

Assessment-4

Vashishth gajjar

19BCE2286

Lab Assessment-9

Q)

```
import java.io.*;
```

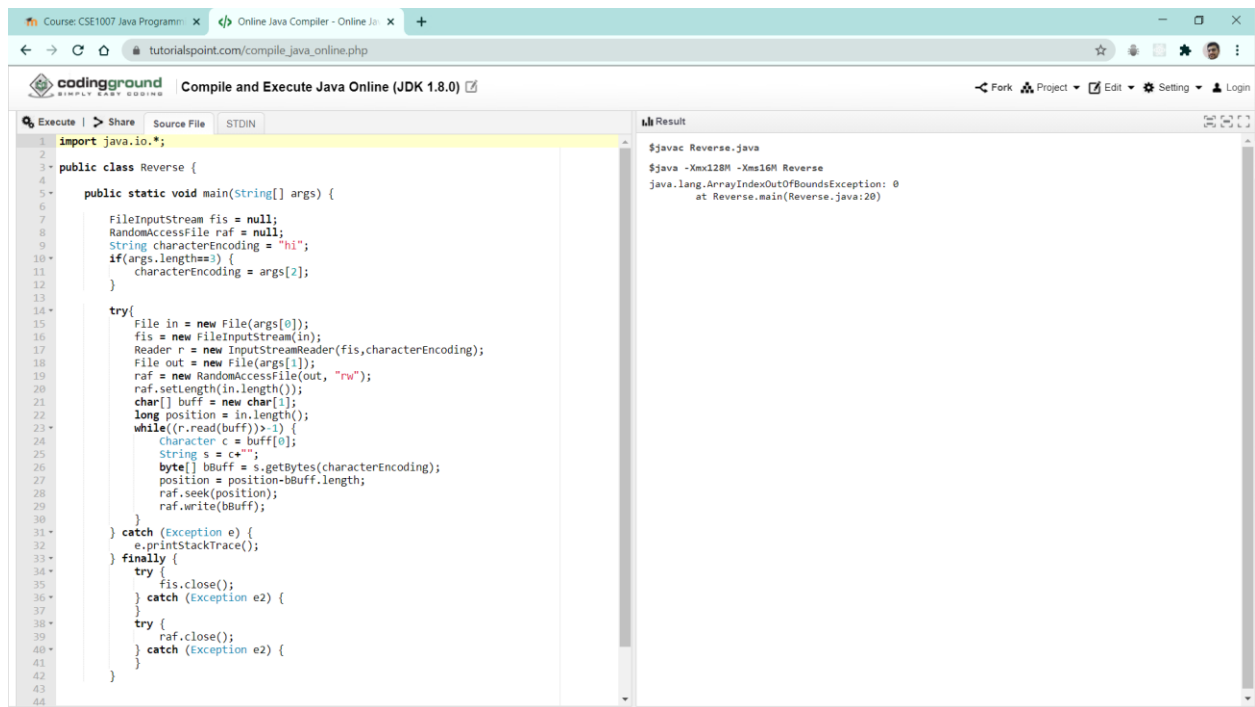
```
public class Reverse {
```

```
public static void main(String[] args) {  
  
    FileInputStream fis = null;  
    RandomAccessFile raf = null;  
    String characterEncoding = "hi";  
    if(args.length==3) {  
        characterEncoding = args[2];  
    }  
  
    try{  
        File in = new File(args[0]);  
        fis = new FileInputStream(in);  
        Reader r = new  
InputStreamReader(fis,characterEncoding);
```

```
File out = new File(args[1]);  
raf = new RandomAccessFile(out,  
"rw");  
  
raf.setLength(in.length());  
  
char[] buff = new char[1];  
  
long position = in.length();  
while((r.read(buff))>-1) {  
    Character c = buff[0];  
  
    String s = c+"";  
  
    byte[] bBuff =  
s.getBytes(characterEncoding);  
  
    position = position-bBuff.length;  
  
    raf.seek(position);  
  
    raf.write(bBuff);  
}
```

```
    }  
    } catch (Exception e) {  
        e.printStackTrace();  
    } finally {  
        try {  
            fis.close();  
        } catch (Exception e2) {  
        }  
        try {  
            raf.close();  
        } catch (Exception e2) {  
        }  
    }  
}
```

}



The screenshot shows a web browser window with the URL `tutorialspoint.com/compile_java_online.php`. The page is titled "Compile and Execute Java Online (JDK 1.8.0)". The left pane displays a Java program named `Reverse.java` that reads a file and writes its reverse to another file. The right pane shows the compilation and execution results, including the command `$javac Reverse.java` and the output `$java -Xmx128M -Xms16M Reverse`, which results in a `java.lang.ArrayIndexOutOfBoundsException: 0` at `Reverse.main(Reverse.java:20)`.

```
1 import java.io.*;
2
3 public class Reverse {
4
5     public static void main(String[] args) {
6
7         FileInputStream fis = null;
8         RandomAccessFile raf = null;
9         String characterEncoding = "hi";
10        if(args.length==3) {
11            characterEncoding = args[2];
12        }
13
14        try{
15            File in = new File(args[0]);
16            fis = new FileInputStream(in);
17            Reader r = new InputStreamReader(fis,characterEncoding);
18            File out = new File(args[1]);
19            raf = new RandomAccessFile(out, "rw");
20            raf.setLength(in.length());
21            char[] buff = new char[1];
22            long position = in.length();
23            while((r.read(buff))>-1) {
24                Character c = buff[0];
25                String s = c+"";
26                byte[] bBuff = s.getBytes(characterEncoding);
27                position = position-bBuff.length;
28                raf.seek(position);
29                raf.write(bBuff);
30            }
31        } catch (Exception e) {
32            e.printStackTrace();
33        } finally {
34            try {
35                fis.close();
36            } catch (Exception e2) {
37            }
38            try {
39                raf.close();
40            } catch (Exception e2) {
41            }
42        }
43    }
44 }
```

Result

```
$javac Reverse.java
$java -Xmx128M -Xms16M Reverse
java.lang.ArrayIndexOutOfBoundsException: 0
    at Reverse.main(Reverse.java:20)
```

Q)

```
import java.util.*;
```

```
import java.io.*;
```

```
public class Exercise58 {
```

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

```
        Scanner in = new Scanner(System.in);
```

```
System.out.print("Input a Sentence: ");  
String line = in.nextLine();  
String upper_case_line = "";  
Scanner lineScan = new Scanner(line);  
while(lineScan.hasNext()) {  
    String word = lineScan.next();  
    upper_case_line +=  
Character.toUpperCase(word.charAt(0)) +  
word.substring(1) + " ";  
}  
  
System.out.println(upper_case_line.trim());  
}  
}
```

Q)

```
import java.util.Scanner;

import java.io.*;

public class LowercaseFileConverter2
{
    public static void main(String[]
args)throws IOException
    {
        String filename;
        String message;
        String filename2;
```

```
Scanner keyboard = new  
Scanner(System.in);
```

```
System.out.print("Enter the  
filename: ");
```

```
filename = keyboard.nextLine();
```

```
FileWriter fwriter = new  
FileWriter(filename);
```

```
PrintWriter outputFile = new  
PrintWriter(fwriter);
```



```
System.out.println("Enter a message:  
");
```

```
message = keyboard.nextLine();
```

```
outputFile.println(message);
```

```
outputFile.close();
```

```
System.out.println("Enter the name  
of the second file: ");
```

```
filename2 = keyboard.nextLine();
```

```
FileReader freader = new  
FileReader(filename2);
```

```
BufferedReader inputFile = new  
BufferedReader(freader);
```

```
String str;
```

```
str = inputFile.readLine();
```

```
while (str != null)
```

```
{
```

```
        System.out.println(str);

        String upper =
message.toLowerCase();

        str = inputFile.readLine(lower);

    }

    inputFile.close();

}

}
```

Q)

```
import java.io.*;

public class Tester {
```

```
private static final String FILE_PATH =  
"data.txt";  
  
public static void main(String args[])  
throws IOException {  
  
    FileUtil fileUtil = new  
FileUtil(FILE_PATH);  
  
    System.out.println("No. of characters in  
file: " + fileUtil.getCharCount());  
  
    }  
}
```

```
class FileUtil {  
  
    static BufferedReader reader = null;
```

```
public FileUtil(String filePath) throws  
FileNotFoundException {  
    File file = new File(filePath);  
    FileInputStream fileStream = new  
FileInputStream(file);  
    InputStreamReader input = new  
InputStreamReader(fileStream);  
    reader = new BufferedReader(input);  
}
```

```
public static int getCharCount() throws  
IOException {  
    int charCount = 0;  
    String data;  
    while((data = reader.readLine()) != null) {
```

```
        charCount += data.length();  
    }  
    return charCount;  
}  
}
```

Q)

```
import java.io.*;  
  
public class FileMerge  
{  
    public static void main(String[] args)  
throws IOException  
    {
```

```
// PrintWriter object for file3.txt

PrintWriter pw = new
PrintWriter("file3.txt");


// BufferedReader object for file1.txt

BufferedReader br = new
BufferedReader(new
FileReader("file1.txt"));


String line = br.readLine();


// loop to copy each line of
// file1.txt to file3.txt
while (line != null)
```

```
{  
    pw.println(line);  
    line = br.readLine();  
}
```

```
br = new BufferedReader(new  
FileReader("file2.txt"));
```

```
line = br.readLine();
```

```
// loop to copy each line of
```

```
// file2.txt to file3.txt
```

```
while(line != null)
```

```
{
```



```
        pw.println(line);  
        line = br.readLine();  
    }
```

```
    pw.flush();
```

```
    // closing resources
```

```
    br.close();
```

```
    pw.close();
```

```
        System.out.println("Merged file1.txt  
and file2.txt into file3.txt");
```

```
    }
```

```
}
```

Lab Assessment-10

Q)

```
import java.util.*;
```

```
class Student {
```

```
    String name;
```

```
    String stu_id;
```

```
    int score;
```

```
    public Student() {
```

```
        this(" ", " ", 0);
```

```
    }
```

```
    public Student(String initName, String initId,  
int initScore) {
```

```
        name = initName;

        stu_id = initId;

        score = initScore;

    }

}
```

```
public class Main {

    public static void main(String[] args) {

        Scanner in = new Scanner(System.in);

        System.out.println("Input number of
students:");

        int n =

Integer.parseInt(in.nextLine().trim());

        System.out.println("Input Student
Name, ID, Score:");

    }

}
```

```
Student stu = new Student();  
Student max = new Student();  
Student min = new Student(" ", " ",  
100);  
  
for (int i = 0; i < n; i++) {  
    stu.name = in.next();  
    stu.stu_id = in.next();  
    stu.score = in.nextInt();  
    if (max.score < stu.score) {  
        max.name = stu.name;  
        max.stu_id = stu.stu_id;  
        max.score = stu.score;  
    }  
    if (min.score > stu.score) {  
        min.name = stu.name;
```

```
        min.stu_id = stu.stu_id;
        min.score = stu.score;
    }

}

    System.out.println("name, ID of the
highest score and the lowest score:");

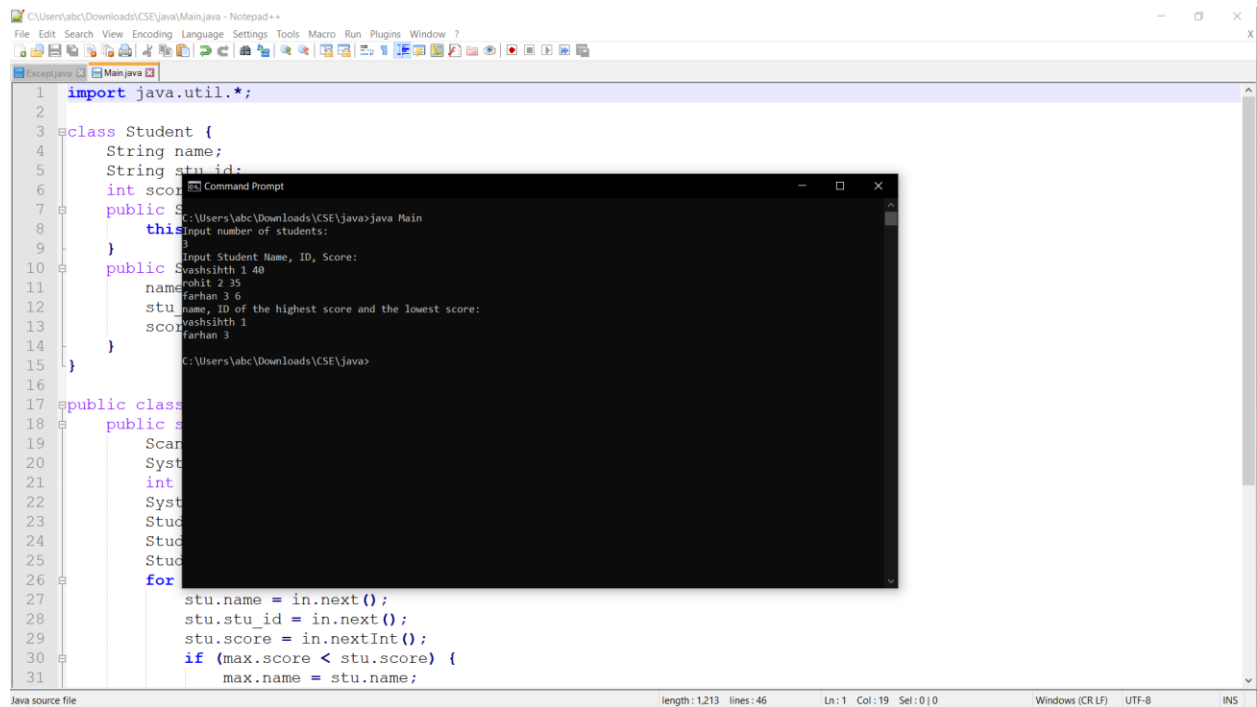
    System.out.println(max.name + " " +
max.stu_id);

    System.out.println(min.name + " " +
min.stu_id);

    in.close();

}

}
```



The screenshot shows a Notepad++ window with a Java file named 'Main.java'. The code defines a 'Student' class with attributes 'name', 'stu_id', and 'score'. It includes a constructor and a 'main' method. The 'main' method uses a 'Scanner' to take input from the user. A 'Command Prompt' window is overlaid on the code, showing the execution of the program. The prompt asks for the number of students (3), then for each student's name, ID, and score. The input provided is: Svashith 1 40, rohit 2 35, and farhan 3 6. The program then outputs the highest and lowest scores: 'Svashith 1' for the highest score and 'farhan 3' for the lowest score.

```
1 import java.util.*;
2
3 class Student {
4     String name;
5     String stu_id;
6     int score;
7     public Student() {}
8     public Student(String name, int stu_id, int score) {
9         this.name = name;
10        this.stu_id = stu_id;
11        this.score = score;
12    }
13    public static void main(String[] args) {
14        Scanner in = new Scanner(System.in);
15        System.out.println("Input number of Students:");
16        int n = in.nextInt();
17        Student[] stu = new Student[n];
18        for (int i = 0; i < n; i++) {
19            System.out.println("Input Student Name, ID, Score:");
20            stu[i] = new Student(in.next(), in.nextInt(), in.nextInt());
21        }
22        Student max = stu[0];
23        Student min = stu[0];
24        for (int i = 1; i < n; i++) {
25            if (max.score < stu[i].score) {
26                max = stu[i];
27            }
28            if (min.score > stu[i].score) {
29                min = stu[i];
30            }
31        }
32        System.out.println("Student with highest score: " + max.name + " " + max.stu_id + " " + max.score);
33        System.out.println("Student with lowest score: " + min.name + " " + min.stu_id + " " + min.score);
34    }
35 }
```

Command Prompt Output:

```
C:\Users\abc\Downloads\CSE\java>java Main
Input number of Students:
3
Input Student Name, ID, Score:
Svashith 1 40
rohit 2 35
farhan 3 6
Student with highest score: Svashith 1 40
Student with lowest score: farhan 3 6
C:\Users\abc\Downloads\CSE\java>
```

Q)

```
package serializationdemo;
```

```
import java.io.*;
```

```
public class EmployeeSerialDemo {
```

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

```
        Employee c = new Employee("Suresh",  
        "E123"); // 2
```

```
        File outFile = new File("empSerial.ser");
```

```
try {  
    FileOutputStream fs = new  
FileOutputStream(outFile);  
    ObjectOutputStream os = new  
ObjectOutputStream(fs);  
    os.writeObject(c); // 3  
    os.close();  
} catch (Exception e) {  
    e.printStackTrace();  
}  
}  
}
```

Lab Assessment-11

Q)

```
import java.sql.*;

class question1 {

    private static String user ="root";

    private static String password = "root";

    private static Connection con=null;

    private static Statement st=null;

    private static ResultSet rs=null;

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

        try {

            Class.forName("com.mysql.cj.jdbc.Driver");
```



```
con=DriverManager.getConnection("jdbc:mysql://localhost", user, password);

    System.out.println("Success");
}

catch (Exception e) {

    System.out.println(e);
}

try{

    st=con.createStatement();

    st.addBatch("CREATE DATABASE Faculty;");

    st.addBatch("USE Faculty;");

    st.addBatch("CREATE TABLE Faculty1(empid
VARCHAR(4),facultyname VARCHAR(15),DOB
DATE,Date_of_Joining DATE,designation
VARCHAR(20));");

    st.addBatch("INSERT INTO Faculty1 VALUES"+
```

```
"('E101','Ravi','1975-02-24','1992-04-27','Associate Professor'),"+
```

```
"('E102','Rahul','1975-02-24','1992-04-27','Senior Professor'),"+
```

```
"('E103','Raman','1975-02-24','1992-04-27','Assistant Professor'),"+
```

```
"('E104','Krishna','1975-02-24','1992-04-27','Senior Professor'),"+
```

```
"('E105','Raghav','1975-02-24','1992-04-27','Head of Department');");
```

```
st.executeBatch();
```

```
rs=st.executeQuery("select * from Faculty1;");
```

```
System.out.println("EmpID\tFaculty  
Name\tDate Of Birth\tDate of Joining\t  
Designation");
```

```
while(rs.next())
```

```
System.out.println(rs.getString("empid")+"\t\t"+rs.g  
etString("facultyname")+"\t\t"+rs.getString("DOB")  
+"\t\t"+rs.getString("Date_of_Joining")+"\t\t"+rs.ge  
tString("designation"));
```

```
System.out.println("\n\n\n\n");
```

```
rs=st.executeQuery(" select * from Faculty1  
where designation='Senior Professor'; ");
```

```
        System.out.println("EmpID\tFaculty  
Name\tDate Of Birth\tDate of Joining\t  
Designation");
```

```
        while(rs.next())
```

```
        System.out.println(rs.getString("empid")+"\t\t"+rs.g  
etString("facultyname")+"\t\t"+rs.getString("DOB")  
+"\t\t"+rs.getString("Date_of_Joining")+"\t\t"+rs.ge  
tString("designation"));
```

```
    }
```

```
    finally {
```

```
        try {
```

```
            if (rs != null) rs.close();
```

```
            if (st != null) st.close();
```

```
            if (con != null) con.close();
```

```
        }
```

```
        catch (SQLException e){  
            System.out.println(e);  
        }  
    }  
}  
}
```

Lab Assessment-12

Q)

Index.html

Name:

Password:

Index.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class Validate extends HttpServlet {
    protected void doPost(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-
8");
        PrintWriter out = response.getWriter();
        try {
            String name = request.getParameter("user");
            String password = request.getParameter("pass");
            if(password.equals("studytonight"))
            {
                RequestDispatcher rd =
request.getRequestDispatcher("Welcome");
                rd.forward(request, response);
            }
        }
    }
}
```

```

else
{
    out.println("<font color='red'><b>You have entered
incorrect password</b></font>");
    RequestDispatcher rd =
request.getRequestDispatcher("index.html");
    rd.include(request, response);
}
}finally {
    out.close();
}

```

Home.java

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class Welcome extends HttpServlet { protected
void doPost(HttpServletRequest request,
HttpServletResponse response) throws ServletException,
IOException {
response.setContentType("text/html;charset=UTF-8");
PrintWriter out = response.getWriter();
try { out.println("

```

Welcome user

```
"); } finally { out.close(); } } }
```

Web.xml

Validate Validate Welcome Welcome Validate /Validate
Welcome /Welcome index.html