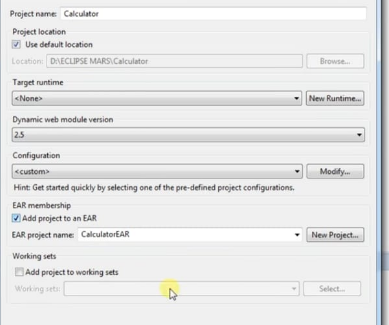
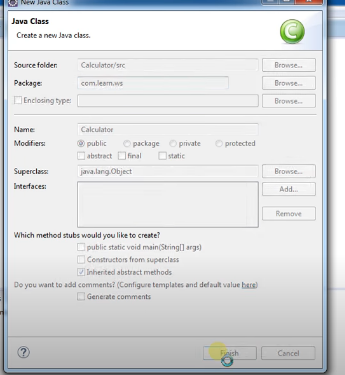
**SOFTWARE ARCHITECTURE EXPERIMENT**

**EXP1: WEB SERVICE**

**1: create new file, new dynamic web project name calculator add project on ear finish**



**2: create class**



**3: code**

**package** com.learn.ws;

**public** **class** Calculator {

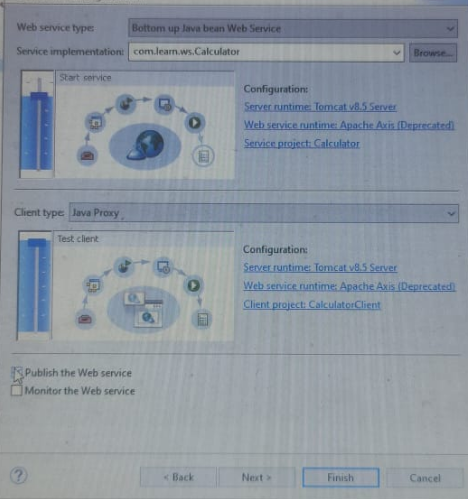
**public** **int** add(**int** a, **int** b) {

**return** a+b;

}

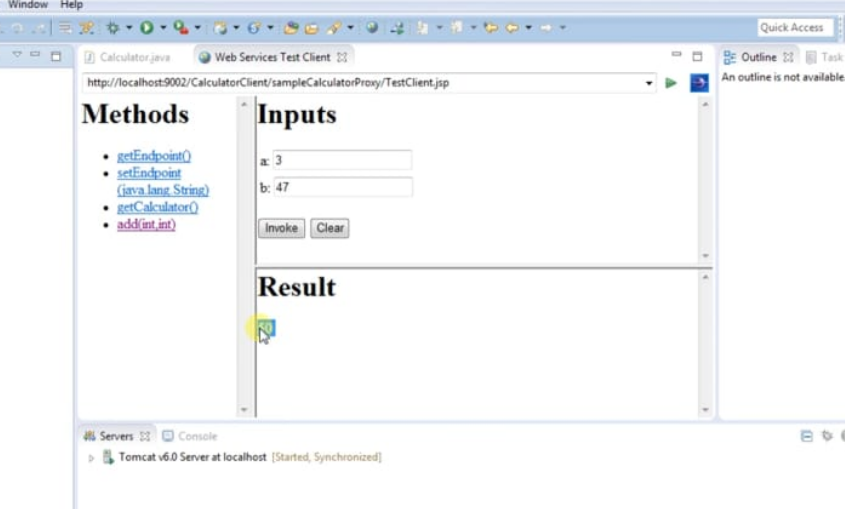
}

**4: right click on project name goto others select web service folder, select web service**



**4:click on next, again next then finish**

**5: it will show test client jsp**



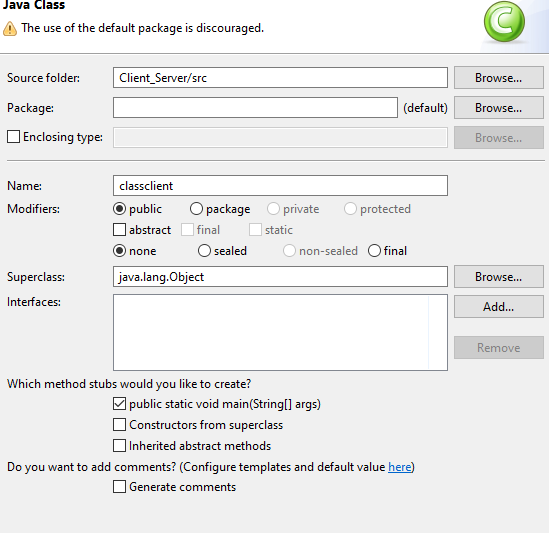
**EXP2: CLIENT SERVER SOCKET PROGRAMMING**

**1: create new java project give name**

**2: open perspective**

**3: create class for client and server**

**4: create package**



**3: create package**

**Code:**

**Client**

**package** com.network;

**import** java.io.\*;

**import** java.net.Socket;

**import** java.util.Scanner;

**public** **class** classclient {

**public** **static** **void** main(String[] args) **throws** Exception {

// **TODO** Auto-generated method stub

Socket csoc = **new** Socket("localhost", 40002);

OutputStream os = csoc.getOutputStream();

DataOutputStream dos = **new** DataOutputStream(os);

InputStream is = csoc.getInputStream();

DataInputStream dis= **new** DataInputStream(is);

**for**(**int** i=0;i<3; i++)

{

Scanner scan = **new** Scanner(System.***in***);

String s = scan.nextLine();

dos.writeUTF(s);

System.***out***.println(dis.readUTF());

}

}

}

**Server:**

**package** com.network;

**import** java.io.\*;

**import** java.net.ServerSocket;

**import** java.net.Socket;

**import** java.util.Scanner;

**public** **class** server {

**public** **static** **void** main(String[] args) **throws** Exception {

// **TODO** Auto-generated method stub

ServerSocket listener = **new** ServerSocket(40002);

System.***out***.println("Server is ready to accept the request");

Socket ssoc = listener.accept();

InputStream is = ssoc.getInputStream();

DataInputStream dis = **new** DataInputStream(is);

OutputStream os = ssoc.getOutputStream();

DataOutputStream dos = **new** DataOutputStream(os);

**for** (**int** i=0;i<3; i++)

{

System.***out***.println(dis.readUTF());

Scanner scan = **new** Scanner(System.***in***);

String msg = scan.nextLine();

dos.writeUTF(msg);

}

}

**Package:**

**package** com.network;

****

**EXP 3: FILE HANDLING**

**1:create a java project**

**2:create a package and name its pkg\_file**

**3: create a class and name its fileWrite add following code**

**Code: create a file**

**package** pkg\_file;

**import** java.io.File; // Import the File class

**import** java.io.FileWriter;

**import** java.io.IOException; // Import the 10Exception class to b errors

**public** **class** fileWrite {

**public** **static** **void** main(String[] args) {

**try** {

File myObj=**new** File("C:\Users\SOURABH JHA\OneDrive\Desktop\New folder");

**if** (myObj.createNewFile()) {

System.***out***.println("File created:"+ myObj.getName());

} **else** {

System.***out***.println("File already exists.");

}

FileWriter myWriter= **new** FileWriter(myObj);

myWriter.write("Hello! How are you today?");

myWriter.close();

System.***out***.println("Successfully wrote to the file.");

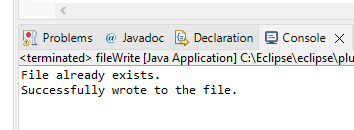
} **catch** (IOException e) {

System.***out***.println("An error occurred.");

e.printStackTrace();

}

}



**2: Read**

**package** pkg\_file;

**import** java.io.File;

**import** java.io.IOException;

**import** java.io.RandomAccessFile;

**import** java.lang.NumberFormatException;

**class** DisplayFile {

**public** **static** **void** main(String data[])

{

**try** {

String nameNumberString;

String name;

**long** number;

// Using file pointer creating the file.

File file = **new** File("C:\\Users\\Admin\\Desktop\\EXPERIMENTS\\DisplayFile.txt");

**if** (!file.exists()) {

// Create a new file if not exists.

file.createNewFile();

}

**try** (// Opening file in reading and write mode.

RandomAccessFile raf = **new** RandomAccessFile(file, "rw")) {

**while** (raf.getFilePointer() < raf.length()) {

nameNumberString = raf.readLine();

String[] lineSplit = nameNumberString.split("!");

name = lineSplit[0];

number = Long.*parseLong*(lineSplit[1]);

System.***out***.println(

"Friend Name: " + name + "\n"

+ "Contact Number: " + number + "\n");

}

}

}

**catch** (IOException ioe)

{

System.***out***.println(ioe);

}

**catch** (NumberFormatException nef)

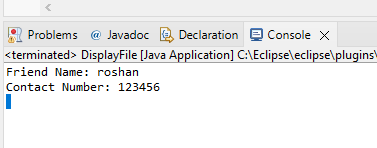
{

System.***out***.println(nef);

}

}

}



**Delete:**

**package** pkg\_file;

**import** java.io.File;

**import** java.io.IOException;

**public** **class** FileDelete

{

**public** **static** **void** main(String[] args)

{

**try**

{

File file = **new** File("C:\\Users\\Admin\\Desktop\\EXPERIMENTS1.dox"); //creates a file instance

file.deleteOnExit(); //deletes the file when JVM terminates

System.***out***.println("Done");

Thread.*sleep*(1000);

}

**catch**(Exception e)

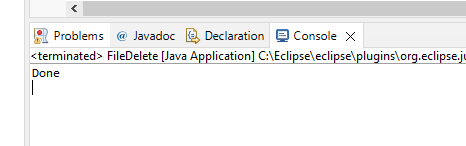
{

e.printStackTrace();

}

}

}



**JAVATPOINT LINK:** [**https://www.javatpoint.com/file-operations-in-java**](https://www.javatpoint.com/file-operations-in-java)

**EXP 4 - JAVA RMI**

Step1 – Create a Java Project

Step2 – Create package name pkg\_RMI

Step3 – Create interface name RMI\_Interface

Step 4 – Create 2 class for RMI\_Server and RMI\_Client

Interface

**package** pkg\_RMI;

**import** java.rmi.Remote;

**import** java.rmi.RemoteException;

**public** **interface** RMI\_interface **extends** Remote{

**public** **void** displayMessage() **throws** RemoteException;

}

Server

package pkg\_RMI;

import java.rmi.server.UnicastRemoteObject;

import java.rmi.AlreadyBoundException;

import java.rmi.RemoteException;

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

import java.rmi.server.UnicastRemoteObject;

public class RMI\_Server extends UnicastRemoteObject implements

RMI\_interface{

public RMI\_Server() throws RemoteException {

super();

}

public static void main(String[] args)throws RemoteException,

AlreadyBoundException {

try {

Registry registry =

LocateRegistry.createRegistry(1878);

registry.bind("hello", new RMI\_Server());

System.out.println("The RMI\_Server is running and ready...");

}

catch (Exception e) {

System.out.println("The RMI\_Server is not running...");

}

}

@Override

public void displayMessage()throws RemoteException{

System.out.println("----------------------------------------------");

System.out.println("Hello Bhai!");

System.out.println("----------------------------------------------");

}

}

Client

**package pkg\_RMI;**

**import java.net.MalformedURLException;**

**import java.rmi.Naming;**

**import java.rmi.NotBoundException;**

**import java.rmi.RemoteException;**

**public class RMI\_Client {**

**public static void main(String[] args) throws**

**MalformedURLException, RemoteException,NotBoundException{**

**try {**

**RMI\_interface helloAPI = (RMI\_interface)**

**Naming.lookup("rmi://localhost:1878/hello");**

**helloAPI.displayMessage();**

**}**

**catch(Exception e)**

**{**

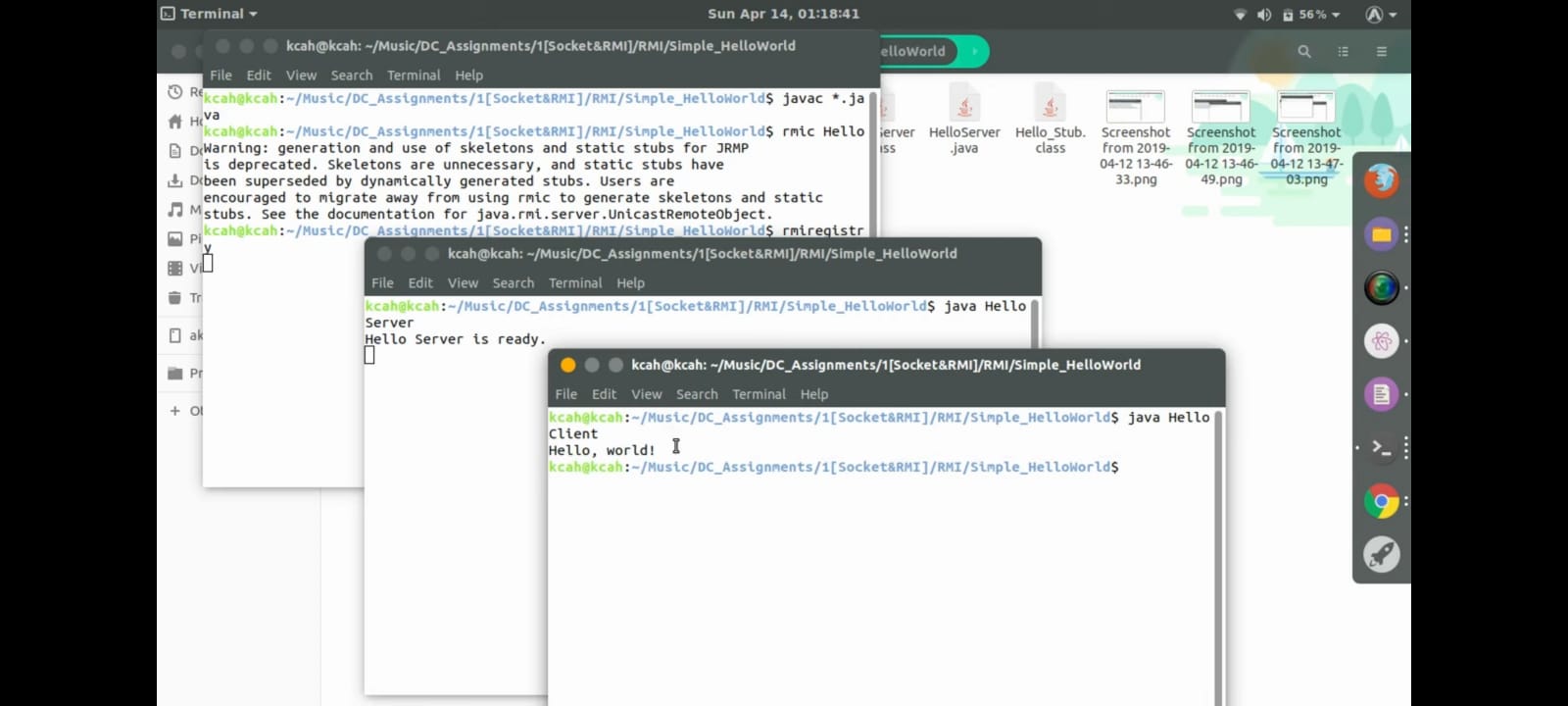
**System.out.println("The RMI APP is not running...");**

**e.printStackTrace();**

**}**

**}**

**}**

**op**

**import** java.io.File;

**import** java.io.IOException;

**import** java.io.RandomAccessFile;

**import** java.lang.NumberFormatException;

**class** UpdateFriend {

**public** **static** **void** main(String data[])

    {

**try** {

            // Get the name of the contact to be updated

            // from the Command line argument

            String newName = data[0];

            // Get the number to be updated

            // from the Command line argument

**long** newNumber = Long.parseLong(data[1]);

            String nameNumberString;

            String name;

**long** number;

**int** index;

            // Using file pointer creating the file.

            File file = **new** File("friendsContact.txt");

**if** (!file.exists()) {

                // Create a new file if not exists.

                file.createNewFile();

            }

            // Opening file in reading and write mode.

            RandomAccessFile raf

                = **new** RandomAccessFile(file, "rw");

**boolean** found = **false**;

            // Checking whether the name

            // of contact already exists.

            // getFilePointer() give the current offset

            // value from start of the file.

**while** (raf.getFilePointer() < raf.length()) {

                // reading line from the file.

                nameNumberString = raf.readLine();

                // splitting the string to get name and

                // number

                String[] lineSplit

                    = nameNumberString.split("!");

                // separating name and number.

                name = lineSplit[0];

                number = Long.parseLong(lineSplit[1]);

                // if condition to find existence of record.

**if** (name == newName

                    || number == newNumber) {

                    found = **true**;

**break**;

                }

            }

            // Update the contact if record exists.

**if** (found == **true**) {

                // Creating a temporary file

                // with file pointer as tmpFile.

                File tmpFile = **new** File("temp.txt");

                // Opening this temporary file

                // in ReadWrite Mode

                RandomAccessFile tmpraf

                    = **new** RandomAccessFile(tmpFile, "rw");

                // Set file pointer to start

                raf.seek(0);

                // Traversing the friendsContact.txt file

**while** (raf.getFilePointer()

                       < raf.length()) {

                    // Reading the contact from the file

                    nameNumberString = raf.readLine();

                    index = nameNumberString.indexOf('!');

                    name = nameNumberString.substring(

                        0, index);

                    // Check if the fetched contact

                    // is the one to be updated

**if** (name.equals(inputName)) {

                        // Update the number of this contact

                        nameNumberString

                            = name + "!"

                              + String.valueOf(newNumber);

                    }

                    // Add this contact in the temporary

                    // file

                    tmpraf.writeBytes(nameNumberString);

                    // Add the line separator in the

                    // temporary file

                    tmpraf.writeBytes(

                        System.lineSeparator());

                }

                // The contact has been updated now

                // So copy the updated content from

                // the temporary file to original file.

                // Set both files pointers to start

                raf.seek(0);

                tmpraf.seek(0);

                // Copy the contents from

                // the temporary file to original file.

**while** (tmpraf.getFilePointer()

                       < tmpraf.length()) {

                    raf.writeBytes(tmpraf.readLine());

                    raf.writeBytes(System.lineSeparator());

                }

                // Set the length of the original file

                // to that of temporary.

                raf.setLength(tmpraf.length());

                // Closing the resources.

                tmpraf.close();

                raf.close();

                // Deleting the temporary file

                tmpFile.delete();

                System.out.println(" Friend updated. ");

            }

            // The contact to be updated

            // could not be found

**else** {

                // Closing the resources.

                raf.close();

                // Print the message

                System.out.println(" Input name"

                                   + " does not exists. ");

            }

        }

**catch** (IOException ioe) {

            System.out.println(ioe);

        }

**catch** (NumberFormatException nef) {

            System.out.println(nef);

        }

    }

}

**import** java.io.File;

**import** java.io.IOException;

**import** java.io.RandomAccessFile;

**import** java.lang.NumberFormatException;

**class** UpdateFriend {

**public** **static** **void** main(String data[])

    {

**try** {

            // Get the name of the contact to be updated

            // from the Command line argument

            String newName = data[0];

            // Get the number to be updated

            // from the Command line argument

**long** newNumber = Long.parseLong(data[1]);

            String nameNumberString;

            String name;

**long** number;

**int** index;

            // Using file pointer creating the file.

            File file = **new** File("friendsContact.txt");

**if** (!file.exists()) {

                // Create a new file if not exists.

                file.createNewFile();

            }

            // Opening file in reading and write mode.

            RandomAccessFile raf

                = **new** RandomAccessFile(file, "rw");

**boolean** found = **false**;

            // Checking whether the name

            // of contact already exists.

            // getFilePointer() give the current offset

            // value from start of the file.

**while** (raf.getFilePointer() < raf.length()) {

                // reading line from the file.

                nameNumberString = raf.readLine();

                // splitting the string to get name and

                // number

                String[] lineSplit

                    = nameNumberString.split("!");

                // separating name and number.

                name = lineSplit[0];

                number = Long.parseLong(lineSplit[1]);

                // if condition to find existence of record.

**if** (name == newName

                    || number == newNumber) {

                    found = **true**;

**break**;

                }

            }

            // Update the contact if record exists.

**if** (found == **true**) {

                // Creating a temporary file

                // with file pointer as tmpFile.

                File tmpFile = **new** File("temp.txt");

                // Opening this temporary file

                // in ReadWrite Mode

                RandomAccessFile tmpraf

                    = **new** RandomAccessFile(tmpFile, "rw");

                // Set file pointer to start

                raf.seek(0);

                // Traversing the friendsContact.txt file

**while** (raf.getFilePointer()

                       < raf.length()) {

                    // Reading the contact from the file

                    nameNumberString = raf.readLine();

                    index = nameNumberString.indexOf('!');

                    name = nameNumberString.substring(

                        0, index);

                    // Check if the fetched contact

                    // is the one to be updated

**if** (name.equals(inputName)) {

                        // Update the number of this contact

                        nameNumberString

                            = name + "!"

                              + String.valueOf(newNumber);

                    }

                    // Add this contact in the temporary

                    // file

                    tmpraf.writeBytes(nameNumberString);

                    // Add the line separator in the

                    // temporary file

                    tmpraf.writeBytes(

                        System.lineSeparator());

                }

                // The contact has been updated now

                // So copy the updated content from

                // the temporary file to original file.

                // Set both files pointers to start

                raf.seek(0);

                tmpraf.seek(0);

                // Copy the contents from

                // the temporary file to original file.

**while** (tmpraf.getFilePointer()

                       < tmpraf.length()) {

                    raf.writeBytes(tmpraf.readLine());

                    raf.writeBytes(System.lineSeparator());

                }

                // Set the length of the original file

                // to that of temporary.

                raf.setLength(tmpraf.length());

                // Closing the resources.

                tmpraf.close();

                raf.close();

                // Deleting the temporary file

                tmpFile.delete();

                System.out.println(" Friend updated. ");

            }

            // The contact to be updated

            // could not be found

**else** {

                // Closing the resources.

                raf.close();

                // Print the message

                System.out.println(" Input name"

                                   + " does not exists. ");

            }

        }

**catch** (IOException ioe) {

            System.out.println(ioe);

        }

**catch** (NumberFormatException nef) {

            System.out.println(nef);

        }

    }

}

**import** java.io.File;

**import** java.io.IOException;

**import** java.io.RandomAccessFile;

**import** java.lang.NumberFormatException;

**class** UpdateFriend {

**public** **static** **void** main(String data[])

    {

**try** {

            // Get the name of the contact to be updated

            // from the Command line argument

            String newName = data[0];

            // Get the number to be updated

            // from the Command line argument

**long** newNumber = Long.parseLong(data[1]);

            String nameNumberString;

            String name;

**long** number;

**int** index;

            // Using file pointer creating the file.

            File file = **new** File("friendsContact.txt");

**if** (!file.exists()) {

                // Create a new file if not exists.

                file.createNewFile();

            }

            // Opening file in reading and write mode.

            RandomAccessFile raf

                = **new** RandomAccessFile(file, "rw");

**boolean** found = **false**;

            // Checking whether the name

            // of contact already exists.

            // getFilePointer() give the current offset

            // value from start of the file.

**while** (raf.getFilePointer() < raf.length()) {

                // reading line from the file.

                nameNumberString = raf.readLine();

                // splitting the string to get name and

                // number

                String[] lineSplit

                    = nameNumberString.split("!");

                // separating name and number.

                name = lineSplit[0];

                number = Long.parseLong(lineSplit[1]);

                // if condition to find existence of record.

**if** (name == newName

                    || number == newNumber) {

                    found = **true**;

**break**;

                }

            }

            // Update the contact if record exists.

**if** (found == **true**) {

                // Creating a temporary file

                // with file pointer as tmpFile.

                File tmpFile = **new** File("temp.txt");

                // Opening this temporary file

                // in ReadWrite Mode

                RandomAccessFile tmpraf

                    = **new** RandomAccessFile(tmpFile, "rw");

                // Set file pointer to start

                raf.seek(0);

                // Traversing the friendsContact.txt file

**while** (raf.getFilePointer()

                       < raf.length()) {

                    // Reading the contact from the file

                    nameNumberString = raf.readLine();

                    index = nameNumberString.indexOf('!');

                    name = nameNumberString.substring(

                        0, index);

                    // Check if the fetched contact

                    // is the one to be updated

**if** (name.equals(inputName)) {

                        // Update the number of this contact

                        nameNumberString

                            = name + "!"

                              + String.valueOf(newNumber);

                    }

                    // Add this contact in the temporary

                    // file

                    tmpraf.writeBytes(nameNumberString);

                    // Add the line separator in the

                    // temporary file

                    tmpraf.writeBytes(

                        System.lineSeparator());

                }

                // The contact has been updated now

                // So copy the updated content from

                // the temporary file to original file.

                // Set both files pointers to start

                raf.seek(0);

                tmpraf.seek(0);

                // Copy the contents from

                // the temporary file to original file.

**while** (tmpraf.getFilePointer()

                       < tmpraf.length()) {

                    raf.writeBytes(tmpraf.readLine());

                    raf.writeBytes(System.lineSeparator());

                }

                // Set the length of the original file

                // to that of temporary.

                raf.setLength(tmpraf.length());

                // Closing the resources.

                tmpraf.close();

                raf.close();

                // Deleting the temporary file

                tmpFile.delete();

                System.out.println(" Friend updated. ");

            }

            // The contact to be updated

            // could not be found

**else** {

                // Closing the resources.

                raf.close();

                // Print the message

                System.out.println(" Input name"

                                   + " does not exists. ");

            }

        }

**catch** (IOException ioe) {

            System.out.println(ioe);

        }

**catch** (NumberFormatException nef) {

            System.out.println(nef);

        }

    }

}

**import** java.io.File;

**import** java.io.IOException;

**import** java.io.RandomAccessFile;

**import** java.lang.NumberFormatException;

**class** UpdateFriend {

**public** **static** **void** main(String data[])

    {

**try** {

            // Get the name of the contact to be updated

            // from the Command line argument

            String newName = data[0];

            // Get the number to be updated

            // from the Command line argument

**long** newNumber = Long.parseLong(data[1]);

            String nameNumberString;

            String name;

**long** number;

**int** index;

            // Using file pointer creating the file.

            File file = **new** File("friendsContact.txt");

**if** (!file.exists()) {

                // Create a new file if not exists.

                file.createNewFile();

            }

            // Opening file in reading and write mode.

            RandomAccessFile raf

                = **new** RandomAccessFile(file, "rw");

**boolean** found = **false**;

            // Checking whether the name

            // of contact already exists.

            // getFilePointer() give the current offset

            // value from start of the file.

**while** (raf.getFilePointer() < raf.length()) {

                // reading line from the file.

                nameNumberString = raf.readLine();

                // splitting the string to get name and

                // number

                String[] lineSplit

                    = nameNumberString.split("!");

                // separating name and number.

                name = lineSplit[0];

                number = Long.parseLong(lineSplit[1]);

                // if condition to find existence of record.

**if** (name == newName

                    || number == newNumber) {

                    found = **true**;

**break**;

                }

            }

            // Update the contact if record exists.

**if** (found == **true**) {

                // Creating a temporary file

                // with file pointer as tmpFile.

                File tmpFile = **new** File("temp.txt");

                // Opening this temporary file

                // in ReadWrite Mode

                RandomAccessFile tmpraf

                    = **new** RandomAccessFile(tmpFile, "rw");

                // Set file pointer to start

                raf.seek(0);

                // Traversing the friendsContact.txt file

**while** (raf.getFilePointer()

                       < raf.length()) {

                    // Reading the contact from the file

                    nameNumberString = raf.readLine();

                    index = nameNumberString.indexOf('!');

                    name = nameNumberString.substring(

                        0, index);

                    // Check if the fetched contact

                    // is the one to be updated

**if** (name.equals(inputName)) {

                        // Update the number of this contact

                        nameNumberString

                            = name + "!"

                              + String.valueOf(newNumber);

                    }

                    // Add this contact in the temporary

                    // file

                    tmpraf.writeBytes(nameNumberString);

                    // Add the line separator in the

                    // temporary file

                    tmpraf.writeBytes(

                        System.lineSeparator());

                }

                // The contact has been updated now

                // So copy the updated content from

                // the temporary file to original file.

                // Set both files pointers to start

                raf.seek(0);

                tmpraf.seek(0);

                // Copy the contents from

                // the temporary file to original file.

**while** (tmpraf.getFilePointer()

                       < tmpraf.length()) {

                    raf.writeBytes(tmpraf.readLine());

                    raf.writeBytes(System.lineSeparator());

                }

                // Set the length of the original file

                // to that of temporary.

                raf.setLength(tmpraf.length());

                // Closing the resources.

                tmpraf.close();

                raf.close();

                // Deleting the temporary file

                tmpFile.delete();

                System.out.println(" Friend updated. ");

            }

            // The contact to be updated

            // could not be found

**else** {

                // Closing the resources.

                raf.close();

                // Print the message

                System.out.println(" Input name"

                                   + " does not exists. ");

            }

        }

**catch** (IOException ioe) {

            System.out.println(ioe);

        }

**catch** (NumberFormatException nef) {

            System.out.println(nef);

        }

    }

}

**import** java.io.File;

**import** java.io.IOException;

**import** java.io.RandomAccessFile;

**import** java.lang.NumberFormatException;

**class** UpdateFriend {

**public** **static** **void** main(String data[])

    {

**try** {

            // Get the name of the contact to be updated

            // from the Command line argument

            String newName = data[0];

            // Get the number to be updated

            // from the Command line argument

**long** newNumber = Long.parseLong(data[1]);

            String nameNumberString;

            String name;

**long** number;

**int** index;

            // Using file pointer creating the file.

            File file = **new** File("friendsContact.txt");

**if** (!file.exists()) {

                // Create a new file if not exists.

                file.createNewFile();

            }

            // Opening file in reading and write mode.

            RandomAccessFile raf

                = **new** RandomAccessFile(file, "rw");

**boolean** found = **false**;

            // Checking whether the name

            // of contact already exists.

            // getFilePointer() give the current offset

            // value from start of the file.

**while** (raf.getFilePointer() < raf.length()) {

                // reading line from the file.

                nameNumberString = raf.readLine();

                // splitting the string to get name and

                // number

                String[] lineSplit

                    = nameNumberString.split("!");

                // separating name and number.

                name = lineSplit[0];

                number = Long.parseLong(lineSplit[1]);

                // if condition to find existence of record.

**if** (name == newName

                    || number == newNumber) {

                    found = **true**;

**break**;

                }

            }

            // Update the contact if record exists.

**if** (found == **true**) {

                // Creating a temporary file

                // with file pointer as tmpFile.

                File tmpFile = **new** File("temp.txt");

                // Opening this temporary file

                // in ReadWrite Mode

                RandomAccessFile tmpraf

                    = **new** RandomAccessFile(tmpFile, "rw");

                // Set file pointer to start

                raf.seek(0);

                // Traversing the friendsContact.txt file

**while** (raf.getFilePointer()

                       < raf.length()) {

                    // Reading the contact from the file

                    nameNumberString = raf.readLine();

                    index = nameNumberString.indexOf('!');

                    name = nameNumberString.substring(

                        0, index);

                    // Check if the fetched contact

                    // is the one to be updated

**if** (name.equals(inputName)) {

                        // Update the number of this contact

                        nameNumberString

                            = name + "!"

                              + String.valueOf(newNumber);

                    }

                    // Add this contact in the temporary

                    // file

                    tmpraf.writeBytes(nameNumberString);

                    // Add the line separator in the

                    // temporary file

                    tmpraf.writeBytes(

                        System.lineSeparator());

                }

                // The contact has been updated now

                // So copy the updated content from

                // the temporary file to original file.

                // Set both files pointers to start

                raf.seek(0);

                tmpraf.seek(0);

                // Copy the contents from

                // the temporary file to original file.

**while** (tmpraf.getFilePointer()

                       < tmpraf.length()) {

                    raf.writeBytes(tmpraf.readLine());

                    raf.writeBytes(System.lineSeparator());

                }

                // Set the length of the original file

                // to that of temporary.

                raf.setLength(tmpraf.length());

                // Closing the resources.

                tmpraf.close();

                raf.close();

                // Deleting the temporary file

                tmpFile.delete();

                System.out.println(" Friend updated. ");

            }

            // The contact to be updated

            // could not be found

**else** {

                // Closing the resources.

                raf.close();

                // Print the message

                System.out.println(" Input name"

                                   + " does not exists. ");

            }

        }

**catch** (IOException ioe) {

            System.out.println(ioe);

        }

**catch** (NumberFormatException nef) {

            System.out.println(nef);

        }

    }

}

**import** java.io.File;

**import** java.io.IOException;

**import** java.io.RandomAccessFile;

**import** java.lang.NumberFormatException;

**class** UpdateFriend {

**public** **static** **void** main(String data[])

    {

**try** {

            // Get the name of the contact to be updated

            // from the Command line argument

            String newName = data[0];

            // Get the number to be updated

            // from the Command line argument

**long** newNumber = Long.parseLong(data[1]);

            String nameNumberString;

            String name;

**long** number;

**int** index;

            // Using file pointer creating the file.

            File file = **new** File("friendsContact.txt");

**if** (!file.exists()) {

                // Create a new file if not exists.

                file.createNewFile();

            }

            // Opening file in reading and write mode.

            RandomAccessFile raf

                = **new** RandomAccessFile(file, "rw");

**boolean** found = **false**;

            // Checking whether the name

            // of contact already exists.

            // getFilePointer() give the current offset

            // value from start of the file.

**while** (raf.getFilePointer() < raf.length()) {

                // reading line from the file.

                nameNumberString = raf.readLine();

                // splitting the string to get name and

                // number

                String[] lineSplit

                    = nameNumberString.split("!");

                // separating name and number.

                name = lineSplit[0];

                number = Long.parseLong(lineSplit[1]);

                // if condition to find existence of record.

**if** (name == newName

                    || number == newNumber) {

                    found = **true**;

**break**;

                }

            }

            // Update the contact if record exists.

**if** (found == **true**) {

                // Creating a temporary file

                // with file pointer as tmpFile.

                File tmpFile = **new** File("temp.txt");

                // Opening this temporary file

                // in ReadWrite Mode

                RandomAccessFile tmpraf

                    = **new** RandomAccessFile(tmpFile, "rw");

                // Set file pointer to start

                raf.seek(0);

                // Traversing the friendsContact.txt file

**while** (raf.getFilePointer()

                       < raf.length()) {

                    // Reading the contact from the file

                    nameNumberString = raf.readLine();

                    index = nameNumberString.indexOf('!');

                    name = nameNumberString.substring(

                        0, index);

                    // Check if the fetched contact

                    // is the one to be updated

**if** (name.equals(inputName)) {

                        // Update the number of this contact

                        nameNumberString

                            = name + "!"

                              + String.valueOf(newNumber);

                    }

                    // Add this contact in the temporary

                    // file

                    tmpraf.writeBytes(nameNumberString);

                    // Add the line separator in the

                    // temporary file

                    tmpraf.writeBytes(

                        System.lineSeparator());

                }

                // The contact has been updated now

                // So copy the updated content from

                // the temporary file to original file.

                // Set both files pointers to start

                raf.seek(0);

                tmpraf.seek(0);

                // Copy the contents from

                // the temporary file to original file.

**while** (tmpraf.getFilePointer()

                       < tmpraf.length()) {

                    raf.writeBytes(tmpraf.readLine());

                    raf.writeBytes(System.lineSeparator());

                }

                // Set the length of the original file

                // to that of temporary.

                raf.setLength(tmpraf.length());

                // Closing the resources.

                tmpraf.close();

                raf.close();

                // Deleting the temporary file

                tmpFile.delete();

                System.out.println(" Friend updated. ");

            }

            // The contact to be updated

            // could not be found

**else** {

                // Closing the resources.

                raf.close();

                // Print the message

                System.out.println(" Input name"

                                   + " does not exists. ");

            }

        }

**catch** (IOException ioe) {

            System.out.println(ioe);

        }

**catch** (NumberFormatException nef) {

            System.out.println(nef);

        }

    }

}

**import** java.io.File;

**import** java.io.IOException;

**import** java.io.RandomAccessFile;

**import** java.lang.NumberFormatException;

**class** UpdateFriend {

**public** **static** **void** main(String data[])

    {

**try** {

            // Get the name of the contact to be updated

            // from the Command line argument

            String newName = data[0];

            // Get the number to be updated

            // from the Command line argument

**long** newNumber = Long.parseLong(data[1]);

            String nameNumberString;

            String name;

**long** number;

**int** index;

            // Using file pointer creating the file.

            File file = **new** File("friendsContact.txt");

**if** (!file.exists()) {

                // Create a new file if not exists.

                file.createNewFile();

            }

            // Opening file in reading and write mode.

            RandomAccessFile raf

                = **new** RandomAccessFile(file, "rw");

**boolean** found = **false**;

            // Checking whether the name

            // of contact already exists.

            // getFilePointer() give the current offset

            // value from start of the file.

**while** (raf.getFilePointer() < raf.length()) {

                // reading line from the file.

                nameNumberString = raf.readLine();

                // splitting the string to get name and

                // number

                String[] lineSplit

                    = nameNumberString.split("!");

                // separating name and number.

                name = lineSplit[0];

                number = Long.parseLong(lineSplit[1]);

                // if condition to find existence of record.

**if** (name == newName

                    || number == newNumber) {

                    found = **true**;

**break**;

                }

            }

            // Update the contact if record exists.

**if** (found == **true**) {

                // Creating a temporary file

                // with file pointer as tmpFile.

                File tmpFile = **new** File("temp.txt");

                // Opening this temporary file

                // in ReadWrite Mode

                RandomAccessFile tmpraf

                    = **new** RandomAccessFile(tmpFile, "rw");

                // Set file pointer to start

                raf.seek(0);

                // Traversing the friendsContact.txt file

**while** (raf.getFilePointer()

                       < raf.length()) {

                    // Reading the contact from the file

                    nameNumberString = raf.readLine();

                    index = nameNumberString.indexOf('!');

                    name = nameNumberString.substring(

                        0, index);

                    // Check if the fetched contact

                    // is the one to be updated

**if** (name.equals(inputName)) {

                        // Update the number of this contact

                        nameNumberString

                            = name + "!"

                              + String.valueOf(newNumber);

                    }

                    // Add this contact in the temporary

                    // file

                    tmpraf.writeBytes(nameNumberString);

                    // Add the line separator in the

                    // temporary file

                    tmpraf.writeBytes(

                        System.lineSeparator());

                }

                // The contact has been updated now

                // So copy the updated content from

                // the temporary file to original file.

                // Set both files pointers to start

                raf.seek(0);

                tmpraf.seek(0);

                // Copy the contents from

                // the temporary file to original file.

**while** (tmpraf.getFilePointer()

                       < tmpraf.length()) {

                    raf.writeBytes(tmpraf.readLine());

                    raf.writeBytes(System.lineSeparator());

                }

                // Set the length of the original file

                // to that of temporary.

                raf.setLength(tmpraf.length());

                // Closing the resources.

                tmpraf.close();

                raf.close();

                // Deleting the temporary file

                tmpFile.delete();

                System.out.println(" Friend updated. ");

            }

            // The contact to be updated

            // could not be found

**else** {

                // Closing the resources.

                raf.close();

                // Print the message

                System.out.println(" Input name"

                                   + " does not exists. ");

            }

        }

**catch** (IOException ioe) {

            System.out.println(ioe);

        }

**catch** (NumberFormatException nef) {

            System.out.println(nef);

        }

    }

}

**import** java.io.File;

**import** java.io.IOException;

**import** java.io.RandomAccessFile;

**import** java.lang.NumberFormatException;

**class** UpdateFriend {

**public** **static** **void** main(String data[])

    {

**try** {

            // Get the name of the contact to be updated

            // from the Command line argument

            String newName = data[0];

            // Get the number to be updated

            // from the Command line argument

**long** newNumber = Long.parseLong(data[1]);

            String nameNumberString;

            String name;

**long** number;

**int** index;

            // Using file pointer creating the file.

            File file = **new** File("friendsContact.txt");

**if** (!file.exists()) {

                // Create a new file if not exists.

                file.createNewFile();

            }

            // Opening file in reading and write mode.

            RandomAccessFile raf

                = **new** RandomAccessFile(file, "rw");

**boolean** found = **false**;

            // Checking whether the name

            // of contact already exists.

            // getFilePointer() give the current offset

            // value from start of the file.

**while** (raf.getFilePointer() < raf.length()) {

                // reading line from the file.

                nameNumberString = raf.readLine();

                // splitting the string to get name and

                // number

                String[] lineSplit

                    = nameNumberString.split("!");

                // separating name and number.

                name = lineSplit[0];

                number = Long.parseLong(lineSplit[1]);

                // if condition to find existence of record.

**if** (name == newName

                    || number == newNumber) {

                    found = **true**;

**break**;

                }

            }

            // Update the contact if record exists.

**if** (found == **true**) {

                // Creating a temporary file

                // with file pointer as tmpFile.

                File tmpFile = **new** File("temp.txt");

                // Opening this temporary file

                // in ReadWrite Mode

                RandomAccessFile tmpraf

                    = **new** RandomAccessFile(tmpFile, "rw");

                // Set file pointer to start

                raf.seek(0);

                // Traversing the friendsContact.txt file

**while** (raf.getFilePointer()

                       < raf.length()) {

                    // Reading the contact from the file

                    nameNumberString = raf.readLine();

                    index = nameNumberString.indexOf('!');

                    name = nameNumberString.substring(

                        0, index);

                    // Check if the fetched contact

                    // is the one to be updated

**if** (name.equals(inputName)) {

                        // Update the number of this contact

                        nameNumberString

                            = name + "!"

                              + String.valueOf(newNumber);

                    }

                    // Add this contact in the temporary

                    // file

                    tmpraf.writeBytes(nameNumberString);

                    // Add the line separator in the

                    // temporary file

                    tmpraf.writeBytes(

                        System.lineSeparator());

                }

                // The contact has been updated now

                // So copy the updated content from

                // the temporary file to original file.

                // Set both files pointers to start

                raf.seek(0);

                tmpraf.seek(0);

                // Copy the contents from

                // the temporary file to original file.

**while** (tmpraf.getFilePointer()

                       < tmpraf.length()) {

                    raf.writeBytes(tmpraf.readLine());

                    raf.writeBytes(System.lineSeparator());

                }

                // Set the length of the original file

                // to that of temporary.

                raf.setLength(tmpraf.length());

                // Closing the resources.

                tmpraf.close();

                raf.close();

                // Deleting the temporary file

                tmpFile.delete();

                System.out.println(" Friend updated. ");

            }

            // The contact to be updated

            // could not be found

**else** {

                // Closing the resources.

                raf.close();

                // Print the message

                System.out.println(" Input name"

                                   + " does not exists. ");

            }

        }

**catch** (IOException ioe) {

            System.out.println(ioe);

        }

**catch** (NumberFormatException nef) {

            System.out.println(nef);

        }

    }

}

**Ip Address of machine**

import java.net.InetAddress;

import java.net.UnknownHostException;

public class Main {

public static void main(String[] args) throws UnknownHostException {

InetAddress localHost = InetAddress.getLocalHost();

System.out.println(localHost.getHostAddress());

    }

}