSTACK_START_TIME=1739782880
+ [[ -r /home/stack/devstack/.stackenv ]]
+ FILES=/home/stack/devstack/files
+ '[' '!' -d /home/stack/devstack/files ']'
```

BSCS602

Pratik Patil

Roll No:58

KERALEEYA SAMAJAM(REGD.) DOMBIVLI'S
MODEL COLLEGE
EMPOWERED AUTONOMOUS

Practical 8**Aim : Study and implementation of Storage as a Service.****On the google drive - create docs. - presentation 10 slides(style and animation) - google form 10 questions (different format) - spreadsheet (calculations) - share.****Presentation :**

Cloud Computing

Cloud Computing - Key Points

1. Definition: Cloud computing provides on-demand access to computing resources over the internet without requiring physical infrastructure.
2. Service Models: It includes IaaS (Infrastructure as a Service), PaaS (Platform as a Service), and SaaS (Software as a Service) for different computing needs.
3. Deployment Models: It can be Public (shared by multiple users), Private (exclusive to one organization), or Hybrid (mix of both).
4. Advantages: Offers scalability, cost savings, remote access, and automatic updates.
5. Challenges: Includes security risks, internet dependency, and regulatory compliance.
6. Examples: Popular cloud platforms include AWS, Google Cloud, Microsoft Azure, and Dropbox.

Data Science

Data Science – Key Points

1. Definition: Data Science is an interdisciplinary field that uses statistics, machine learning, and domain knowledge to extract insights from data.
2. Key Components: Includes data collection, cleaning, analysis, visualization, and predictive modeling.
3. Techniques: Machine learning, deep learning, data mining, and statistical analysis.
4. Tools & Technologies: Python, R, SQL, TensorFlow, Pandas, and Hadoop are commonly used.
5. Applications: Used in healthcare, finance, marketing, artificial intelligence, and business analytics.
6. Challenges: Data quality, privacy concerns, model interpretability, and handling large datasets.

Information Retrieval

Information Retrieval – Key Points

1. Definition: Information retrieval (IR) is the process of obtaining relevant information from large datasets or document collections.
2. Objective: Helps users find useful information quickly from structured or unstructured data sources.
3. Key Components: Includes query processing, indexing, ranking algorithms, and relevance feedback.
4. Techniques: Boolean search, vector space model, machine learning-based retrieval, and natural language processing (NLP).
5. Applications: Search engines (Google, Bing), library systems, recommendation systems, and big data analysis.
6. Challenges: Handling large-scale data, improving accuracy, dealing with ambiguous queries, and optimizing response time.

Cyber Forensics

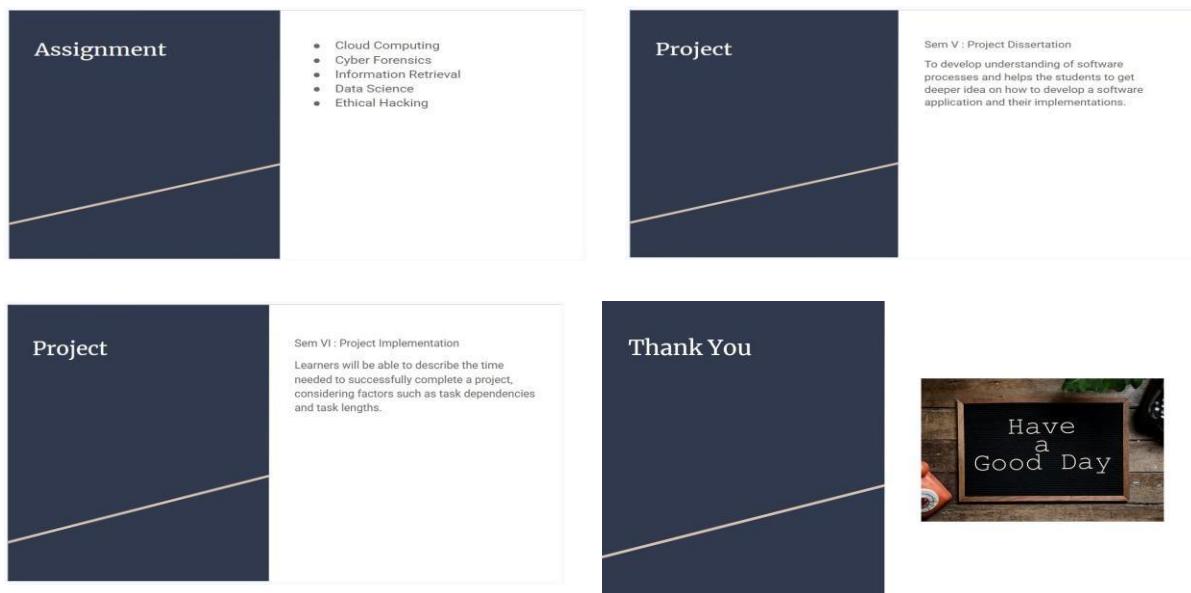
Cyber Forensics – Key Points

1. Definition: Cyber forensics involves investigating digital crimes by collecting, analyzing, and preserving electronic evidence.
2. Objectives: Identify, recover, and analyze digital data to support legal cases and cybersecurity.
3. Types: Includes computer forensics, network forensics, mobile forensics, and cloud forensics.
4. Process: Involves data acquisition, analysis, documentation, and presentation of findings.
5. Challenges: Encryption, anti-forensic techniques, data volatility, and legal complexities.
6. Tools & Examples: EnCase, Autopsy, FTK, and Wireshark are commonly used for investigations.

Ethical Hacking

Ethical Hacking – Key Points

1. Definition: Ethical hacking involves legally testing and securing computer systems to identify and fix vulnerabilities.
2. Objective: Helps organizations strengthen cybersecurity by simulating real cyberattacks.
3. Types: Includes penetration testing, web application hacking, network security testing, and social engineering.
4. Techniques & Tools: Uses methods like password cracking, SQL injection, and tools like Metasploit, Nmap, and Wireshark.
5. Applications: Used in cybersecurity, risk assessment, compliance testing, and securing sensitive data.
6. Challenges: Legal and ethical boundaries, evolving threats, and staying updated with security trends.



Google Form :

← Preview mode Published Copy responder link

CC_Pr8 Form

The name, email address and photo associated with your Google Account will be recorded when you upload files and submit this form.

What is your name?

Your answer _____

What is your DOB?

Date _____
dd-mm-yyyy

Select your gender

Male
 Female

← Preview mode Published Copy responder link

what are your hobbies?

Singing
 Dancing
 Drawing
 Reading

Tell me about yourself.

Your answer _____

Upload your identity proof.

Upload 1 supported file: PDF or image. Max 10 MB.

BSCS602

Pratik Patil

Roll No:58

← Preview mode

Published Copy responder link

Which social media platforms do you use for the following activities?

	Facebook	Instagram	Twitter	LinkedIn	WhatsApp
News	<input type="checkbox"/>				
Networking	<input type="checkbox"/>				
Entertainment	<input type="checkbox"/>				
Business	<input type="checkbox"/>				

What is your preferred mode of transportation?

Choose

← Preview mode

Published Copy responder link

How satisfied are you with your current work-life balance?

1	2	3	4	5	
Not satisfied	<input type="radio"/> Very satisfied				

How often do you engage in following activities?

	Daily	Weekly	Rarely	Never
Exercise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Read Books	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Watch TV	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Submit Clear form

Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google - [Terms of Service](#) · [Privacy Policy](#)

Spreadsheet :

CC Pr8

File Edit View Insert Format Data Tools Extensions Help

L24

	A	B	C	D	E	F	G	H
1	Roll No	Name	Marks_1	Marks_2	Marks_3	Total	Average	
2	1	athulya	85	90	88	263	87.66666667	
3	2	siddhi	90	84	80	254	84.66666667	
4	3	sidhaarth	88	82	75	245	81.66666667	
5	4	abel	90	94	69	253	84.33333333	
6	5	ashwin	89	65	48	202	67.33333333	
7	6	abc	84	71	52	207	69	
8	7	pqr	78	61	49	188	62.66666667	
9	8	xyz	65	86	78	229	76.33333333	
10	9	mno	69	72	62	203	67.66666667	
11	10	def	72	80	82	234	78	
12								

KERALEEYA SAMAJAM(REGD.) DOMBIVLI'S
MODEL COLLEGE
EMPOWERED AUTONOMOUS

BSCS602

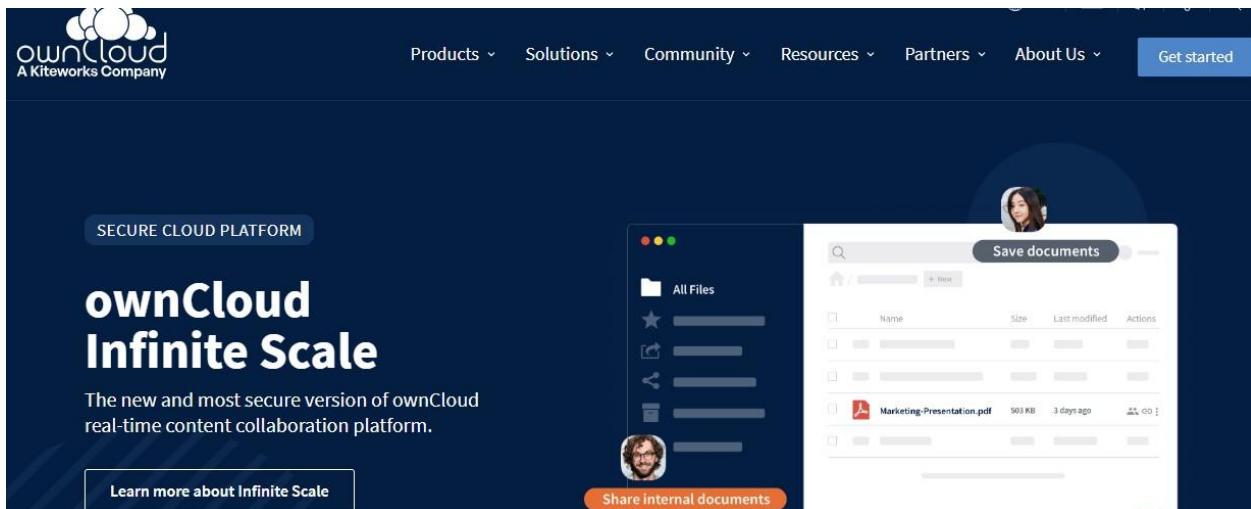
Pratik Patil

Roll No: 58

Practical 9

Study and implementation of Identity management

Step 1: Open owncloud



Step 2: Select the plan ,here we select the 1TB Free plan

Plan	Storage	Price	Contract Type	Support Options
1 TB NEXTCLOUD PRO (C)	1 TB Storage space	€6.99 per month	GDPR PPD-Contract (german)	GDPR PPD-Contract (german), Phone / Email Support, eigener Domainname
3 TB NEXTCLOUD PREMIUM (C)	3 TB Speicherplatz	starts at €99.90 per month	GDPR PPD-Contract (german)	GDPR PPD-Contract (german), Phone / Email Support, eigener Domainname
5 TB NEXTCLOUD BUSINESS (C)	Optional, up to 100 TB storage	starts at €149.90 per month	GDPR PPD-Contract (german)	GDPR PPD-Contract (german), Phone / Email Support, priority support, Talk High Performance Backend, eigener Domainname, optional Meetingserver

1 TB NEXTCLOUD PRO (C)
starts at **€6.99 per month**
GDPR PPD-Contract (german)
1 TB Storage space
Storage: Cluster
Nextcloud 30.0.0
full version
connect with https / (incl. ssl Cert)
Collabora/onlyOffice CE [info]
Email Gateway [info]
daily Backup [info]
Auto Upgrade [info]
5 User included
additional user starts at ~50/mo.
Phone / Email Support
contract term only one month

[Info / Demo](#)
[Select](#)

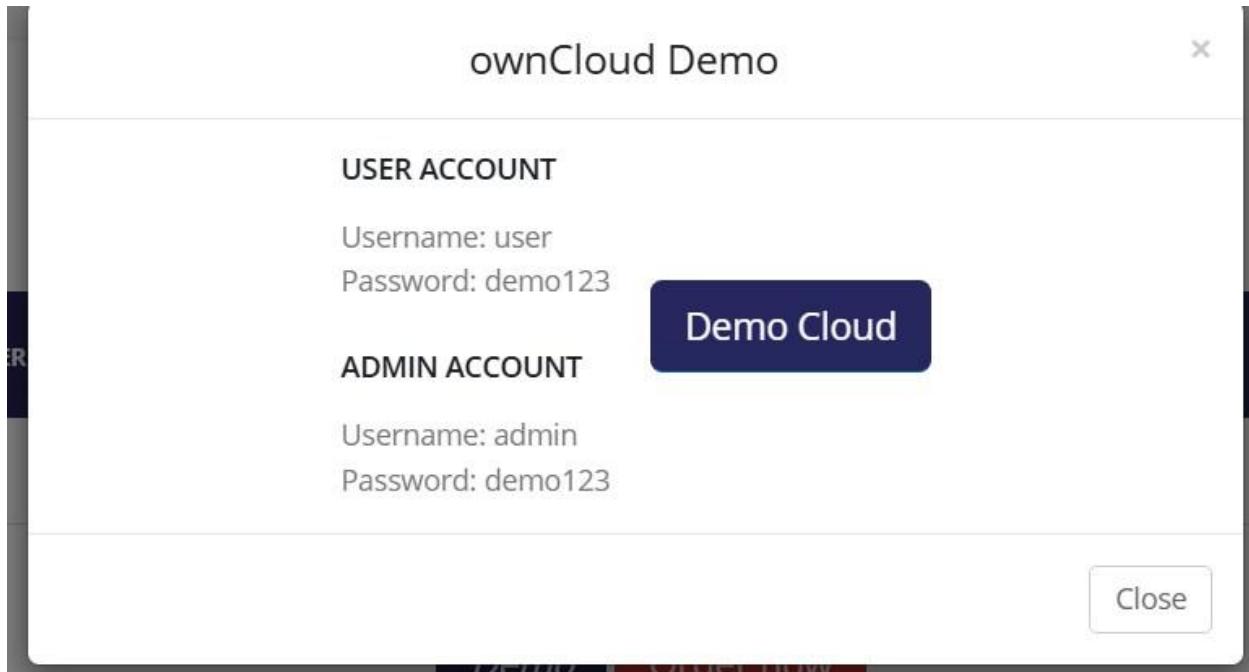
3 TB NEXTCLOUD PREMIUM (C)
starts at **€99.90 per month**
GDPR PPD-Contract (german)
3 TB Speicherplatz
Storage: Cluster
Nextcloud 30.0.0
full version
connect with https / (incl. ssl Cert)
Collabora/onlyOffice CE [info]
Email Gateway [info]
daily Backup [info]
Auto Upgrade [info]
10 Benutzer
additional user starts at ~50/mo.
Phone / Email Support
eigener Domainname
contract term only one month

[Info / Demo](#)
[Select](#)

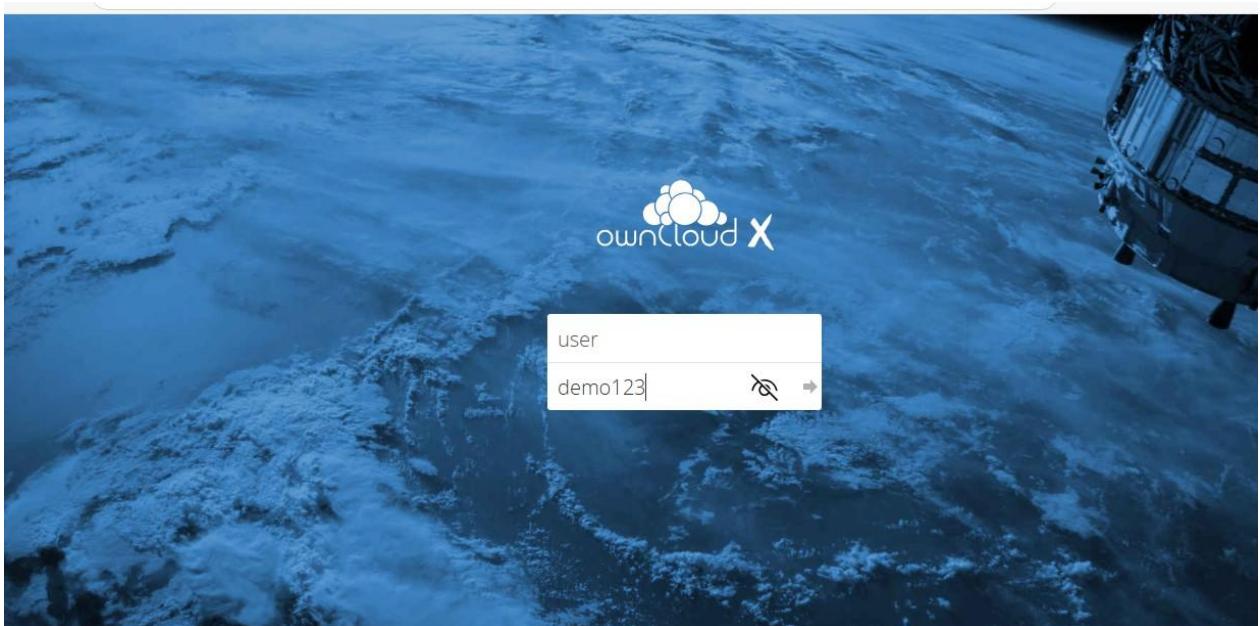
5 TB NEXTCLOUD BUSINESS (C)
starts at **€149.90 per month**
GDPR PPD-Contract (german)
5 TB Speicherplatz
Optional, up to 100 TB storage
Nextcloud 30.0.0
full version
connect with https / (incl. ssl Cert)
Collabora/onlyOffice CE [info]
Email Gateway [info]
daily Backup [info]
Auto Upgrade [info]
1 User included
additional user starts at 1,-/mo.
Phone / Email Support
priority support
Talk High Performance Backend
eigener Domainname
optional Meetingserver
contract term only one month

[Info / Demo](#)
[Select](#)

Step 3: Select The Info/Demo Option



Step 4: Using The The Username and Password Provided login to the Owncloud user and Admin account



Step 5: Hence You are Logged into the user account

The screenshot shows the ownCloud web interface. On the left, there's a sidebar with a navigation menu: 'All Files' (selected), 'Favorites', 'Shared with you', 'Shared by you', 'Shared via a link', 'In the day', and 'Deleted files'. The main area displays a list of files and folders:

Name	Size	Changed
Documents	45 KB	3 minutes ago
Photos	663 KB	6 years ago
ownCloud Manual.Pdf	4.7 MB	6 years ago
test.Odt	8 KB	6 years ago

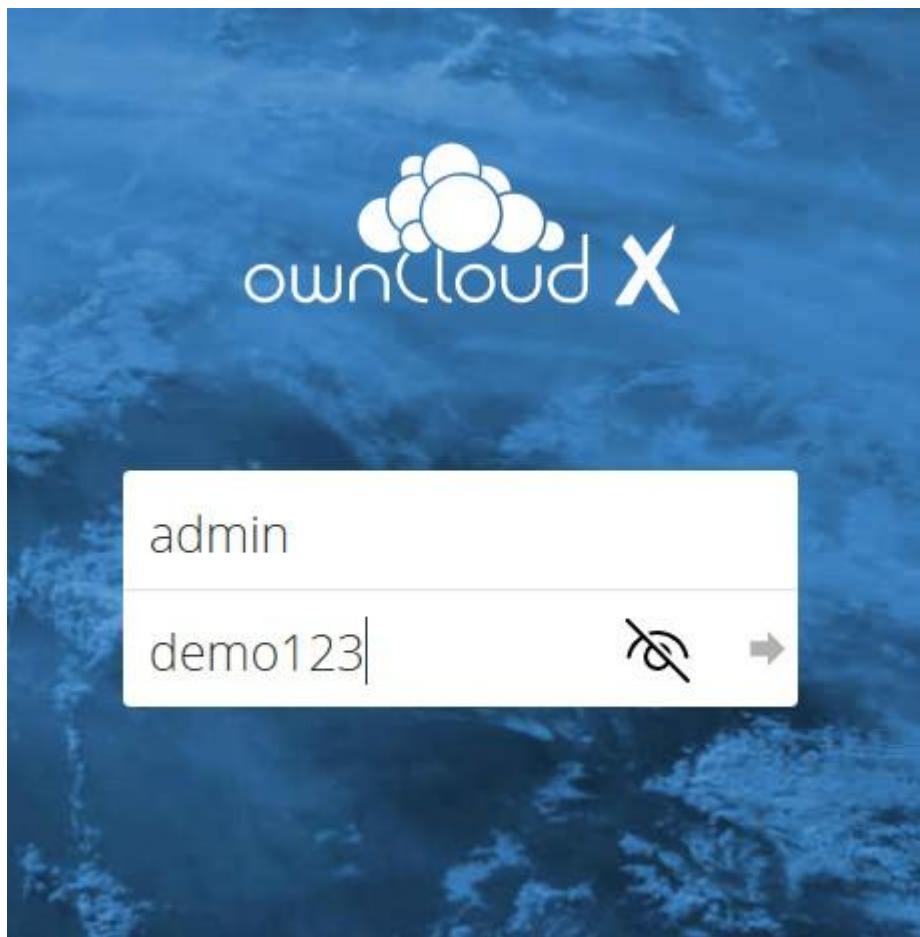
Below the list, it says '2 folders and 2 files' and '5.4 MB'.

Step 6: Now try to upload any file

The screenshot shows the ownCloud interface with the 'Upload' modal open. The modal contains a list of file types with their corresponding icons:

- Upload (with an upward arrow icon)
- Folder (with a folder icon)
- Document (with a document icon)
- Table (with a table icon)
- Presentation (with a presentation icon)
- Document (with a document icon)
- Spreadsheet (with a spreadsheet icon)
- Presentation (with a presentation icon)

Step 7: Now log out of the user account and log into the admin account



Step 8: Here you are logged into the admin account whi

A screenshot of the ownCloud dashboard. At the top, there is a dark header bar with the ownCloud logo on the left and a search bar and "Admin" dropdown on the right. Below the header, there is a navigation bar with a back arrow, a plus sign for adding new files, and other navigation icons. The main area is a file list table with columns for Name, Size, and Changed. The table contains four items: "Documents" (35 KB, 6 years ago), "Photos" (663 KB, 6 years ago), "ownCloud Manual.Pdf" (4.7 MB, 6 years ago), and "test.Odt" (8 KB, 6 years ago). Each item has a preview thumbnail, a share icon, and a three-dot menu icon.

Date:-

Practical 10**Aim : Study Cloud Security management.**

Step 1: Search AWS in Google and open the following page. Proceed to click on “Create AWS Account”.



Step 2: Select the option of “Sign in to an existing AWS account”.



Step 3: Select the option of “Sign in using root user email”.

The screenshot shows the 'IAM user sign in' page. It has fields for 'Account ID (12 digits) or account alias', 'IAM username', and 'Password'. There are 'Show Password' and 'Having trouble?' links. Below these is an orange 'Sign in' button. Further down are links for 'Create a new AWS account' and 'Remember this account'. At the bottom, a red box highlights the 'Sign in using root user email' link.

Step 4: Provide the email id in the given field and click on Next.

Sign in

The screenshot shows the AWS sign-in page for creating a new user. It has two radio button options: 'Root user' (selected) and 'IAM user'. Below these is a 'Root user email address' input field containing 'sidb.modelcollege@gmail.com'. A large blue 'Next' button is at the bottom. Below the 'Next' button is a note about AWS Customer Agreement and Privacy Notice, followed by links for 'New to AWS?' and 'Create a new AWS account'.

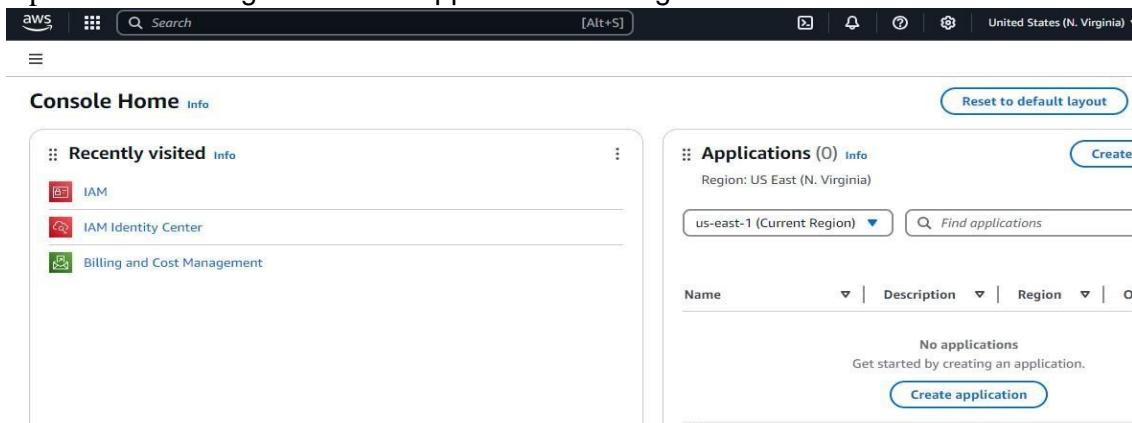
Step 5: Enter the password of the root user and sign in to the account.

Root user sign in ⓘ

Email: sidb.modelcollege@gmail.com

The screenshot shows the AWS sign-in page for the root user. It has a 'Password' input field containing '*****' and a 'Forgot password?' link. A large blue 'Sign in' button is at the bottom. Below the 'Sign in' button are links for 'Sign in to a different account' and 'Create a new AWS account'.

Step 6: The following window will appear after the login to the account is successful.



Step 7: Search for “IAM” in the search bar and select the first option.

Step 8: The following dashboard will appear from this select the option of “Users”.

Step 9: Click on “Create User” to create a new user into the account.

Step 10: Provide the name of the user and then tick the checkbox. Following dropdown will appear from that select “I want to create an IAM user.”

Specify user details

User details

User name: Batch1

Provide user access to the AWS Management Console - optional (checked)

I want to create an IAM user (selected)

Step 11: Click on “Custom password” and enter the password you want to provide then tick the checkbox seen below and proceed to click on next.

Console password

Autogenerated password (radio button)

Custom password (radio button selected)

Show password (checkbox checked)

If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user.

Console password

Autogenerated password (radio button)

Custom password (radio button selected)

Show password (checkbox checked)

If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user.

Step 12: The following window will appear. Check every detail and click on “Create User”.

Permissions options

- Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
- Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.
- Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

User groups (2)

Group name	Users	Attached policies	Created
group1	1	AdministratorAccess	2025-01-31 (3 days ago)
group2	1	AdministratorAccess	2025-01-31 (3 days ago)

Set permissions boundary - optional

Step 4: Review and create

Permissions summary

Name	Type	Used as
IAMUserChangePassword	AWS managed	Permissions policy

Tags - optional
Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.
No tags associated with the resource.

Create user

Step 13: Here we can see that the user has been successfully created. Download the csv file of the credentials which is required for the further use.

User created successfully
You can view and download the user's password and email instructions for signing in to the AWS Management Console.

Console sign-in details

Console sign-in URL: <https://588738572450.signin.aws.amazon.com/console>

User name: Batch1

Console password: ***** Show

Download .csv file **Return to users list**

CloudShell **Feedback** © 2025, Amazon Web Services, Inc. or its affiliates. **Privacy** **Terms** **Cookie preferences**

Step 14: Here we can see the user we created just now.

The screenshot shows the AWS IAM 'Users' page. At the top, there is a search bar and a 'Create user' button. Below the header, a table lists three users:

User name	Path	Groups	Last activity	MFA	Password age	Console
sid1	/	1	3 days ago	Virtual	3 days	January
Siddy	/	0	-	-	1 minute	-
user2	/	1	3 days ago	Virtual	2 days	January

Step 15: Click on “User groups” and select the option on “Create group”.

The screenshot shows the AWS IAM 'User groups' page. On the left, there is a navigation menu with 'User groups' selected. The main area displays a table of existing user groups:

Group name	Users	Permissions	Creation time
group1	1	Defined	3 days ago
group2	1	Defined	3 days ago

Step 16: Give the group name in the respective field and select the users to add into the group.

The screenshot shows the 'Create user group' page. On the left, there is a navigation menu with 'User groups' selected. The main area has two sections:

- Name the group**: A field labeled 'User group name' with the value 'B1@CC'.
- Add users to the group - Optional (1/6)**: A table showing users available to be added:

User name	Groups	Last activity	Creation time
Batch1	0	25 minutes ago	28 minutes ago
sid1	1	17 days ago	17 days ago

Step 17: Here Attach the permission policies, select the first option as shown below. Click on create group.

The screenshot shows the AWS IAM 'Create user group' interface. In the center, a modal window titled 'Attach permissions policies - Optional (1/1032)' is open. It displays a list of AWS managed policies. The 'AdministratorAccess' policy is selected and highlighted with a blue border. Other policies listed include 'AdministratorAccess', 'AdministratorAccess', 'AIOpsAssistantPolicy', 'AIOpsConsoleAdmin...', and 'AIOpsOperatorAcc...'. At the bottom of the modal, there is descriptive text: 'Provides full access to AWS services an...'.

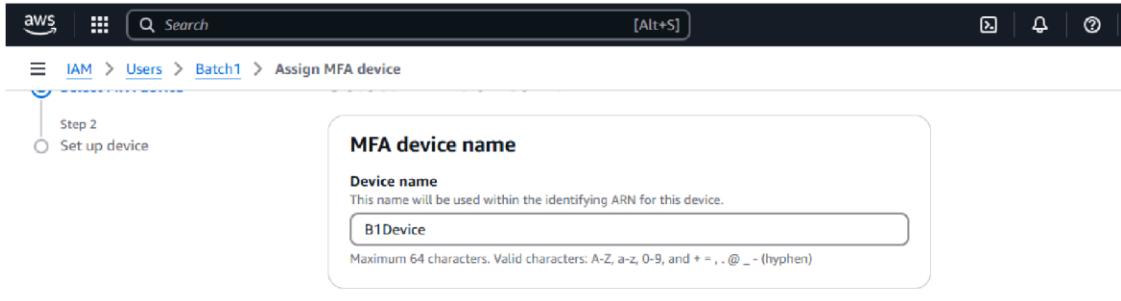
Step 18: Here the group is successfully created as shown below.

The screenshot shows the AWS IAM 'User groups' page. A green success message at the top left states 'B1@CC user group created.' Below this, the 'User groups (5)' section is displayed. A table lists five groups: 'B1@CC', 'group1', 'group2', 'group3', and 'group5'. Each group entry includes columns for 'Group name', 'Users', 'Permissions', and 'Creation time'. The 'B1@CC' group was created 'Now'.

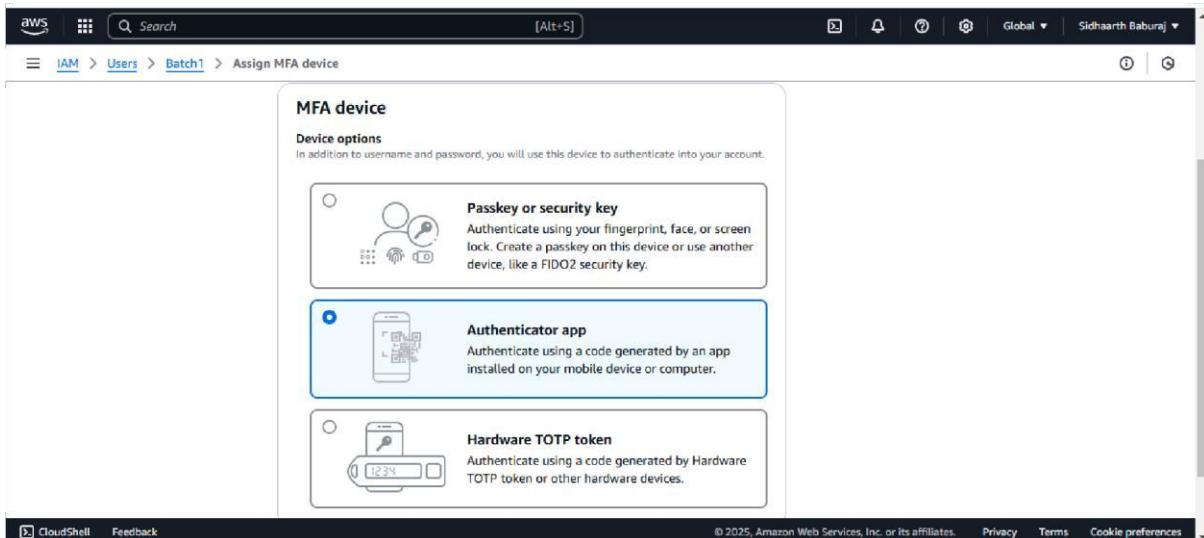
Step 19: Go to the Users and click on the link of the user just created, The following window will appear in that click on the option of “Enable without MFA” and select the option of “Enable MFA”.

The screenshot shows the AWS IAM 'Users' page. A user named 'Batch1' is selected. The 'Summary' section shows the ARN as 'arn:aws:iam::588738572450:user/Batch1'. Under 'Console access', the status is 'Enabled without MFA'. There is a callout bubble pointing to the 'Enable MFA' button. Other tabs in the summary include 'Delete', 'Groups', 'Tags', 'Security credentials', and 'Last Accessed'. The 'Permissions' tab is active, showing the 'Permissions policies' section with a search bar and filter options.

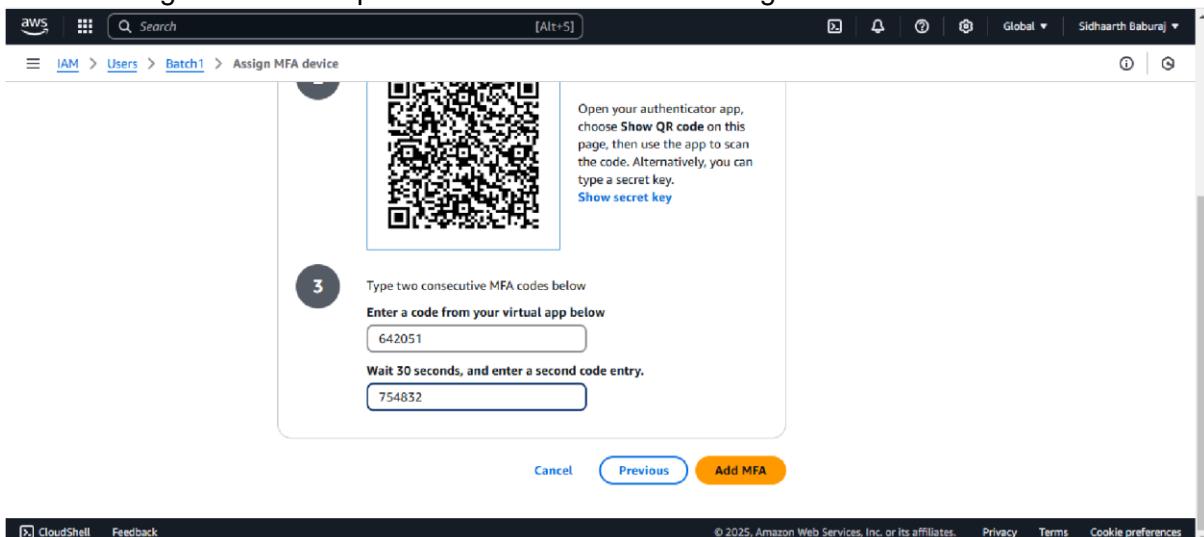
Step 20: Give a name to the MFA device.



Step 21: Select the option of “Authenticator app”.



Step 22: Click on Show QR Code to get the code to scan. Scan the QR Code and wait till the MFA code is generated and provide the code in the following fields. Click on Add MFA.



Step 23: Here the MFA device is successfully assigned.

The screenshot shows the AWS IAM User Details page for a user named 'Batch1'. A prominent green banner at the top states 'MFA device assigned' with a note: 'You can register up to 8 MFA devices of any combination of the currently supported MFA types with your AWS account root and IAM user. With multiple MFA devices, you only need one MFA device to sign in to the AWS console or create a session through the AWS CLI with that user.' Below the banner, the 'Security credentials' tab is selected, showing details like ARN (arn:aws:iam::588738572450:user/Batch1), Console access (Enabled with MFA), and Last console sign-in (Today). Other tabs include 'Permissions', 'Groups', 'Tags', and 'Last Accessed'. The 'Console sign-in' section displays a 'Console sign-in link' (https://588738572450.signin.aws.amazon.com/console) and a 'Console password' (Updated 30 minutes ago). The 'Last console sign-in' was 31 minutes ago.

Step 24: Go back to the login page and enter the details required. The details can be obtained from the credential.csv file . Click on Sign-in.

The screenshot shows the 'IAM user sign in' page. It requires the following inputs:

- Account ID (12 digits) or account alias: 588738572450
- IAM username: Batch1
- Password: (redacted)
- Show Password:
- Having trouble?: [Having trouble?](#)
- Sign in button
- Sign in using root user email link
- Create a new AWS account link

Step 25: Enter the MFA code and click on Sign-in.

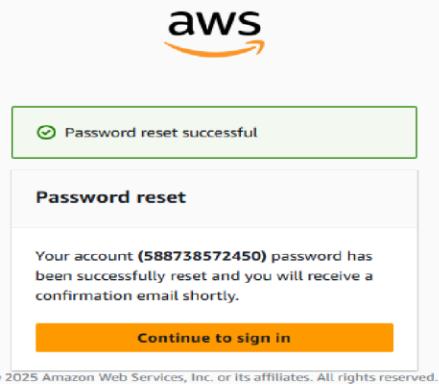
This is a step in the multi-factor authentication process:

- Keeping you secure**: Your account is protected with **multi-factor authentication (MFA)**. To finish signing in, enter the code from your MFA device below.
- MFA code**: An input field containing 'enter code'.
- Sign in** button.
- Sign in to a different account** link.
- Trouble signing in?** link.

Step 26: Here we have to create a new password for that enter the old password first and then the new one. Click on Confirm Password Change.

The screenshot shows the AWS Password Reset page. It displays a message: "Your account (588738572450) password has expired or requires a reset. To continue, please verify your old and set a new password for Batch1 (not you?).". Below this, there are fields for "Old Password" (Batch1@201), "New Password" (Batch1@001), and "Confirm New Password" (Batch1@001). There are "Show Password" checkboxes next to each input field. A green "Matches" label is positioned between the "New Password" and "Confirm New Password" fields. At the bottom is a large orange "Confirm Password Change" button.

Step 27: Here we can see that the password has been reset successfully. Click on continue to sign-in.



Step 28: Here we are signed in into the Console home again as an IAM User.

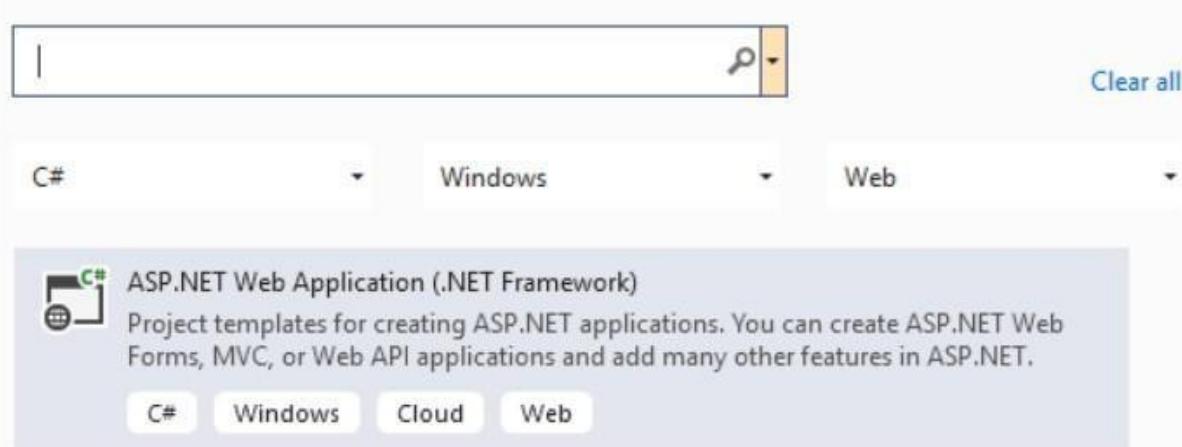
The screenshot shows the AWS Console Home page. The top navigation bar includes the AWS logo, a search bar, and the URL "Batch1 @ 5887-3857-2450". The main content area has a sidebar titled "Recently visited" with links to "IAM", "Billing and Cost Management", and "IAM identity Center". The main panel is titled "Applications (0)" and shows a message: "No applications. Get started by creating an application." It includes a "Create application" button. At the bottom of the page are links for "View all services", "Go to myApplications", and standard footer links like "CloudShell", "Feedback", "Privacy", "Terms", and "Cookie preferences".

Date:-

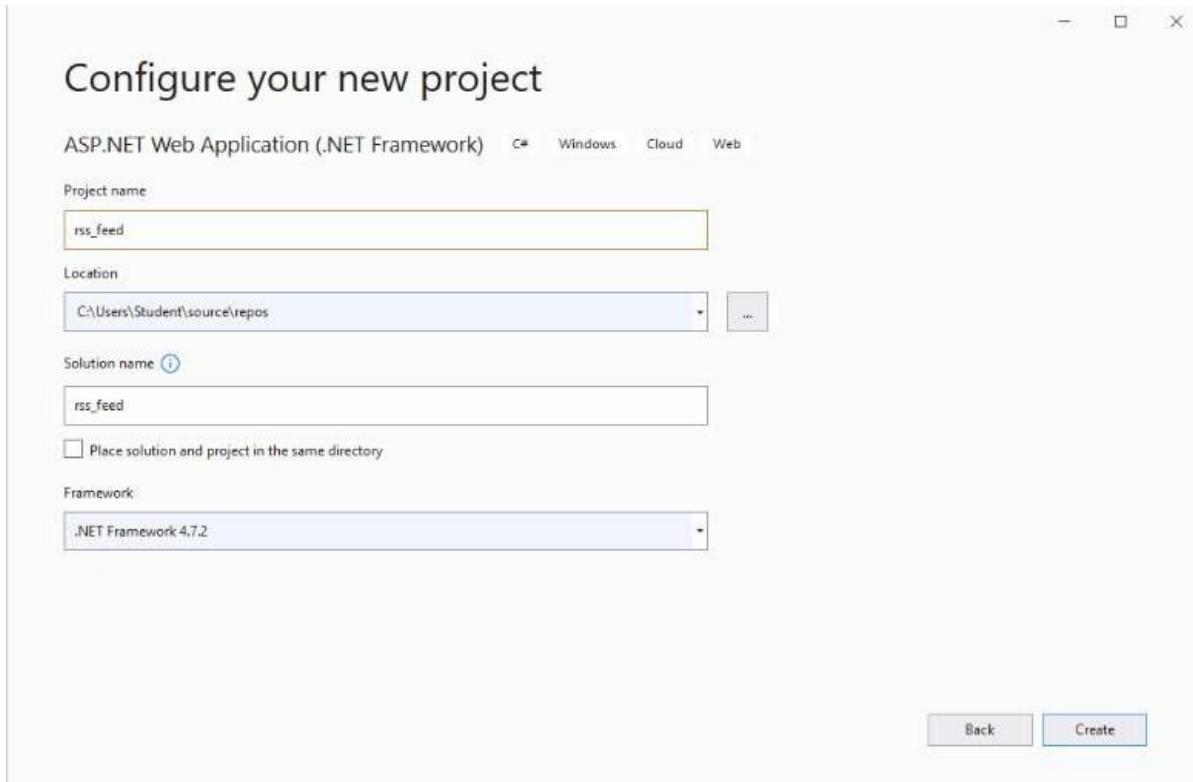
Practical 11

Aim : Write a program for the web feed.

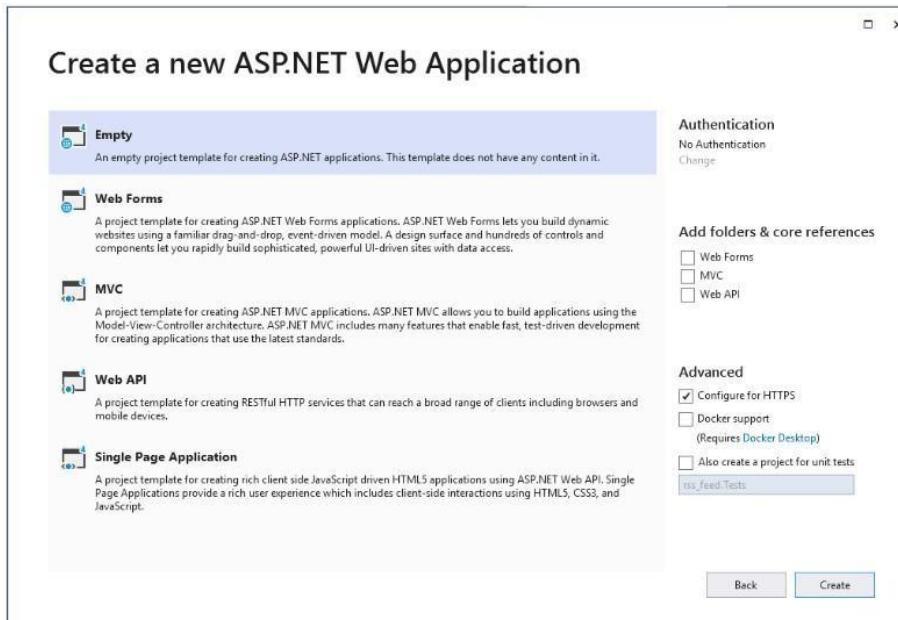
Step 1: Start Visual Studio → Create a new project → All Languages → C# → All Platform → Windows → All project types → Web → Select ASP.NET Web App.



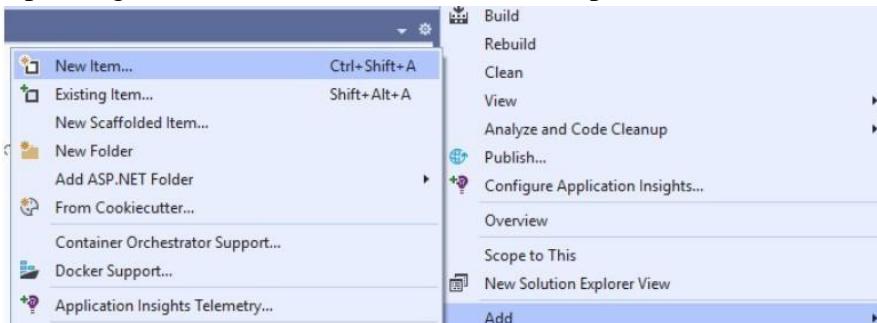
Step 2: Give name for the project as ‘RSSFeed’ → Click on Create.



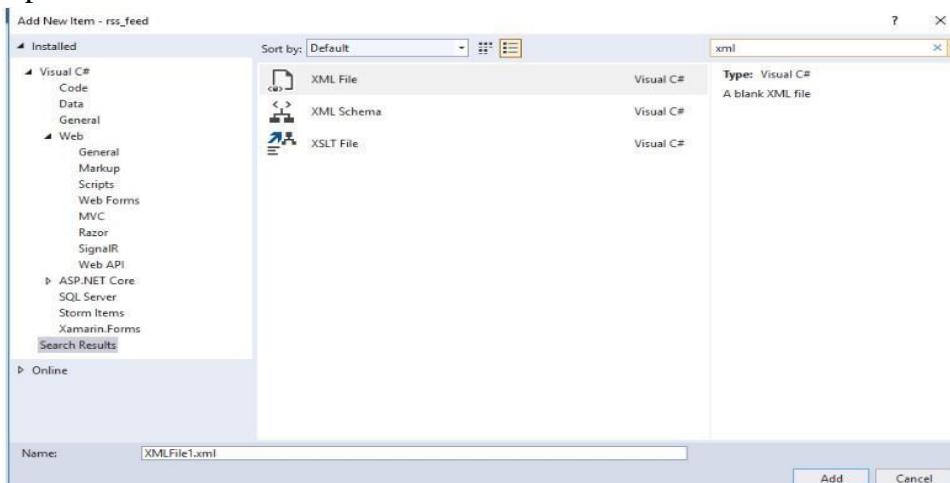
Step 3: Select ‘Empty’ → click on create.



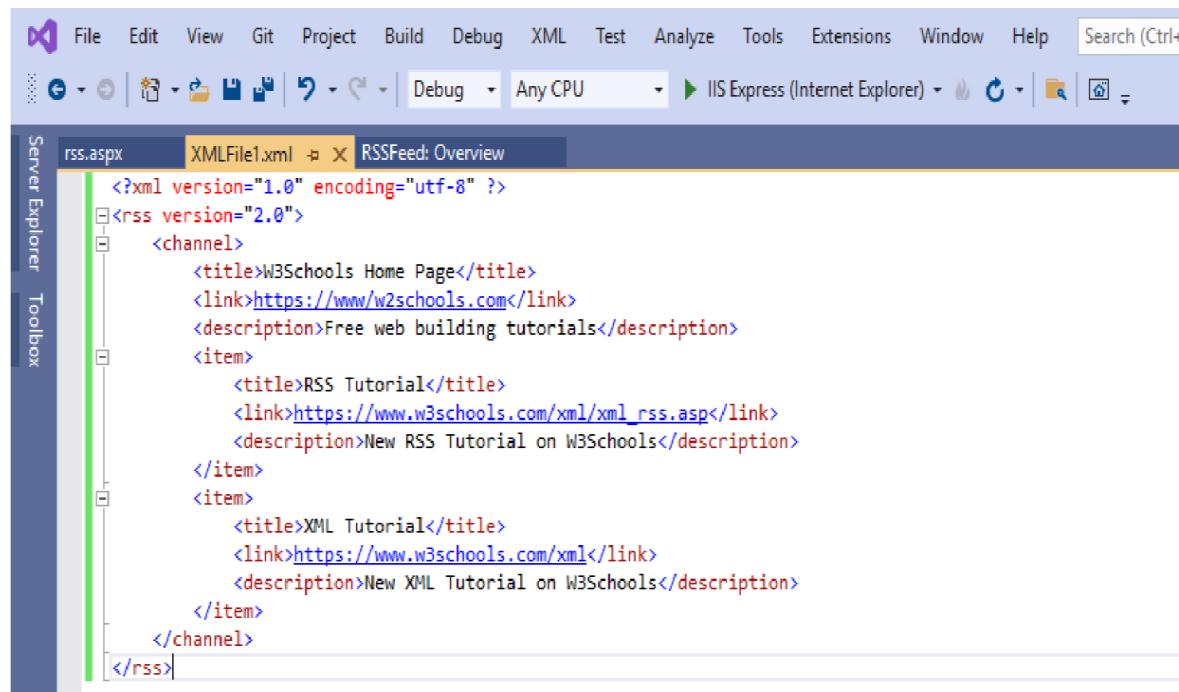
Step 4: Right click on ‘RSSFeed’ in solution explorer → New Item → Add.



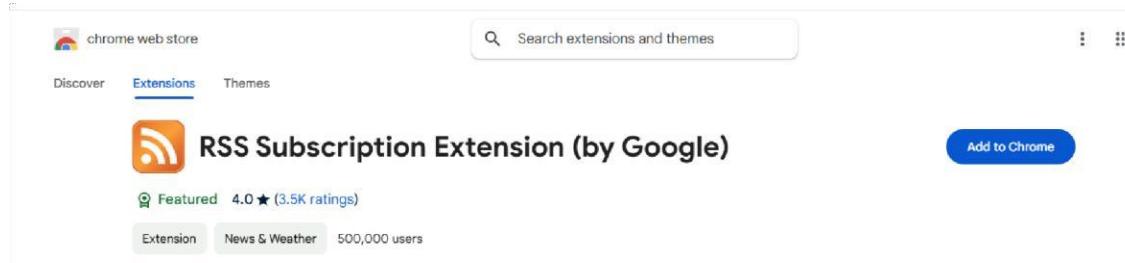
Step 5: Now search for ‘XML’ → select ‘XML File’.



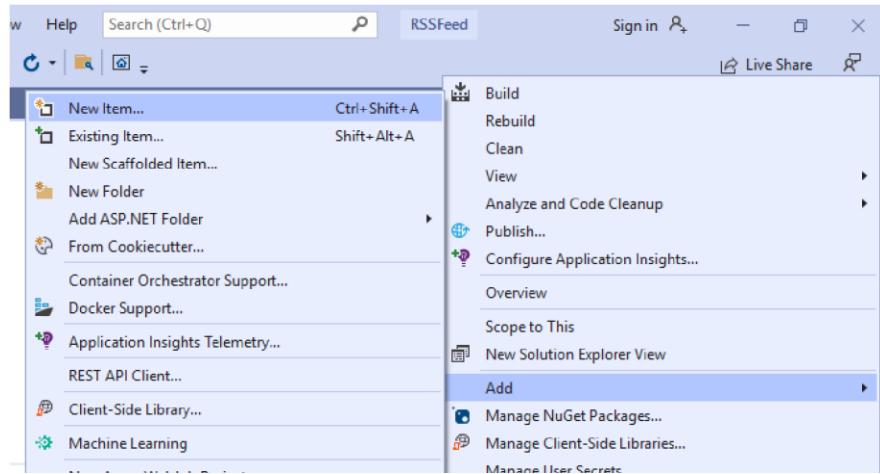
```
//XMLFile.xml
<?xml version="1.0" encoding="utf-8" ?>
<rss version="2.0">
    <channel>
        <title>W3Schools Home Page</title>
        <link>https://www.w3schools.com</link>
        <description>Free web building tutorials</description>
        <item>
            <title>RSS Tutorial</title>
            <link>https://www.w3schools.com/xml/xml_rss.asp</link>
            <description>New RSS Tutorial on W3Schools</description>
        </item>
        <item>
            <title>XML Tutorial</title>
            <link>https://www.w3schools.com/xml</link>
            <description>New XML Tutorial on W3Schools</description>
        </item>
    </channel>
</rss>
```



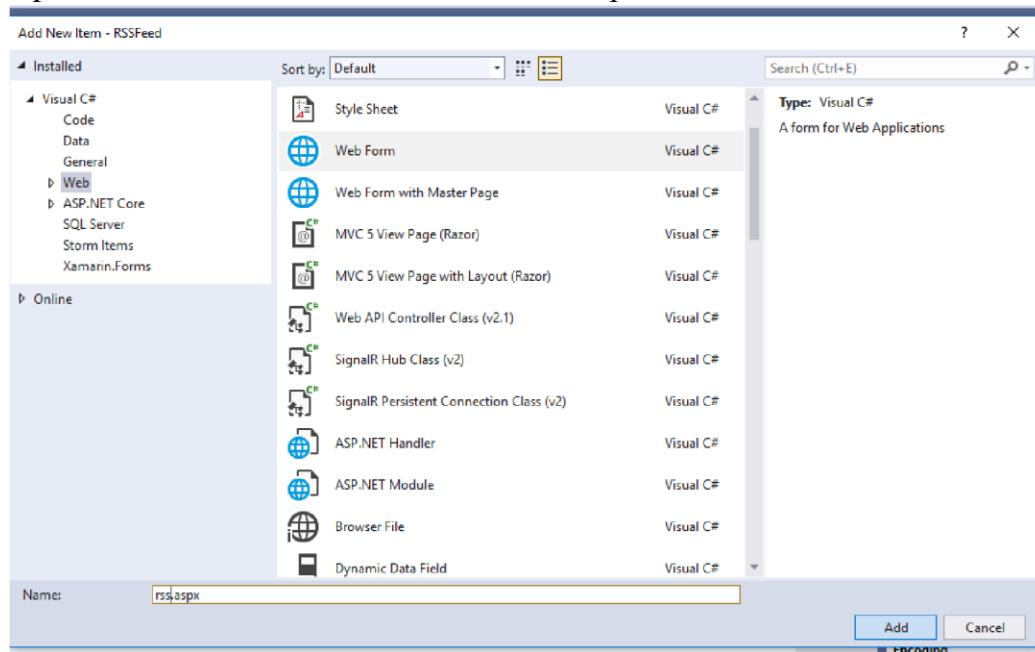
Step 6: Search for RSS Subscription Extension in google



Step 7: Right click on RSSFeed → New Item → Add.



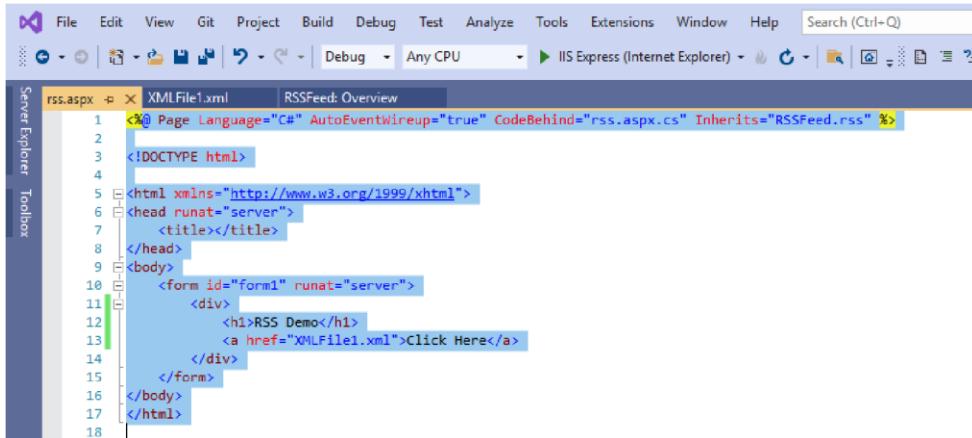
Step 8: Select ‘Web Form’ and name it as ‘rss.aspx’ and click on ‘Add’.



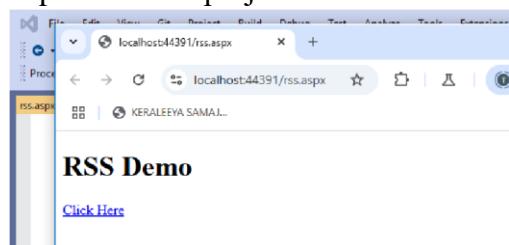
```
//rss.aspx
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="rss.aspx.cs"
Inherits="RSSFeed.rss" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <h1>RSS Demo</h1>
            <a href="XMLFile1.xml">Click Here</a>
        </div>
    </form>
</body>
</html>
```



Step 9: Run the project → Click on ‘Click Here’



The following page is displayed.

A screenshot of a web browser window displaying an RSS feed. The title of the feed is "Feed for W3Schools Home Page". It includes a subscribe button for "Newsblur" and a "Subscribe Now" button. There is also a checkbox for "Always use this reader to subscribe to feeds". Below the feed title, there are two items listed:

- RSS Tutorial**
New RSS Tutorial on W3Schools
- XML Tutorial**
New XML Tutorial on W3Schools

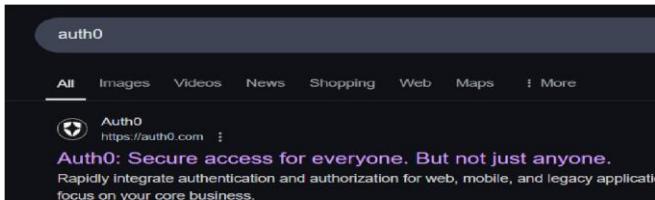
The browser interface shows tabs for "Subscribe to this feed" and "localhost:44391/rss.aspx". The address bar shows "localhost:44391/rss.aspx". The title bar of the browser says "RSS Subscript chrome-extens... Verify it's you".

Date:-

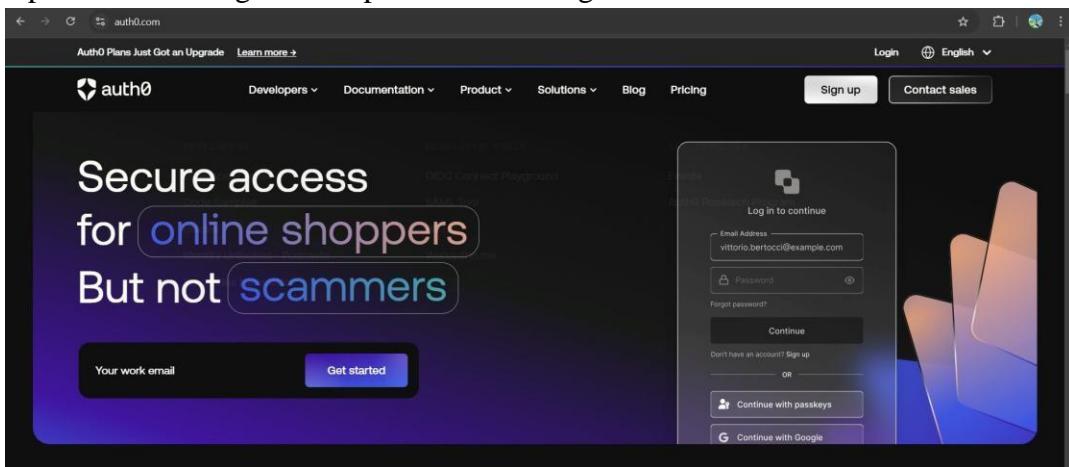
Practical 12

Aim: Study and implementation of Single-Sing-On.

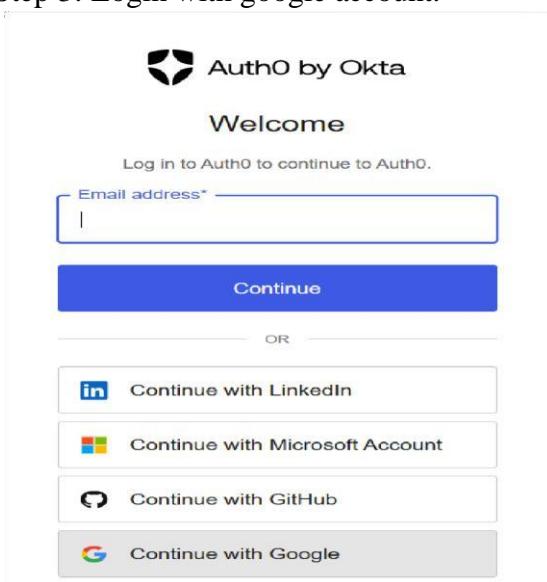
Step 1: Search auth0 in chrome.



Step 2: The AuthPage0 will open. Click on Login.



Step 3: Login with google account.



Step 4: Select account type as ‘Other’. Click next.

Account Type
Are you creating this account for yourself or on behalf of a company?



I need advanced settings

We've assigned your data region to the United States and given you a tenant name. Check this box if you need to process your data in a different region to comply with privacy laws.

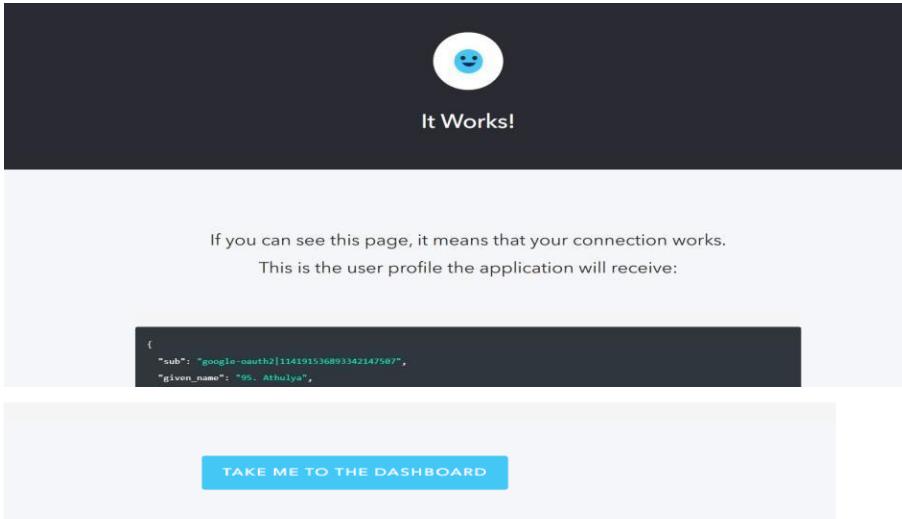
NEXT

Step 5: The dashboard will get open.

Step 6: Go to social connections. (Authentication → Social)

Step 7: Click on three dots on google- auth0.

Step 8: The following page will be displayed showing “It works”. Now click on “Take me to the dashboard”.



Step 9: Click on create connection.

Social Connections

+ Create Connection

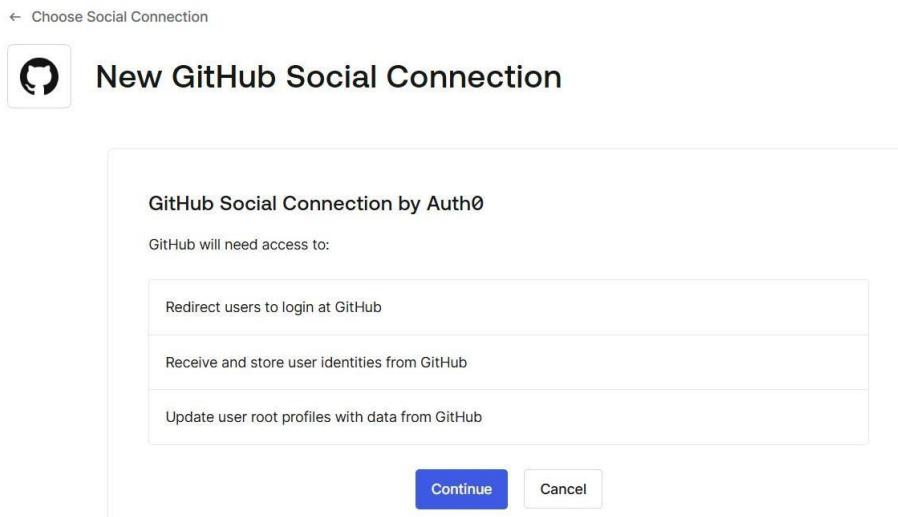
Configure social connections like Facebook, Twitter, Github and others so that you can let your users login with them. Show more >

Step 10: Select any one option from the given social connections.

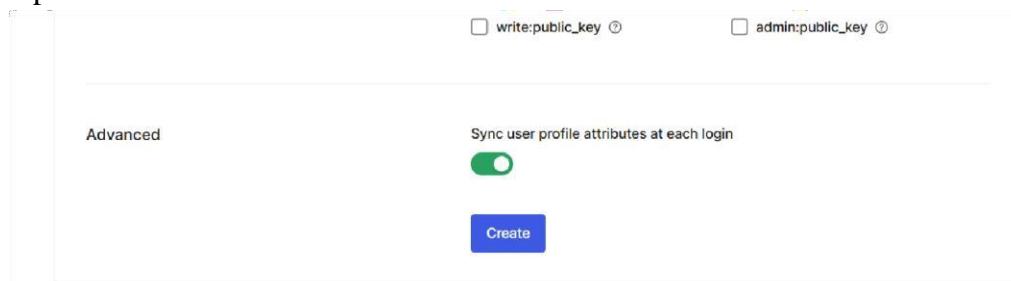
New Social Connection

Connection Type	Description	Status
Klarna	Allow your users to Sign in with their Klarna account	ADDED
Google / Gmail	Allow your users to login with their Google Account	ADDED
Facebook	A fast and convenient way for users to log into your app with Facebook	ADDED
Apple	The easy way to add Sign in with Apple to your app or website	
Microsoft	Enable your users via their trusted Microsoft Account	ADDED
LinkedIn	Leverage the largest professional social network to enhance your sign-in...	
Github	Enable the GitHub login option for your application	ADDED
Dropbox	Auth0 with Dropbox provides social login from the world's leading...	
Bitbucket	Enable the Bitbucket login option for your application	

Step 11: Click on continue.



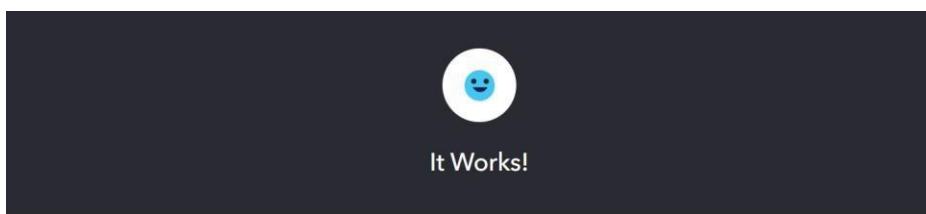
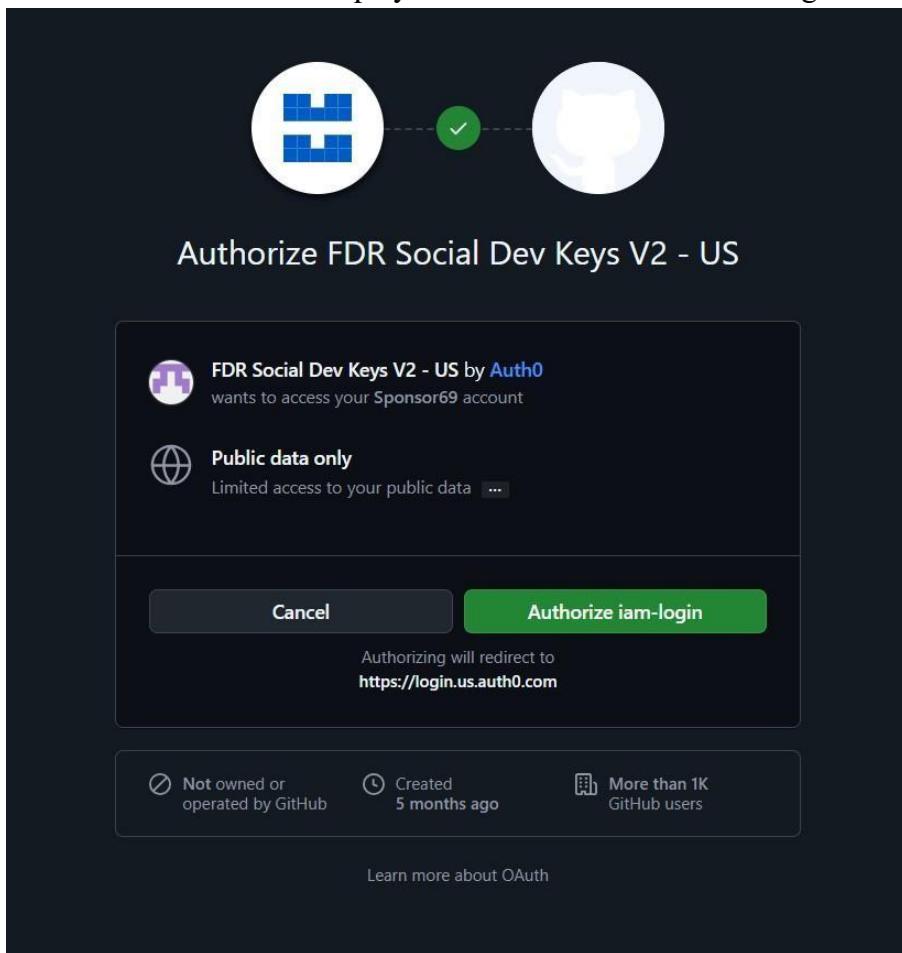
Step 12: Click on create.



Step 13: Enable the default app option and click on 'Try Connection'. Give the required credentials and login to your account.



The below screen will be displayed. Click on ‘Authorize iam-login’.



If you can see this page, it means that your connection works.

This is the user profile the application will receive:

```
{
  "sub": "github|196524837",
  "nickname": "Sponsor69",
  "name": "",
  "picture": "https://avatars.githubusercontent.com/u/196524837?v=4",
  "updated_at": "2025-02-07T15:09:59.131Z"
}
```

BSCS602

Pratik Patil

Roll No: 58

KERALEEYA SAMAJAM(REGD.) DOMBIVLI'S
MODEL COLLEGE
EMPOWERED AUTONOMOUS

Date:-

Practical 13**Aim : User Management in the Cloud**

Step 1:- Search en.ocloud.de

Select your Cloud

A little help:

Choose Nextcloud if you...

- have no experience with ownCloud
- don't want to set up or configure anything
- want to be on the safe side
- just want it to work



Nextcloud

Select



ownCloud

Select

Step 3:- Select NextCloud.

Select your Cloud

A little help:

Choose Nextcloud if you...

- have no experience with ownCloud
- don't want to set up or configure anything
- want to be on the safe side
- just want it to work



Nextcloud

Select

Nextcloud with [exclusive oCloud.de Features](#): mail2Cloud - save files via email into your cloud, onlyoffice - edit word documents online ...



ownCloud

Select

The original ownCloud has enjoyed an excellent reputation for several years and has a large team of developers.

Use your own cloud in 100 seconds - fast install

/nextcloud.html

Step 4:- Select Storage Space.

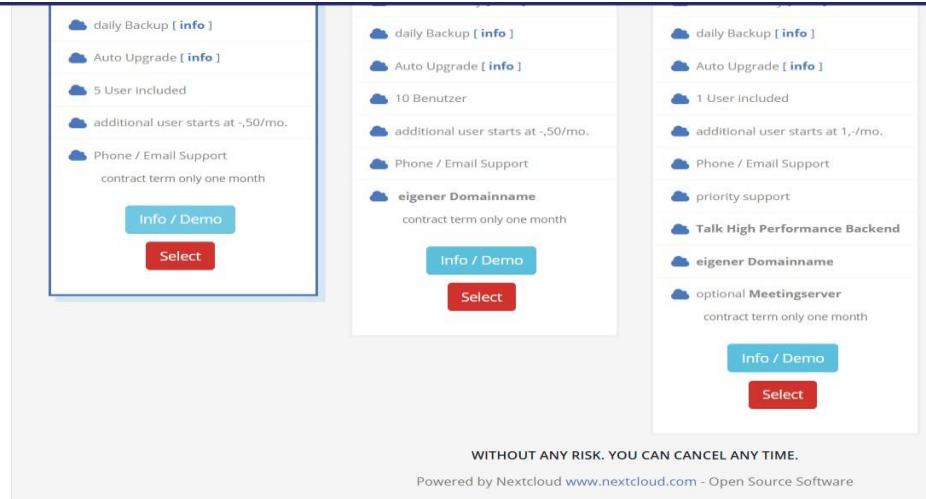
Select your storage space:

CLOUD | Storage type: Cluster
▶ DEDICATED | Storage type: All-in-One-Server (up to 1,50 €/month)

Ideal for most users. Affordable price with excellent performance.

1 TB NEXTCLOUD PRO (C)	3 TB NEXTCLOUD PREMIUM (C)	5 TB NEXTCLOUD BUSINESS (C)
starts at €6.99 per month	starts at €99.90 per month	starts at €149.90 per month
GDPR PPD-Contract (german) 1 TB Storage space Storage: Cluster Nextcloud 30.0.0	GDPR PPD-Contract (german) 3 TB Speicherplatz Storage: Cluster Nextcloud 30.0.0	GDPR PPD-Contract (german) 5 TB Speicherplatz Optionally, up to 100 TB storage Nextcloud 30.0.0

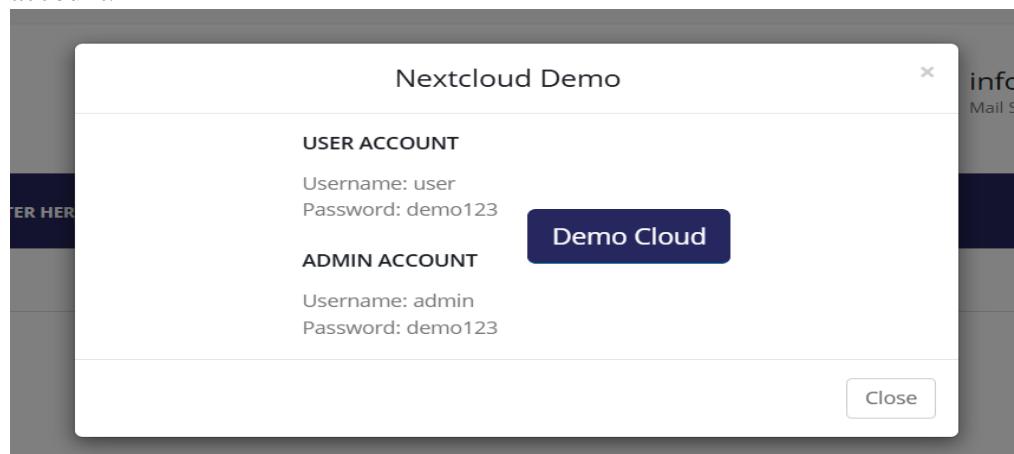
Step 5:- Select Info/Demo.



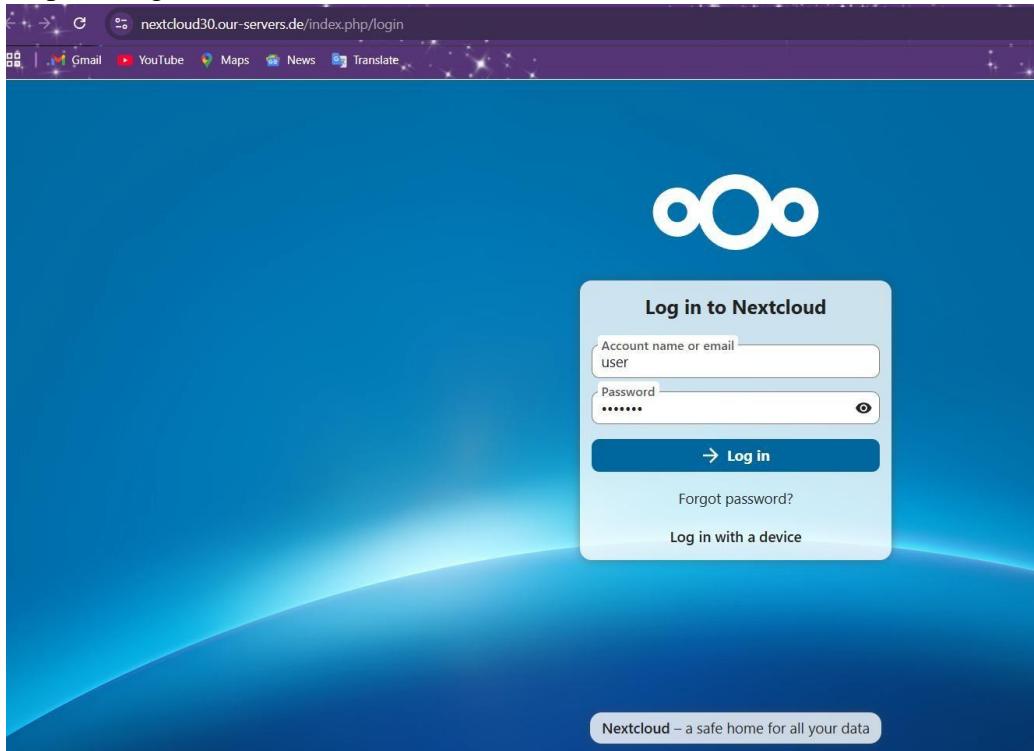
Step 6:- Click on Demo.

The screenshot shows the Nextcloud homepage with a dark blue header bar containing links for HOME, PRODUCTS, REGISTER HERE - GET YOUR PERSONAL CLOUD, DATA SECURITY, FAQ, CUSTOMER AREA, and a language dropdown. Below the header, the word 'Nextcloud' is followed by a horizontal line and two buttons: 'Demo' (highlighted in blue) and 'Order now'. A sub-section titled '... puts you in control' contains text about Nextcloud's data management features. To the right is a large image of a Nextcloud login screen with fields for 'Benutzername' and 'Passwort' over a background of mountains at dusk. A circular arrow icon with a downward arrow is positioned below the main content.

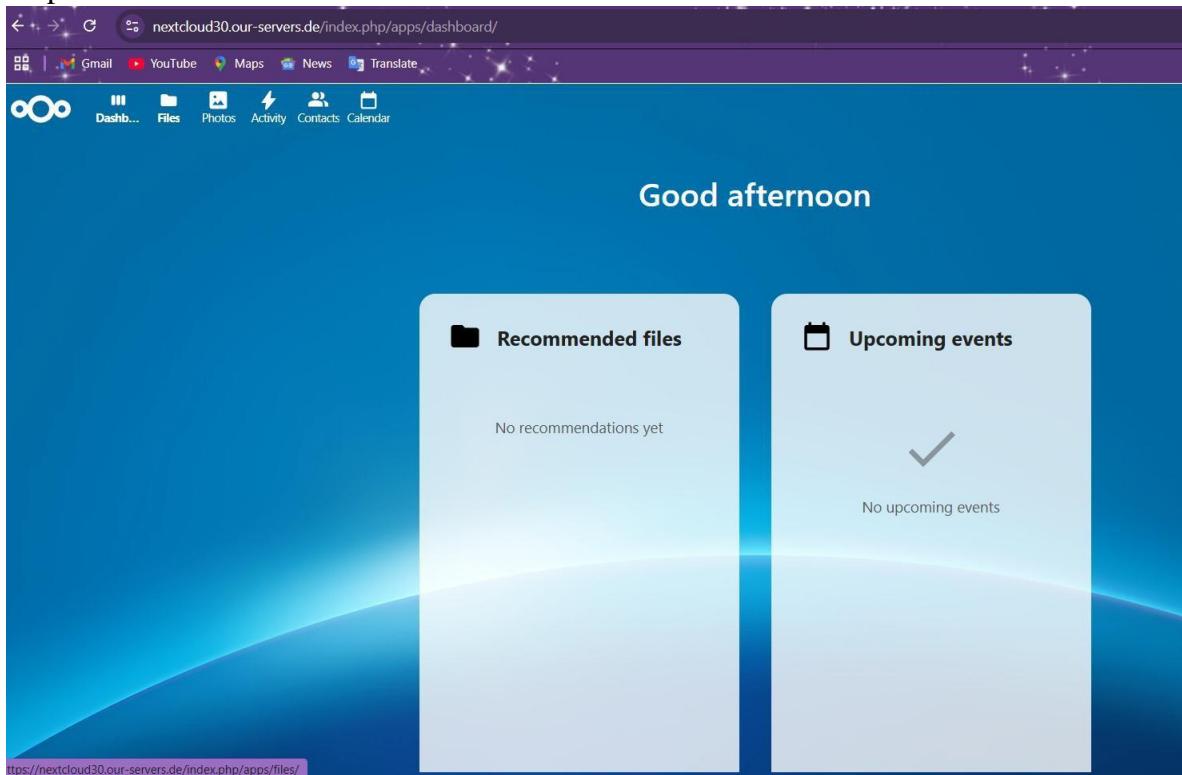
Step 7:- Note down the username and password for user and admin account.



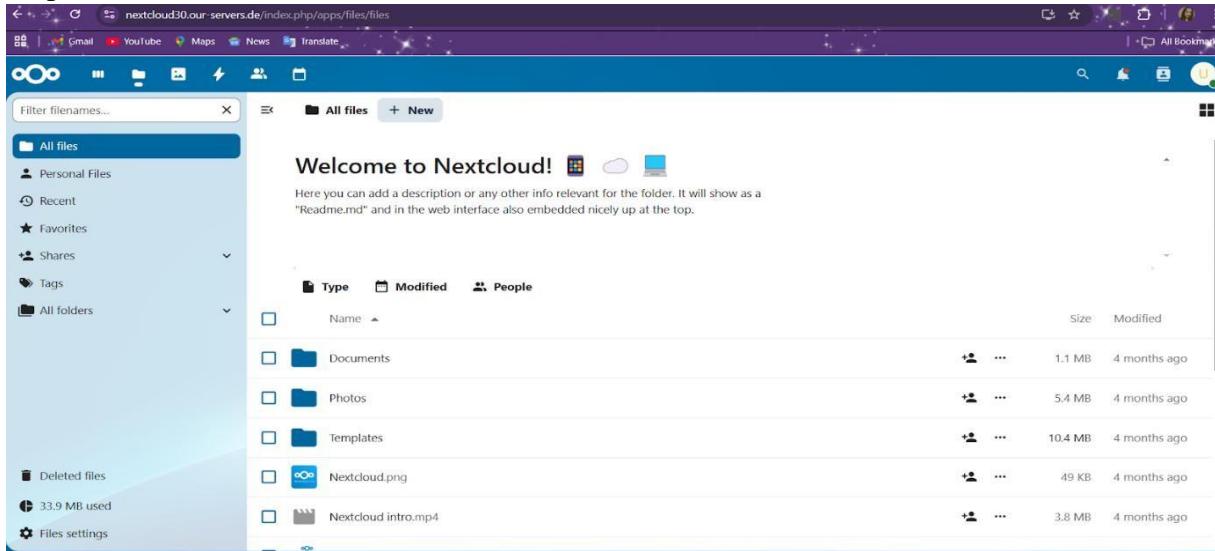
Step 8:- Login to the Nextcloud with a user account.



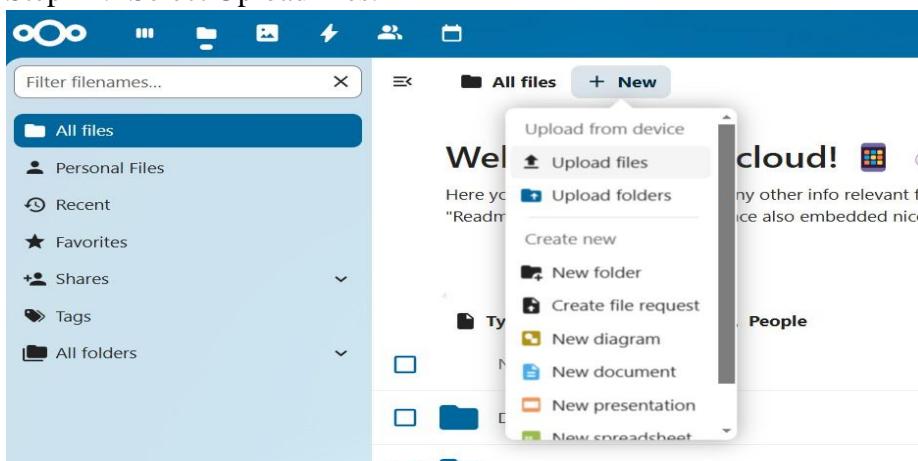
Step 9:- Click on Files.



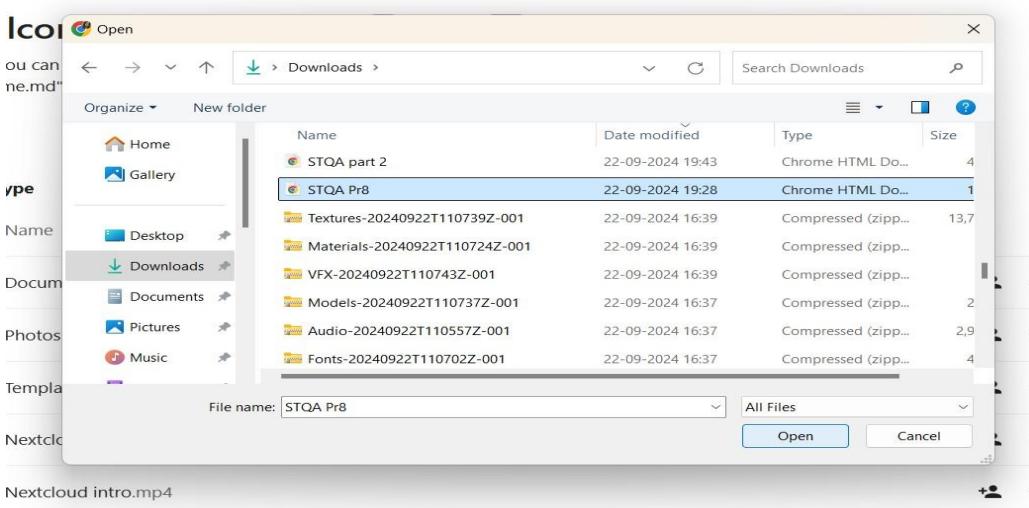
Step 10:- To add files click on New.



Step 11:- Select Upload files.



Step 12:- Browse the file you want to upload → select open.



Step 13:- You can see the file is uploaded.

The screenshot shows the Nextcloud web interface. On the left, there's a sidebar with links for Personal Files, Recent, Favorites, Shares, Tags, and All folders. The main area displays a list of files and folders. One file, 'Readme.md', is highlighted. The list includes:

Name	Type	Modified	Size	Modified
Readme.md	File	4 months ago	< 1 KB	< 1 KB
Reasons to use Nextcloud.pdf	File	4 months ago	954 KB	4 months ago
STQA Pr8.pdf	File	5 months ago	141 KB	141 KB
Templates credits.md	File	4 months ago	2 KB	2 KB

Total: 7 files and 3 folders

Step 14:- To add a folder click on New → Select Upload Folders.

The screenshot shows the Nextcloud web interface with the 'New' button highlighted. A dropdown menu appears with various options. The 'Upload folders' option is selected and highlighted.

Step 15:- Browse the desired folder → click on Upload.

The screenshot shows a file selection dialog box titled 'Select Folder to Upload'. It displays a list of folders in the 'Downloads' directory. The 'DiskMon' folder is selected and highlighted. At the bottom right, there are 'Upload' and 'Cancel' buttons.

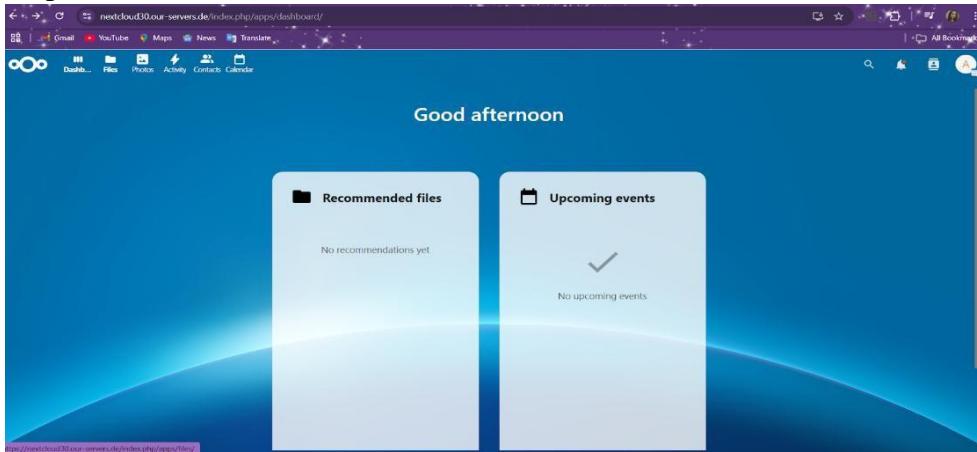
Step 16:- You can see the folder is uploaded.

Name	Type	Size	Modified
DiskMon	Folder	0 KB	a few seconds ...
Documents	Folder	1.1 MB	4 months ago
Photos	Folder	5.4 MB	4 months ago

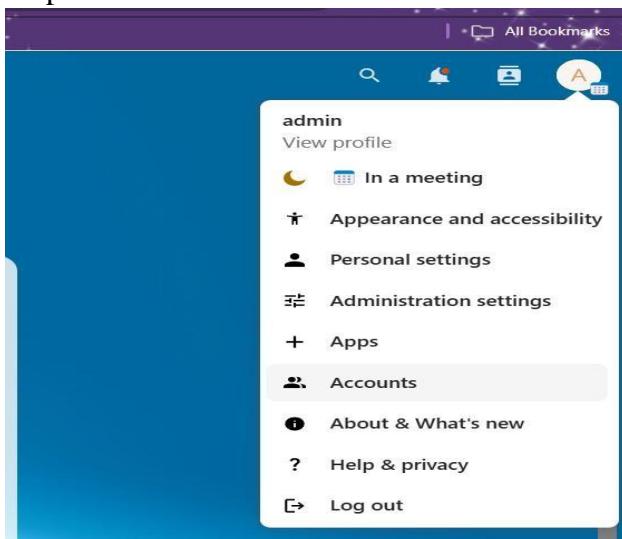
Step 17:- Logout from the account.

Step 18:- Login to the NextCloud as an admin.

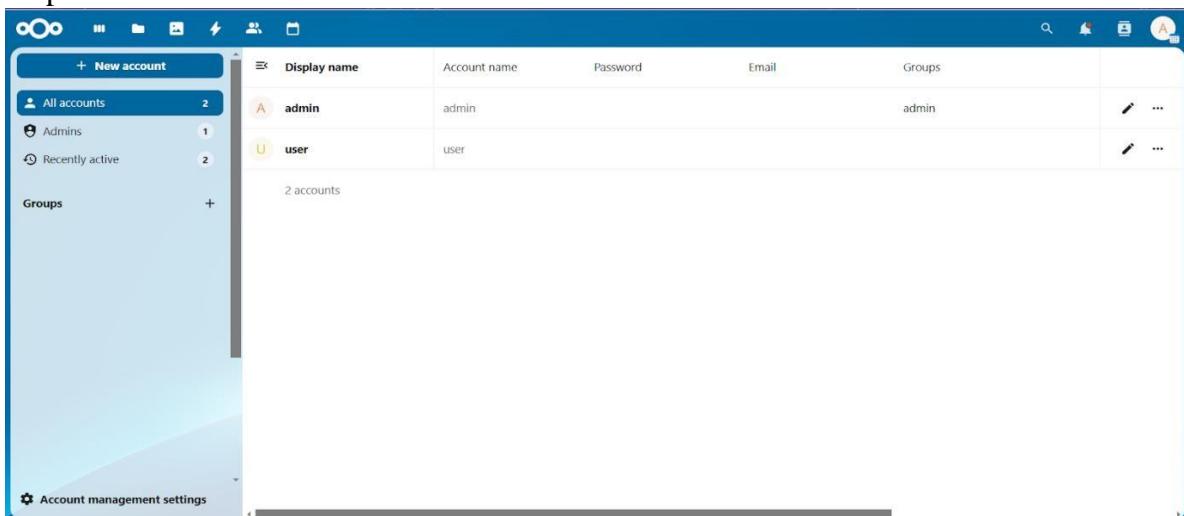
Step 19:- To create a user, click on Profile.



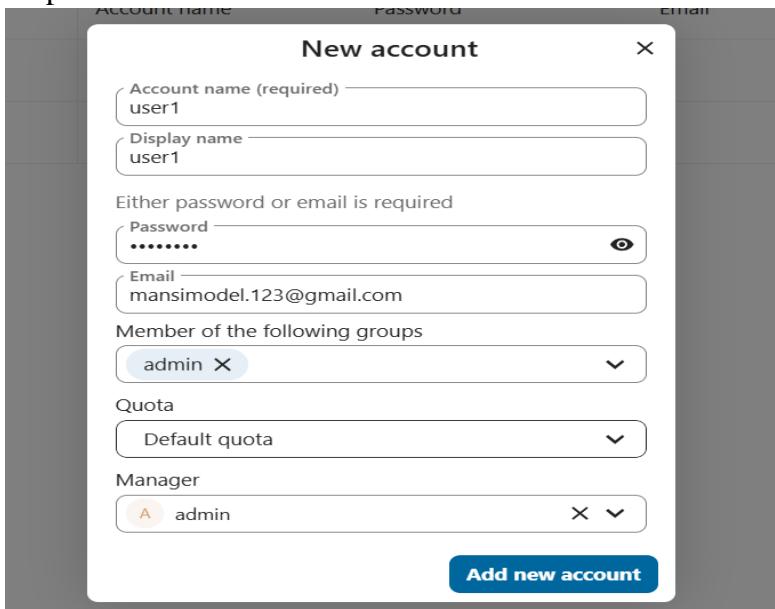
Step 20:- Select Accounts



Step 21:- Click on New account.



Step 22:- Fill in the details → Click on Add new account.

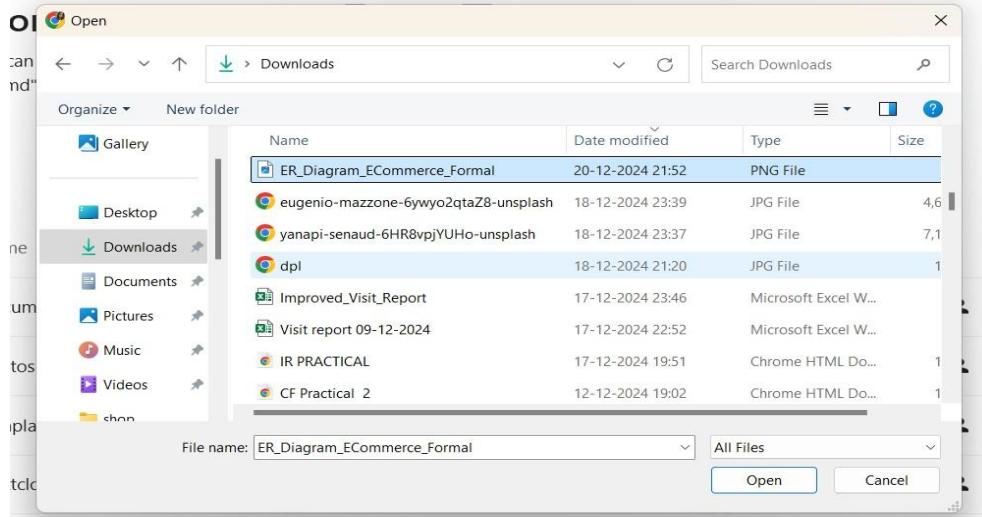


Step 23:- You can see the new user has been created successfully.

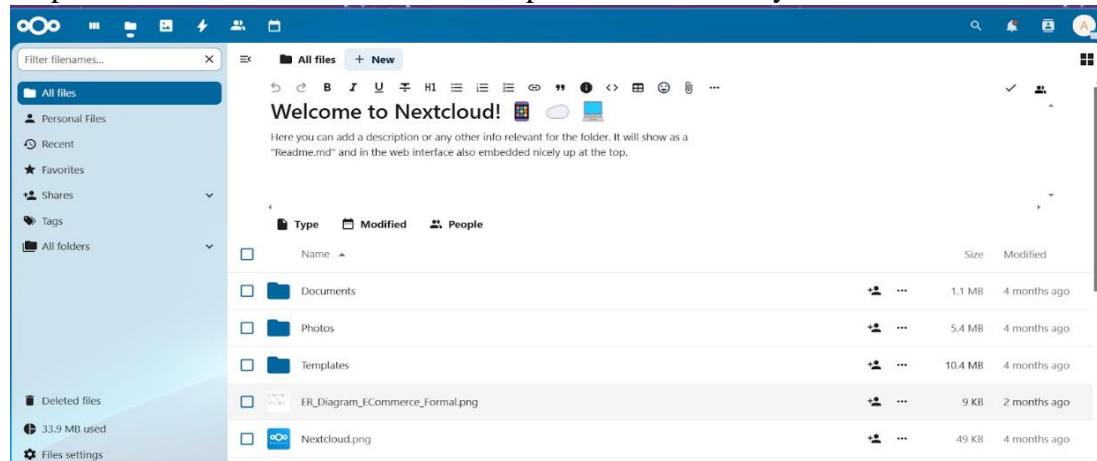
Display name	Account name	Password	Email	Groups
U user1	user1		mansimodel.123@gmail.c...	admin
A admin	admin			admin
U user	user			

Step 24:- To add a file, click on New → select Upload files.

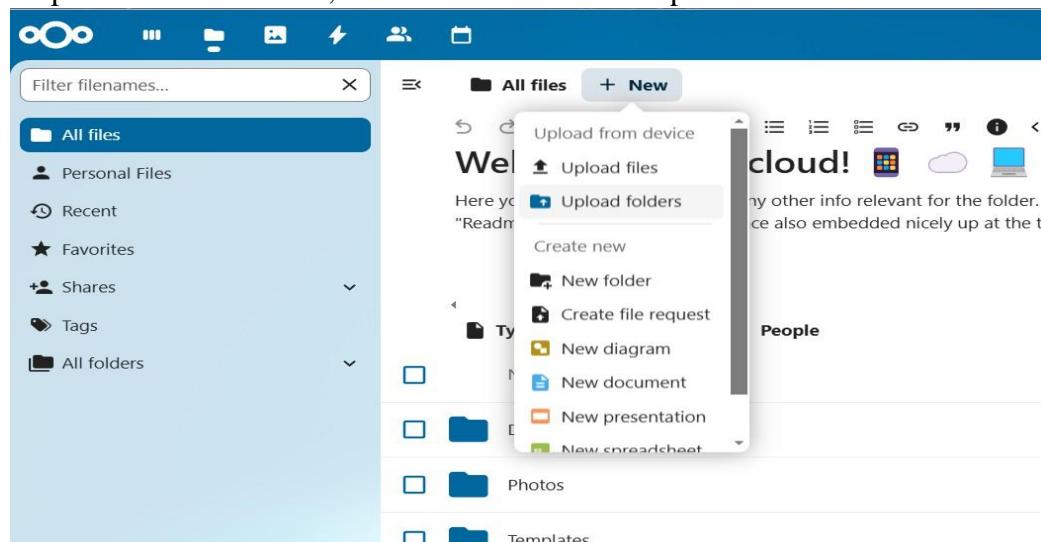
Step 25:- Browse the desired file → click on open.



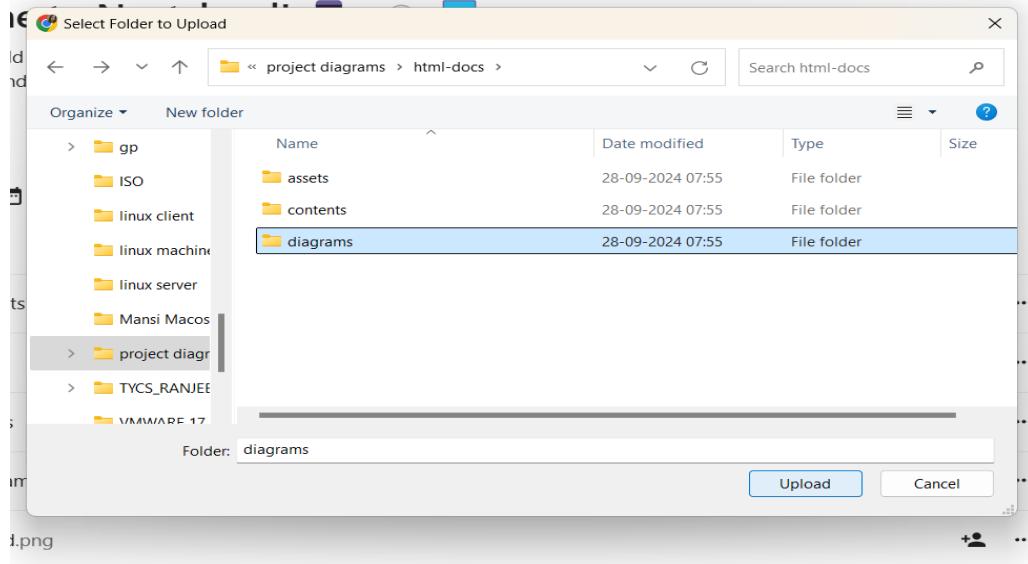
Step 26:- You can see the file has been uploaded successfully.



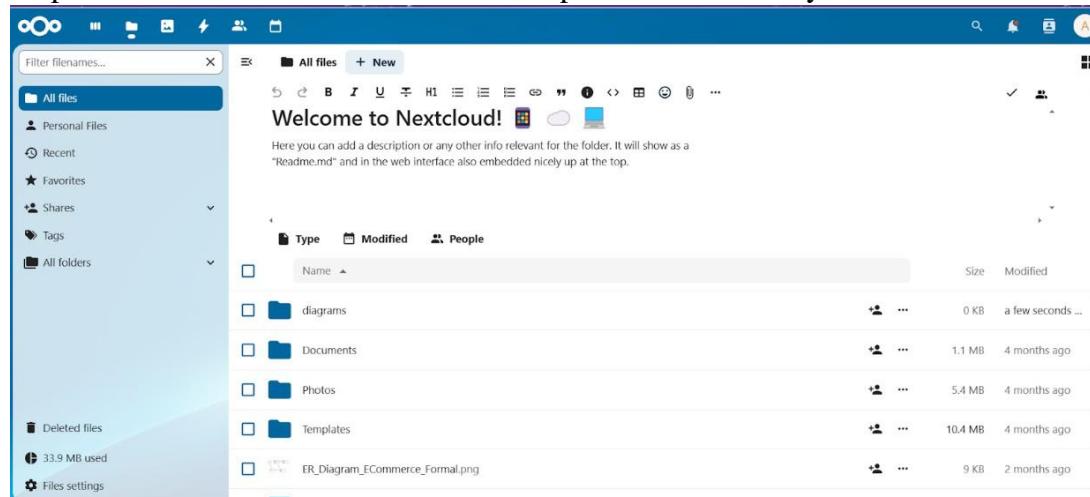
Step 27:- To add a folder, click on New → Select Upload folders.



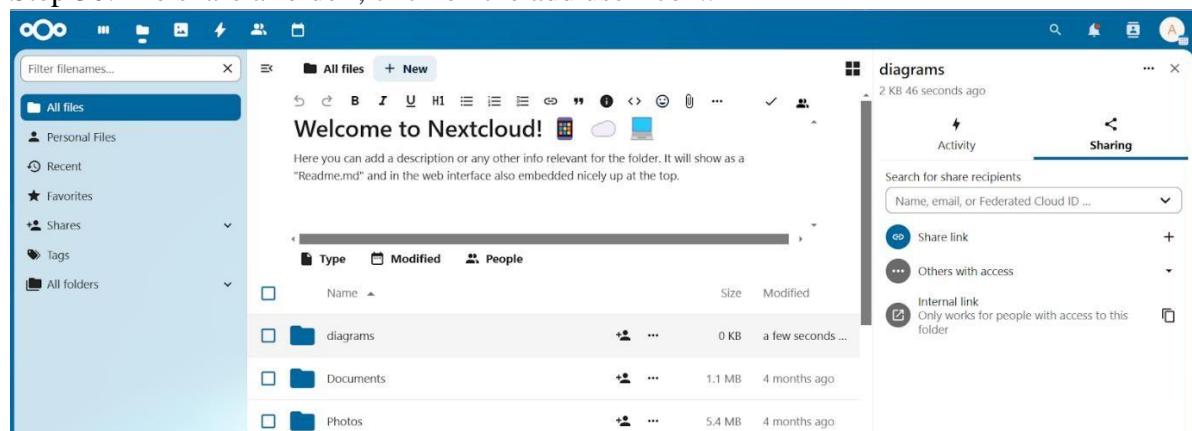
Step 28:- Browse the desired folder → click on upload.



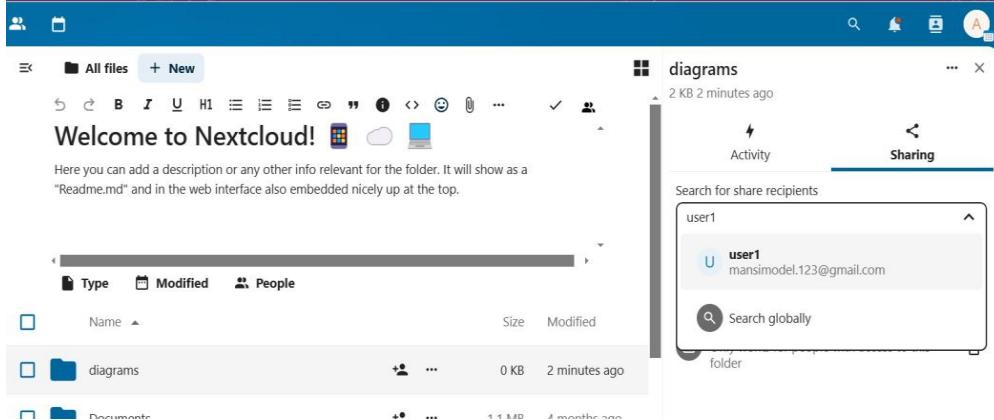
Step 29:- You can see the folder has been uploaded successfully.



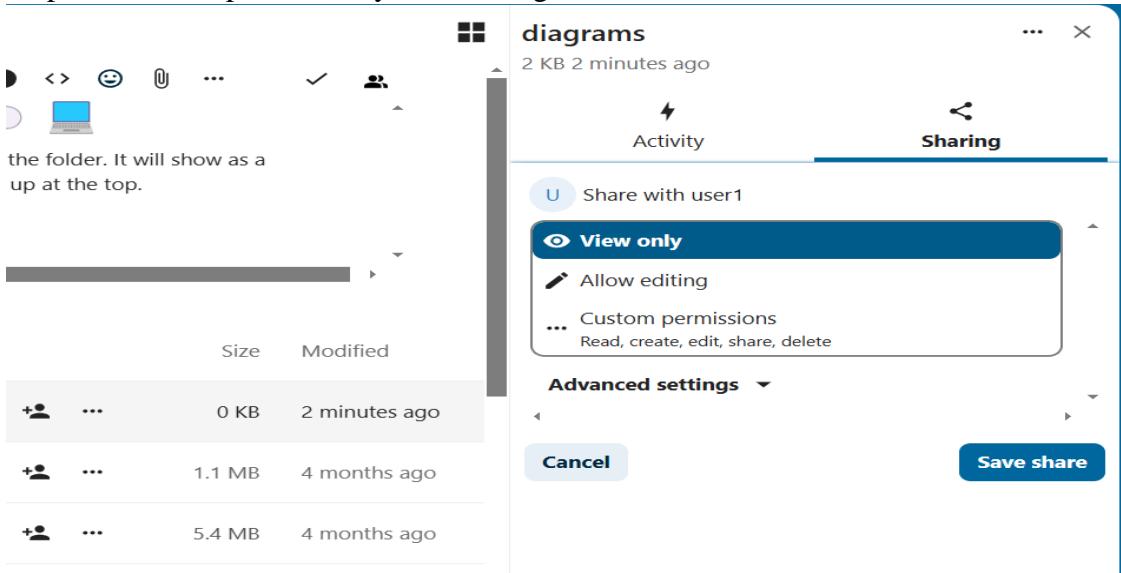
Step 30:- To share a folder , click on the add user icon..



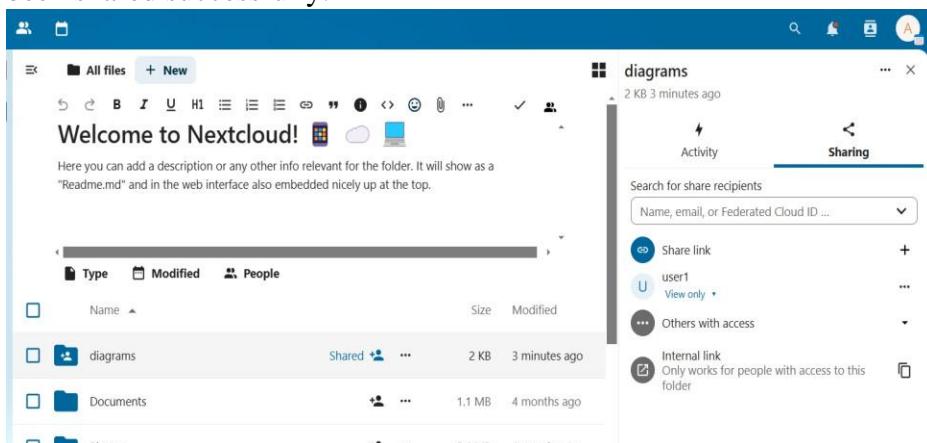
Step 31:- Select the user you want to share with .



Step 32:- Set the permission you want to give → click on Save share.



Step 33:- You can see ‘shared’ front of the folder indicating the folder has been shared successfully.



Pratik Patil

Roll No: 58

Step 34:- To share a file, click on add user icon → select the user you want to share with.

The screenshot shows a file list on the left and a sharing dialog on the right. The file list includes:

- diagrams (Shared, 2 KB, 3 minutes ago)
- Documents (Shared, 1.1 MB, 4 months ago)
- Photos (Shared, 5.4 MB, 4 months ago)
- Templates (Shared, 10.4 MB, 4 months ago)
- ER_Diagram_ECommerce_Formal.png (Shared, 9 KB, 2 months ago)
- Nextcloud.png (Shared, 49 KB, 4 months ago)

The sharing dialog shows a search bar with "user" and a list of users:

- user (Shared)
- user (Shared)
- user1 (mansimodel.123@gmail.com)

A "Search globally" button is also present.

Step 35:- Set the permission you want to give → click on Save share.

The screenshot shows the sharing settings for the file "ER_Diagram_ECommerce_Formal.p...". The "Sharing" tab is selected. The "Share with user" section shows:

- View only (disabled)
- Allow editing** (selected)
- Custom permissions (Read, create, edit, share, delete)

A "Save share" button is visible at the bottom right.

Step 36:- You can see 'shared' in front of the file indicating the file has been shared successfully.

The screenshot shows the file list again, with the "ER_Diagram_ECommerce_Formal.png" file now showing "Shared" in front of its name, indicating it has been successfully shared.

BSCS602

Pratik Patil

Roll No: 58

KERALEEYA SAMAJAM(REGD.) DOMBIVLI'S
MODEL COLLEGE
EMPOWERED AUTONOMOUS