­­­LAB ASSIGNMENT

JAVA

ADARSH RANA

MCA 2C

20711150

1.Write a java program that implements [Array](http://45.116.207.86/moodle/mod/url/view.php?id=6252) Index out of bound Exception using built-in-Exception.

package filehandeling;

public class Quest1 {

    public static void main(String[] args)

    {

        int ar[]={1,2,3,4};

        try

        {

            System.out.println(ar[7]);

        }

        catch(IndexOutOfBoundsException e)

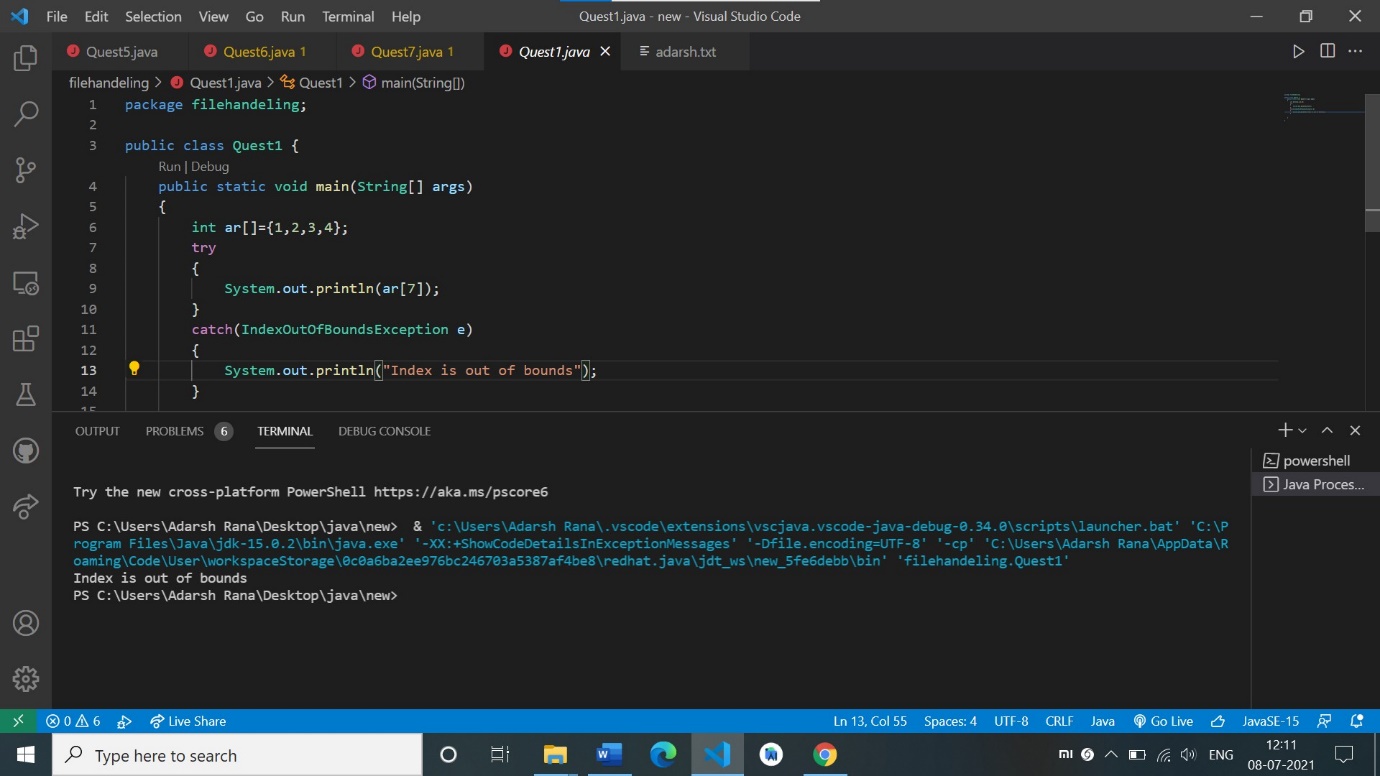
        {

            System.out.println("Index is out of bounds");

        }

    }

}



2. Write a java program that implements bank truncations using user defined exception .

package filehandeling;

class MyException extends Exception

    {

        public MyException(String s)

        {

            super(s);

        }

    }

public class Quest2 {

    public static void main(String[] args)

    {

        int balance\_amount=10000;

        int transactio\_amount=1200;

        try

        {

        throw new MyException("Transaction amount is more than balance");

        }

        catch( MyException e)

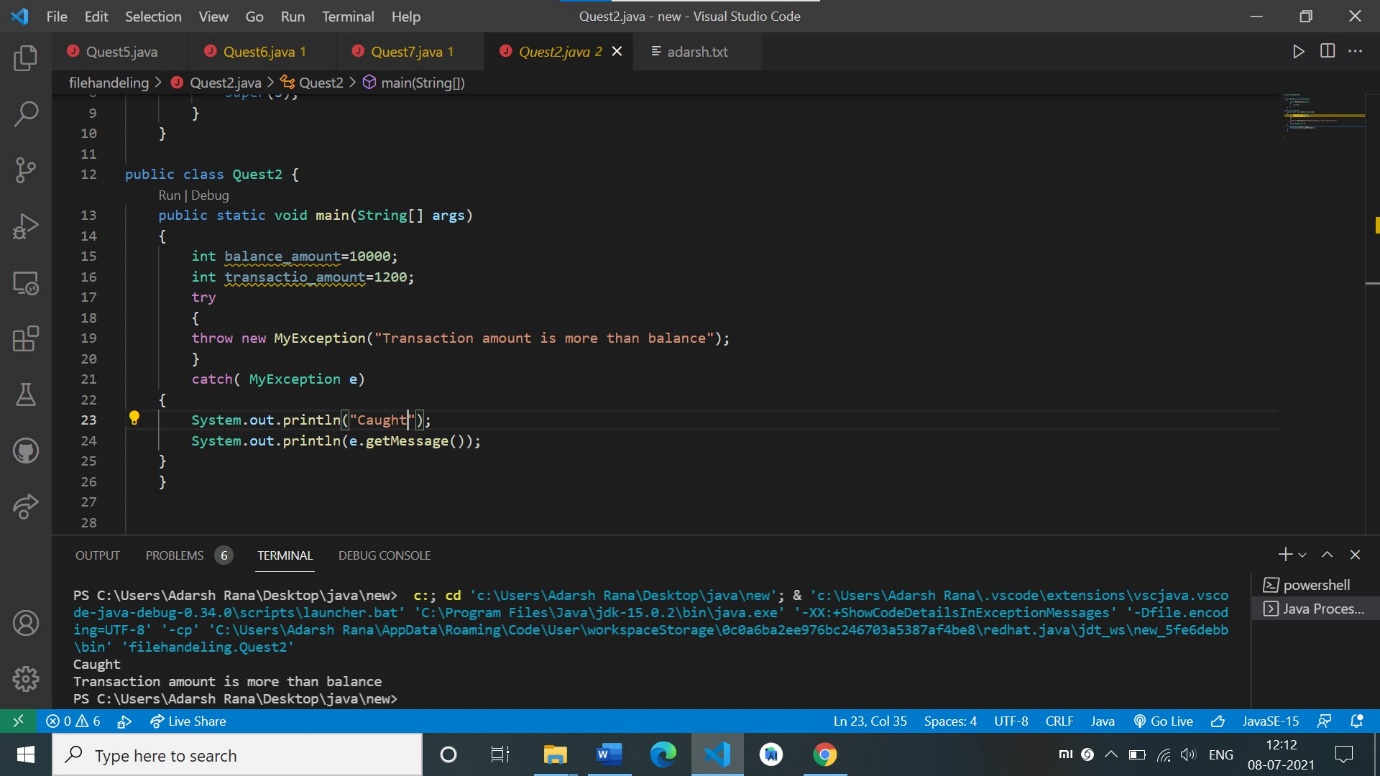
    {

        System.out.println("Caught");

        System.out.println(e.getMessage());

    }

    }



3. Write a java program to identify the significance of finally block in handling exceptions.

package filehandeling;

public class Quest3 {

    public static void main(String[] args)

    {

        int a[]={1,2};

        try

        {

            System.out.println(a[3]);

        }

        catch (IndexOutOfBoundsException e)

        {

            System.out.println("Accessing value out of bonds");

        }

        finally

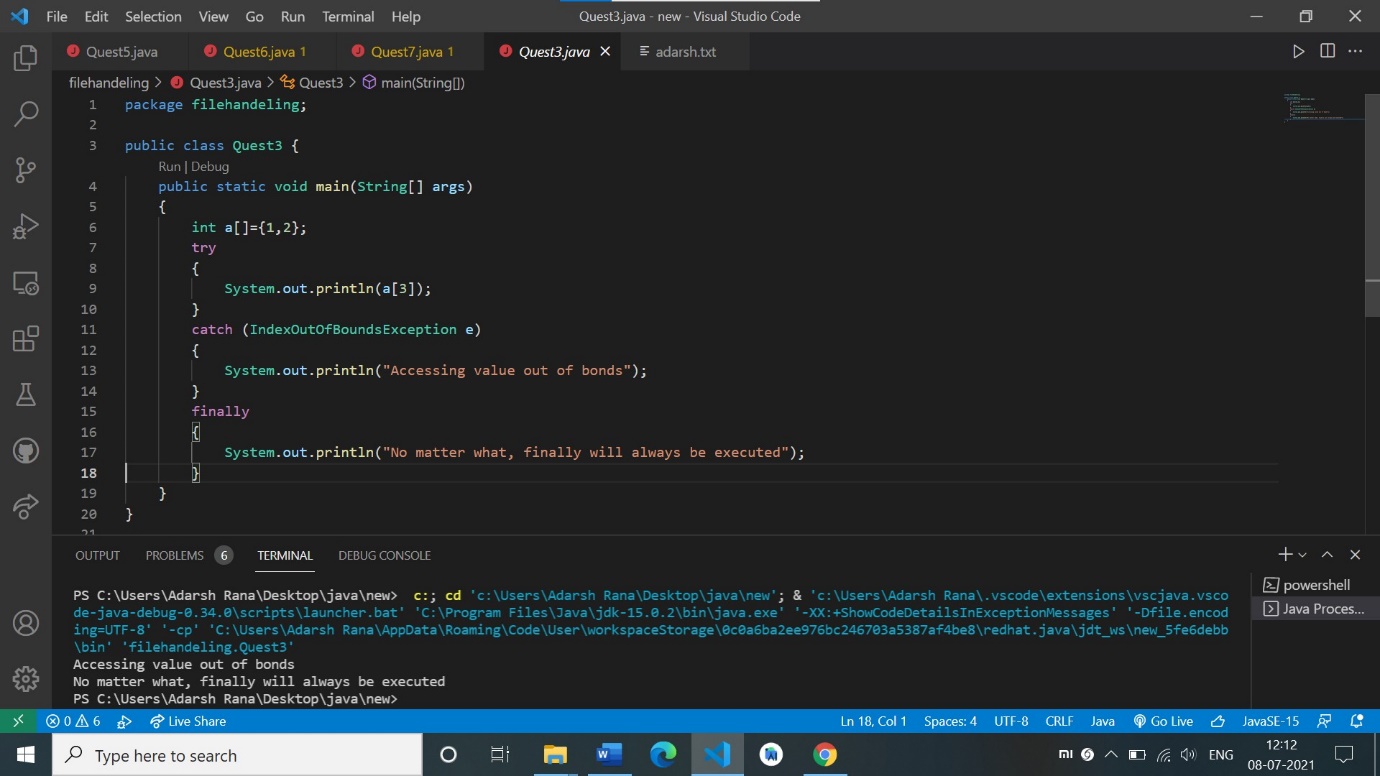
        {

            System.out.println("No matter what, finally will always be executed");

        }

    }

}



4. . Write a Java program that reads a file name from the user then displays information about whether that file exists, file is writable, the type of file and length of the file in bytes.

package filehandeling;

import java.io.File;

public class Quest4 {

    public static void main(String[] args)

    {

        File myobj=new File("C:\\Users\\Adarsh Rana\\Desktop\\java\\new\\firsr\\adarsh.txt");

        if(myobj.exists())

        {

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

            System.out.println("File is writable: "+myobj.canWrite());

            System.out.println("File is readable: "+myobj.canRead());

            System.out.println("Size of file in bytes: "+myobj.length());

            System.out.println("Is file a directory: "+myobj.isDirectory() );

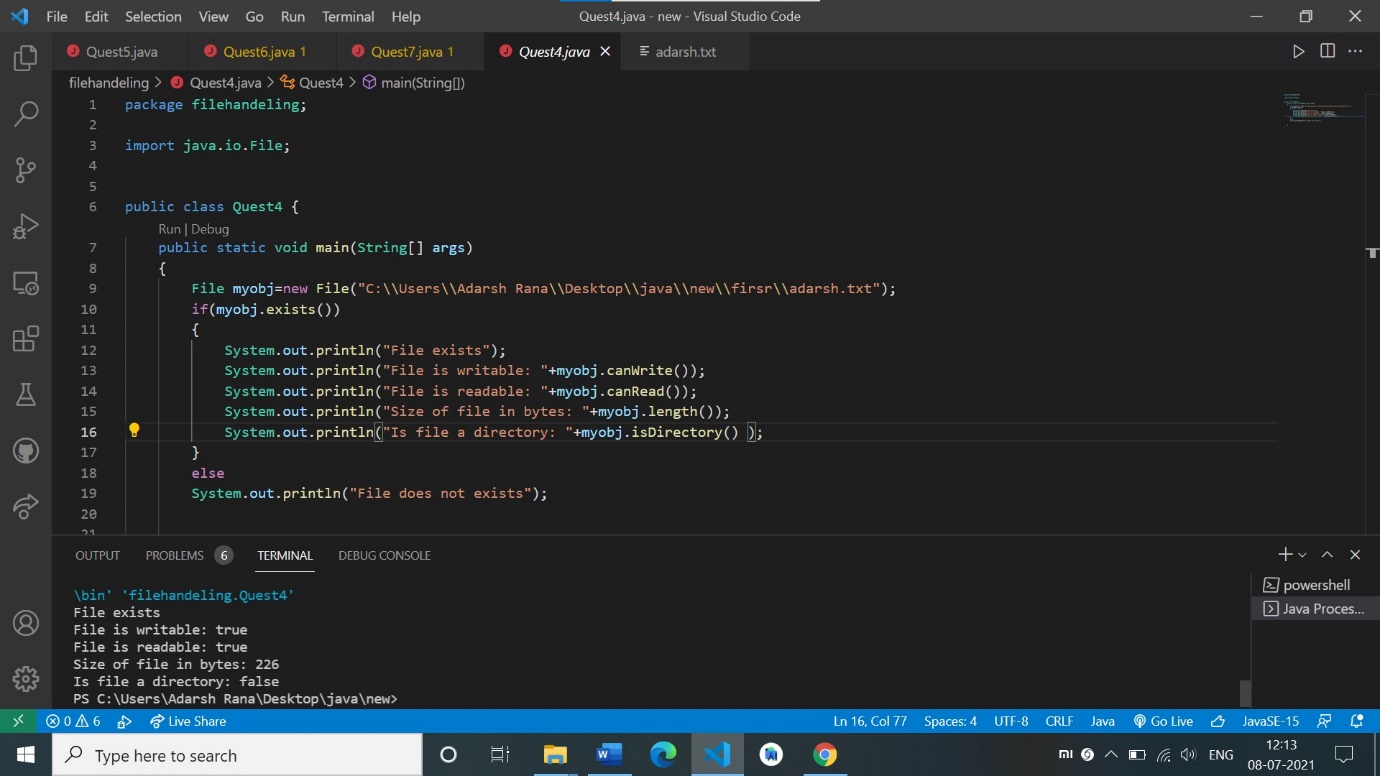
        }

        else

        System.out.println("File does not exists");

    }

}



 5. Write a Java program that reads a file and displays the file on the screen with a line number before each line.

package filehandeling;

import java.io.File;

import java.io.FileNotFoundException;

import java.util.Scanner;

public class Quest5 {

    public static void main(String[] args)

    {

        File fileobj1=new File("C:\\Users\\Adarsh Rana\\Desktop\\java\\new\\firsr\\adarsh.txt");

        try{

        Scanner fileobj2=new Scanner(fileobj1);

        int lineNumber=1;

        while(fileobj2.hasNext())

        {

            String line=fileobj2.nextLine();

            System.out.println(lineNumber+":"+line);

            lineNumber++;

        }

        fileobj2.close();

    }

    catch(FileNotFoundException e)

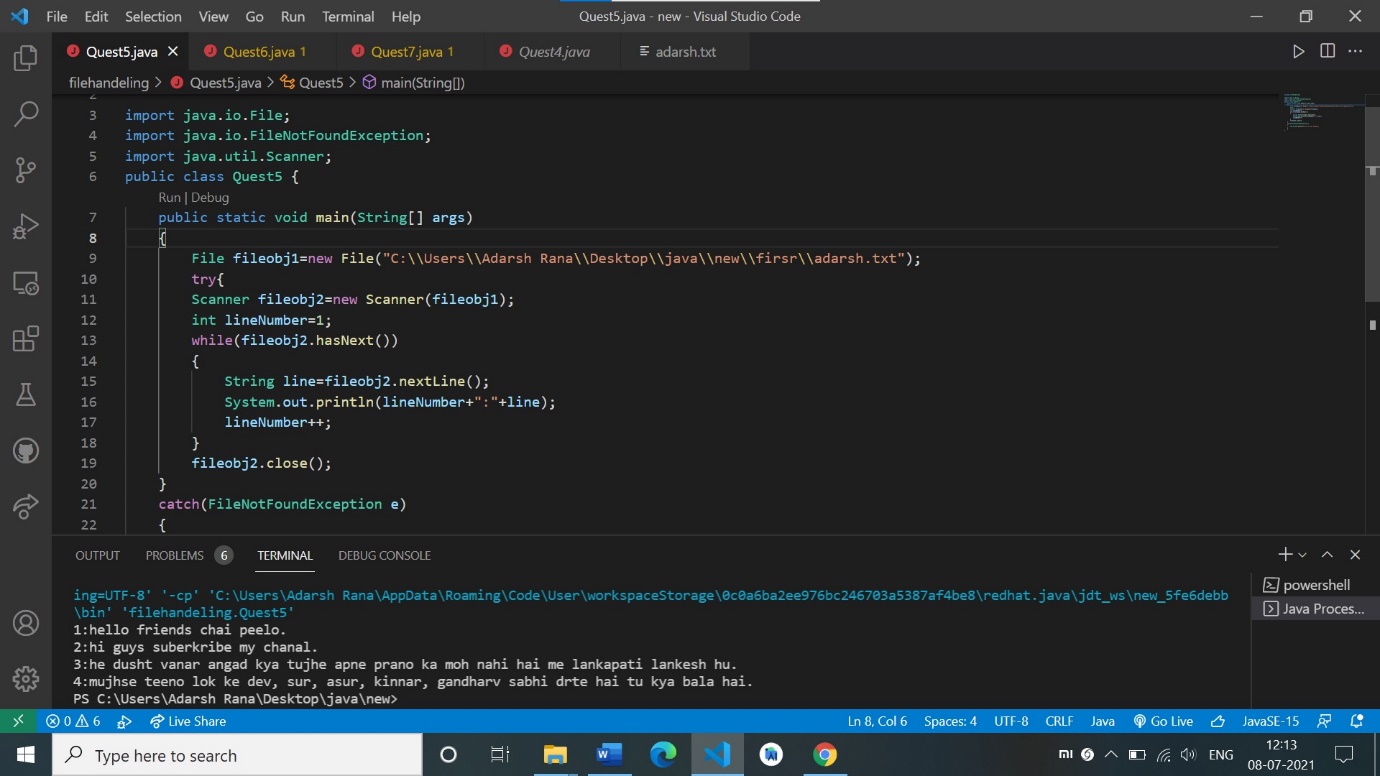
    {

        System.out.println("File is not found");

    }

    }

}



6. Write a Java program that reads a file and displays the no of lines and words in that file.

package filehandeling;

import java.io.File;

import java.io.FileNotFoundException;

import java.util.Scanner;

public class Quest6 {

    public static void main(String[] args)

    {

        File fileobj=new File("C:\\Users\\Adarsh Rana\\Desktop\\java\\new\\firsr\\adarsh.txt");

        if(fileobj.exists())

        {

            try

            {

                Scanner filescan=new Scanner(fileobj);

                int no\_of\_lines=0;

                // int no\_of\_characters=0;

                while(filescan.hasNext())

                {

                    String line=filescan.nextLine();

                    no\_of\_lines++;

                }

                filescan.close();

                System.out.println("Total number of lines: "+no\_of\_lines);

                System.out.println("Total number of characters: "+ (fileobj.length()-2\*no\_of\_lines));//every character is of one byte and every enter is of 2 byte.

            }

            catch(FileNotFoundException e)

            {

                System.out.println("File does not exist");

            }

        }

        else

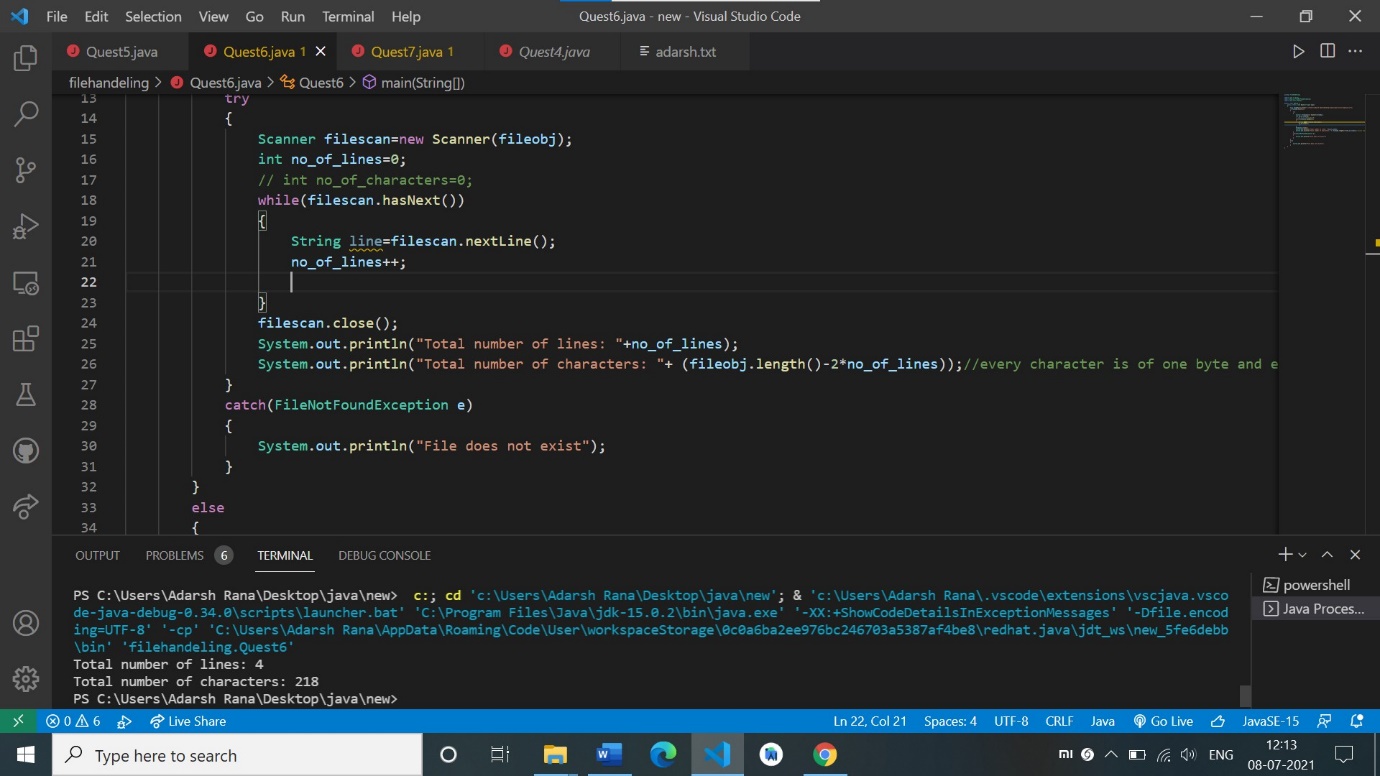
        {

            System.out.println("File does not exists");

        }

    }

}



7. Write a Java program that reads to copy source seat. File and display on the console

package filehandeling;

import java.io.\*;

import java.util.Scanner;

public class Quest7 {

    public static void main(String args[]) {

        String srcfname, dstfname;

        Scanner scan = new Scanner(System.in);

        System.out.print("Enter Source File Name (with extension like source.txt) : ");

        srcfname = scan.nextLine();

        System.out.print("Enter Destination File Name (with extension like destination.txt) : ");

        dstfname = scan.nextLine();

        InputStream inStream = null;

        OutputStream outStream = null;

        try {

            File file1 = new File(srcfname);

            File file2 = new File(dstfname);

            inStream = new FileInputStream(file1);

            outStream = new FileOutputStream(file2);

            byte[] buffer = new byte[1024];

            int length;

            while ((length = inStream.read(buffer)) > 0) {

                outStream.write(buffer, 0, length);

            }

            if (inStream != null)

                inStream.close();

            if (outStream != null)

                outStream.close();

            System.out.println("File Copied..");

        } catch (IOException e) {

            e.printStackTrace();

        }

    }

}

