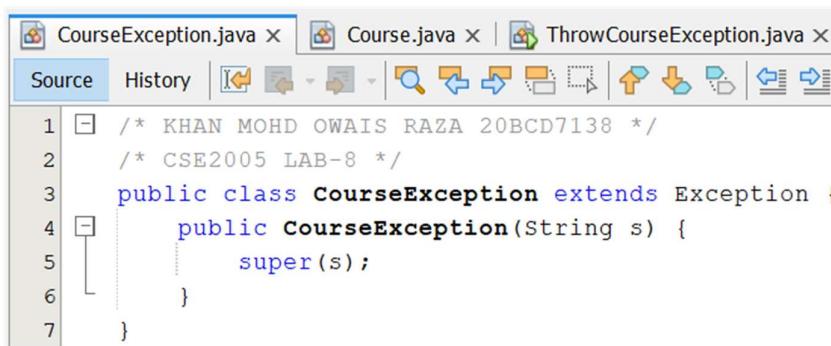


**KHAN MOHD. OWAIS RAZA
20BCD7138**

Q.1] a) Create a Checked CourseException class and whose constructor receives a String that holds a college course's department (for example, CIS), a course number (for example, 101), and a number of credits (for example, 3). Save the file as CourseException.java. Create a Course class with the same fields and whose constructor requires values for each field. Upon construction, throw a CourseException if the department does not consist of three letters, if the course number does not consist of three digits between 100 and 499 inclusive, or if the credits are less than 1 or more than 6. Save the class as Course.java. Write an application that establishes an array of at least six Course objects with valid and invalid values. Display an appropriate message when a Course object is created successfully and when one is not. Save the file as ThrowCourseException.java.

CourseException.java –



```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  public class CourseException extends Exception {
4      public CourseException(String s) {
5          super(s);
6      }
7  }
```

Course.java –

The screenshot shows a Java development environment with three tabs at the top: "CourseException.java x", "Course.java x", and "ThrowCourseException.java x". The "Course.java" tab is active. Below the tabs is a toolbar with various icons for file operations like new, open, save, and search. The main area displays the code for the "Course" class.

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8*/
3  public class Course {
4      String department;
5      int courseNumber;
6      double credits;
7      final int DEPT_LENGTH = 3;
8      final int LOW_NUM = 100;
9      final int HIGH_NUM = 499;
10     final double LOW_CREDITS = 0.5;
11     final double HIGH_CREDITS = 6;
12     public Course() {
13         department="";
14         courseNumber=0;
15         credits=0;
16     }
17     public Course(String dept,int num,double creditValue)
18         throws CourseException
19     {
20         boolean isBad = false;
21         String message = "";
22         department = dept;
23         courseNumber=num;
24         credits = creditValue;
25         if (dept.length()!=DEPT_LENGTH){
26             message += "\nThe Course code number must be 3 numerals ";
27         }
28         if (num<LOW_NUM|| num> HIGH_NUM){
29             message+="Course number should be between 100 and 499";
30         }
31         if (creditValue<LOW_CREDITS|| creditValue> HIGH_CREDITS){
32             message+="\nCredit must be between 1 or 6";
33         }
34         if (!message.equals("")){
35             throw(new CourseException(message+
36                                         "\nDepartment: "
37                                         +department+
38                                         "\nCourse number: "
39                                         + courseNumber+
40                                         "\nCredits: "
41                                         +credits));
42         }
43     }
44     public String toString() {
45         return "\nDepartment: "
46                     +department+
47                     "\nCourse number: "
48                     + courseNumber+
49                     "\nCredits: "
50                     +credits;
51     }
}
```

ThrowCourseException.java –

The screenshot shows a Java IDE interface with three tabs at the top: "CourseException.java", "Course.java", and "ThrowCourseException.java". The "ThrowCourseException.java" tab is active. Below the tabs is a toolbar with various icons. The main area displays the source code for the "ThrowCourseException" class.

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  public class ThrowCourseException {
4      public static void main(String[] args) {
5          var courses = new Course[14];
6          System.out.println("WELCOME TO MID-TOWN HIGH JUNIOR COLLEGE, NEW YORK");
7          System.out.println("-----");
8          System.out.println("-----");
9          System.out.println("-----");
10         System.out.println("-----");
11         System.out.println("-----");
12         System.out.println("-----");
13         System.out.println("-----");
14         System.out.println("-----");
15         System.out.println("-----");
16         System.out.println("-----");
17         System.out.println("-----");
18         System.out.println("-----");
19         System.out.println("-----");
20         System.out.println("-----");
21         System.out.println("-----");
22         System.out.println("-----");
23         System.out.println("-----");
24         System.out.println("-----");
25         System.out.println("-----");
26         System.out.println("-----");
27         System.out.println("-----");
28         System.out.println("-----");
29         String[] crsDepartments = {"MAT", "PHY",
30                                     "ELE1", "ELE2",
31                                     "CHY1", "CHY2",
32                                     "ENG", "EVS",
33                                     "EMS", "EMD",
34                                     "QMEC", "THM",
35                                     "CS", "FME"};
36         int[] crsNumbers = {101, 204,
37                            306, 200,
38                            303, 601,
39                            112, 231,
40                            229, 402,
41                            403, 209,
42                            319, 200};
43         double[] crsCredits = {1, 2, 3, 4, 4, 5, 6, 4, 5, 4, 4, 3, 5, 2, 5};
44         int x;
45         for(x = 0; x < courses.length; ++x)
46             courses[x] = new Course();
47         for(x = 0; x < courses.length; ++x)
48         {
49             try
50             {
51                 courses[x] = new Course(crsDepartments[x],
52                                         crsNumbers[x], crsCredits[x]);
53             }
54             catch(CourseException error)
55             {
56             }
57         }
58     }
59 }
```

```

56     System.out.println("Course: \n" + x +
57             " Error: \n" + error.getMessage());
58 }
59 }
60 System.out.println("\nCourse values - ");
61 for(x = 0; x < courses.length; ++x)
62     System.out.println(courses[x].toString());
63 }
64 }

```

Output Console -

Output - Run (ThrowCourseException) - Editor

Output - Run (ThrowCourseException) X

cd C:\Users\Owais\Documents\NetBeansProjects\Lab8; "JAVA_HOME=C:\\Program Files\\Java\\\"
Running NetBeans Compile On Save execution. Phase execution is skipped and output direct
Scanning for projects...

-----< com.ksvmediaplayer:Lab8 >-----

Building Lab8 1.0-SNAPSHOT [jar]

--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab8 ---
WELCOME TO MID-TOWN HIGH JUNIOR COLLEGE, NEW YORK

| Course Name | Course Code |
|----------------------------------|-------------|
| Mathematics & Statistics | MAT |
| Physics | PHY |
| Electronics Paper-1 | ELE1 |
| Electronics Paper-2 | ELE2 |
| Chemistry Part-1 | CHY1 |
| Chemistry Part-2 | CHY2 |
| English | ENG |
| Environmental Studies | EVS |
| Engineering Mechanics (Statics) | EMS |
| Engineering Mechanics (Dynamics) | EMD |
| Quantum Mechanics | QMEC |
| Engineering Thermodynamics | THM |
| Computer Science | CS |
| Fluid Mechanics | FME |

< >

Output - Run (ThrowCourseException) - Editor

Output - Run (ThrowCourseException) X

| | | |
|------------------|--|-----|
| Computer Science | | CS |
| Fluid Mechanics | | FME |

Course:
2 Error:

The Course code number must be 3 numerals
Department: ELE1
Course number: 306
Credits: 3.0
Course:
3 Error:

The Course code number must be 3 numerals
Department: ELE2
Course number: 200
Credits: 4.0
Course:
4 Error:

The Course code number must be 3 numerals
Department: CHY1
Course number: 303
Credits: 5.0
Course:
5 Error:

The Course code number must be 3 numerals
Course number should be between 100 and 499
Department: CHY2
Course number: 601

Output - Run (ThrowCourseException) - Editor

Output - Run (ThrowCourseException) X

Department: CHY2
Course number: 601
Credits: 6.0
Course:
10 Error:

The Course code number must be 3 numerals
Department: QMEC
Course number: 403
Credits: 3.0
Course:
12 Error:

The Course code number must be 3 numerals
Department: CS
Course number: 319
Credits: 2.0

Course values -

Department: MAT
Course number: 101
Credits: 1.0

Department: PHY
Course number: 204
Credits: 2.0

Department:
Course number: 0
Credits: 0.0

Output - Run (ThrowCourseException) - Editor

Output - Run (ThrowCourseException) x

```
Department:  
Course number: 0  
Credits: 0.0  
  
Department:  
Course number: 0  
Credits: 0.0  
  
Department:  
Course number: 0  
Credits: 0.0  
  
Department: ENG  
Course number: 112  
Credits: 4.0  
  
Department: EVS  
Course number: 231  
Credits: 5.0  
  
Department: EMS  
Course number: 229  
Credits: 4.0  
  
Department: EMD  
Course number: 402  
Credits: 4.0  
  
Department:  
Course number: 0  
Credits: 0.0
```

Output - Run (ThrowCourseException) - Editor

Output - Run (ThrowCourseException) x

```
Credits: 5.0  
  
Department: EMS  
Course number: 229  
Credits: 4.0  
  
Department: EMD  
Course number: 402  
Credits: 4.0  
  
Department:  
Course number: 0  
Credits: 0.0  
  
Department: THM  
Course number: 209  
Credits: 5.0  
  
Department:  
Course number: 0  
Credits: 0.0  
  
Department: FME  
Course number: 200  
Credits: 5.0
```

```
BUILD SUCCESS
```

```
Total time: 1.775 s  
Finished at: 2021-12-13T18:32:49+05:30
```

Q.1] b) Now modify the above CourseException class to display different message based on the cause of the exception as shown below :

| Different Message to be displayed based on the cause, while exception is caught |
|---|
| Course Department should be a Three Letter String |
| Course Number should be 3 digits Numeric |
| Course Credits should be in the range 1 to 6. |

CourseException.java –



```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  public class CourseException extends Exception {
4      public CourseException(String s) {
5          super(s);
6      }
7  }
```

Course.java –



The screenshot shows a Java IDE interface with three tabs at the top: "CourseException.java x", "Course.java x", and "ThrowCourseException.java x". The "Course.java" tab is active. Below the tabs is a toolbar with various icons. The main area displays the source code for the "Course" class.

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8*/
3  public class Course {
4      String department;
5      int courseNumber;
6      double credits;
7      final int DEPT_LENGTH = 3;
8      final int LOW_NUM = 100;
9      final int HIGH_NUM = 499;
10     final double LOW_CREDITS = 1;
11     final double HIGH_CREDITS = 6;
12     public Course() {
13         department="";
14         courseNumber=0;
15         credits=0;
16     }
17     public Course(String dept,int num,double creditValue)
18         throws CourseException
19     {
20         boolean isBad = false;
21         String message = "";
22         department = dept;
23         courseNumber=num;
24         credits = creditValue;
25         if (dept.length()!=DEPT_LENGTH){
26             message += "\nThe Course code number must be 3 numerals ";
27         }
28         if (num > LOW_NUM || num < HIGH_NUM){
29             message+="Course number should be between 100 and 499";
30         }
31         if (creditValue > LOW_CREDITS || creditValue < HIGH_CREDITS){
32             message+="\nCredit must be between 1 or 6";
33         }
34         if(!message.equals("")){
35             throw(new CourseException(message+"\nDepartment: "
36                                     +department+
37                                     "\nCourse number: "
38                                     + courseNumber+
39                                     "\nCredits: "
40                                     +credits));
41         }
42     }
43     public String toString() {
44         return "\nDepartment: "
45                     +department+
46                     "\nCourse number: "
47                     + courseNumber+
48                     "\nCredits: "
49                     +credits;
50     }
51 }
```

ThrowCourseException.java –

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  public class ThrowCourseException {
4      public static void main(String[] args) {
5          var courses = new Course[14];
6          System.out.println("WELCOME TO MID-TOWN HIGH JUNIOR COLLEGE, NEW YORK");
7          System.out.println("-----");
8          System.out.println("-----");
9          System.out.println("-----");
10         System.out.println("-----");
11         System.out.println("-----");
12         System.out.println("-----");
13         System.out.println("-----");
14         System.out.println("-----");
15         System.out.println("-----");
16         System.out.println("-----");
17         System.out.println("-----");
18         System.out.println("-----");
19         System.out.println("-----");
20         System.out.println("-----");
21         System.out.println("-----");
22         System.out.println("-----");
23         System.out.println("-----");
24         System.out.println("-----");
25         System.out.println("-----");
26         System.out.println("-----");
27         System.out.println("-----");
28         System.out.println("-----");
29         String[] crsDepartments = {"MAT", "PHY",
30             "ELE1", "ELE2",
31             "CHY1", "CHY2",
32             "ENG", "EVS",
33             "EMS", "EMD",
34             "QMEC", "THM",
35             "CS", "FME"};
36         int[] crsNumbers = {101, 204,
37             306, 200,
38             303, 601,
39             112, 231,
40             229, 402,
41             403, 209,
42             319, 200};
43         double[] crsCredits = {1, 2, 3, 4, 5, 6, 6, 5, 4, 3, 2, 1, 2, 3};
44         int x;
45         for(x = 0; x < courses.length; ++x)
46             courses[x] = new Course();
47         for(x = 0; x < courses.length; ++x)
48         {
49             try
50             {
51                 courses[x] = new Course(crsDepartments[x],
52                     crsNumbers[x], crsCredits[x]);
53             }
54             catch(CourseException error)
55             {
56                 System.out.println("Course: \n" + x + "
```

```

57     }
58     }
59     }
60     System.out.println("\nCourse values - ");
61     for(x = 0; x < courses.length; ++x)
62     {
63         System.out.println(courses[x].toString());
64     }

```

Output console –

Output - Run (ThrowCourseException) - Editor

Output - Run (ThrowCourseException) X

cd C:\Users\Owais\Documents\NetBeansProjects\Lab8; "JAVA_HOME=C:\\Program Files\\Java\\jdk-17.0.1" Running NetBeans Compile On Save execution. Phase execution is skipped and output directories are up-to-date. Scanning for projects...

--< com.ksvmediaplayer:Lab8 >-----

[Building Lab8 1.0-SNAPSHOT]-----

[jar]-----

--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab8 ---

WELCOME TO MID-TOWN HIGH JUNIOR COLLEGE, NEW YORK

| Course Name | Course Code |
|----------------------------------|-------------|
| Mathematics & Statistics | MAT |
| Physics | PHY |
| Electronics Paper-1 | ELE1 |
| Electronics Paper-2 | ELE2 |
| Chemistry Part-1 | CHY1 |
| Chemistry Part-2 | CHY2 |
| English | ENG |
| Environmental Studies | EVS |
| Engineering Mechanics (Statics) | EMS |
| Engineering Mechanics (Dynamics) | EMD |
| Quantum Mechanics | QMEC |
| Engineering Thermodynamics | THM |
| Computer Science | CS |
| Fluid Mechanics | FME |

Output - Run (ThrowCourseException) - Editor

Output - Run (ThrowCourseException) | FME

```
Fluid Mechanics
-----
Course:
0 Error:

Course number should be between 100 and 499
Credit must be between 1 or 6
Department: MAT
Course number: 101
Credits: 1.0
Course:
1 Error:

Course number should be between 100 and 499
Credit must be between 1 or 6
Department: PHY
Course number: 204
Credits: 2.0
Course:
2 Error:

The Course code number must be 3 numerals
Course number should be between 100 and 499
Credit must be between 1 or 6
Department: ELE1
Course number: 306
Credits: 3.0
Course:
3 Error:

mbo Course code number must be 3 numerals
```

Output - Run (ThrowCourseException) - Editor

Output - Run (ThrowCourseException) | FME

```
Course:
3 Error:

The Course code number must be 3 numerals
Course number should be between 100 and 499
Credit must be between 1 or 6
Department: ELE2
Course number: 200
Credits: 4.0
Course:
4 Error:

The Course code number must be 3 numerals
Course number should be between 100 and 499
Credit must be between 1 or 6
Department: CHY1
Course number: 303
Credits: 5.0
Course:
5 Error:

The Course code number must be 3 numerals
Course number should be between 100 and 499
Credit must be between 1 or 6
Department: CHY2
Course number: 601
Credits: 6.0
Course:
6 Error:

Course number should be between 100 and 499
Credit must be between 1 or 6
```

Output - Run (ThrowCourseException) - Editor

Output - Run (ThrowCourseException) x

```
Course number should be between 100 and 499
Credit must be between 1 or 6
Department: ENG
Course number: 112
Credits: 6.0
Course:
7 Error:

Course number should be between 100 and 499
Credit must be between 1 or 6
Department: EVS
Course number: 231
Credits: 5.0
Course:
8 Error:

Course number should be between 100 and 499
Credit must be between 1 or 6
Department: EMS
Course number: 229
Credits: 4.0
Course:
9 Error:

Course number should be between 100 and 499
Credit must be between 1 or 6
Department: EMD
Course number: 402
Credits: 3.0
Course:
10 Error:
```

Output - Run (ThrowCourseException) - Editor

Output - Run (ThrowCourseException) x

```
The Course code number must be 3 numerals
Course number should be between 100 and 499
Credit must be between 1 or 6
Department: QMEC
Course number: 403
Credits: 2.0
Course:
11 Error:

Course number should be between 100 and 499
Credit must be between 1 or 6
Department: THM
Course number: 209
Credits: 1.0
Course:
12 Error:

The Course code number must be 3 numerals
Course number should be between 100 and 499
Credit must be between 1 or 6
Department: CS
Course number: 319
Credits: 2.0
Course:
13 Error:

Course number should be between 100 and 499
Credit must be between 1 or 6
Department: FME
Course number: 200
Credits: 3.0
```

```
Output - Run (ThrowCourseException) - Editor
Output - Run (ThrowCourseException) x

Course values - 

Department:
Course number: 0
Credits: 0.0

Department:
```

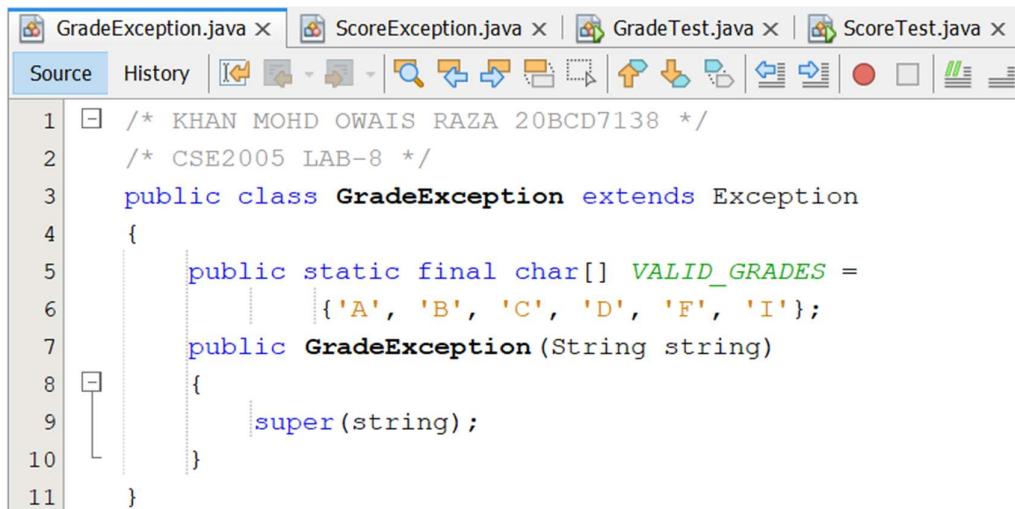
```
Output - Run (ThrowCourseException) - Editor
Output - Run (ThrowCourseException) x

Department:
Course number: 0
Credits: 0.0

-----
BUILD SUCCESS
-----
Total time: 1.822 s
Finished at: 2021-12-13T21:24:44+05:30
-----
```

Q.2] a) Develop a ScoreGradeTest class to display a series of five Student ID numbers and asks the user to enter a numeric test scores of three subjects and a grade.
 Develop a checked ScoreException class and throw it when the user enters an invalid score (greater than 100 and less than 0).
 Develop a checked GradeException class and throw it when the user enters an invalid grade. (Other than A, B, C, D, F).
 Also throw GradeException if the user enters wrong grade for their average test score.
 Differentiate these two GradeException types with appropriate messages.

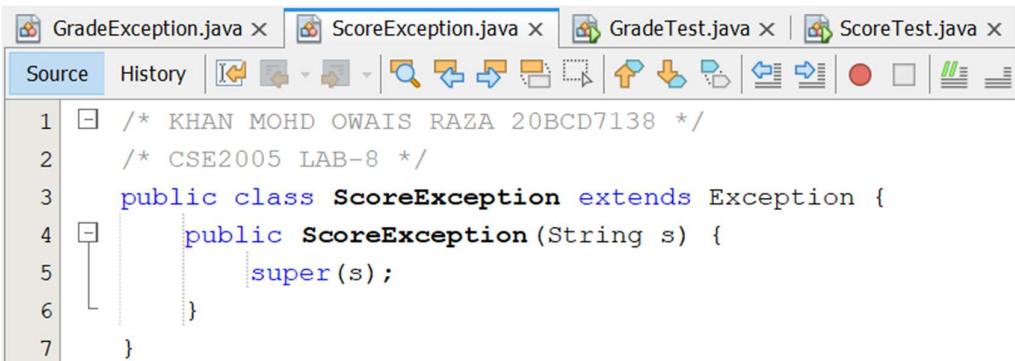
GradeException.java –



```

1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  public class GradeException extends Exception
4  {
5      public static final char[] VALID_GRADES =
6          {'A', 'B', 'C', 'D', 'F', 'I'};
7      public GradeException(String string)
8      {
9          super(string);
10     }
11 }
```

ScoreException.java –



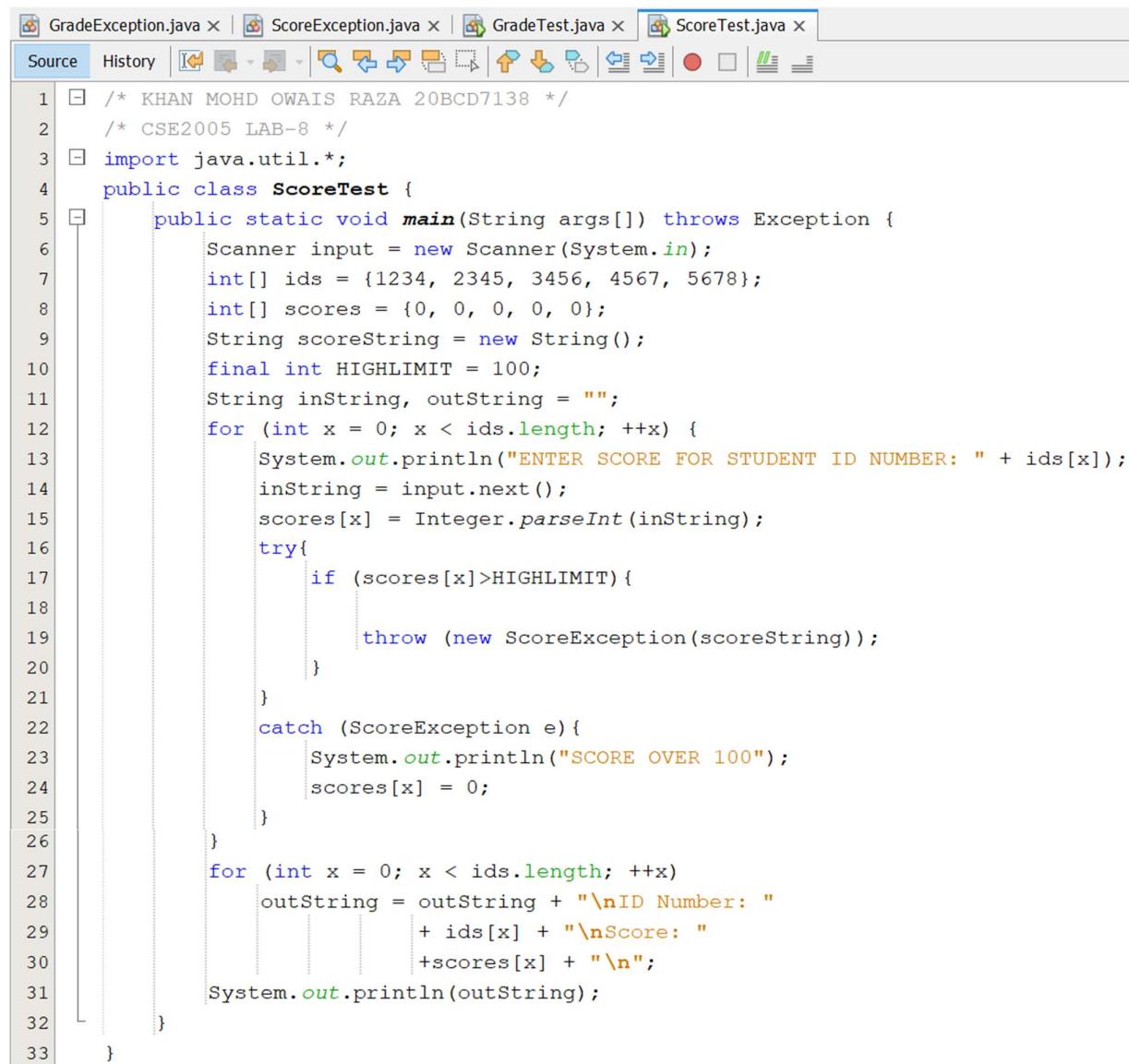
```

1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  public class ScoreException extends Exception {
4      public ScoreException(String s) {
5          super(s);
6      }
7 }
```

GradeTest.java –

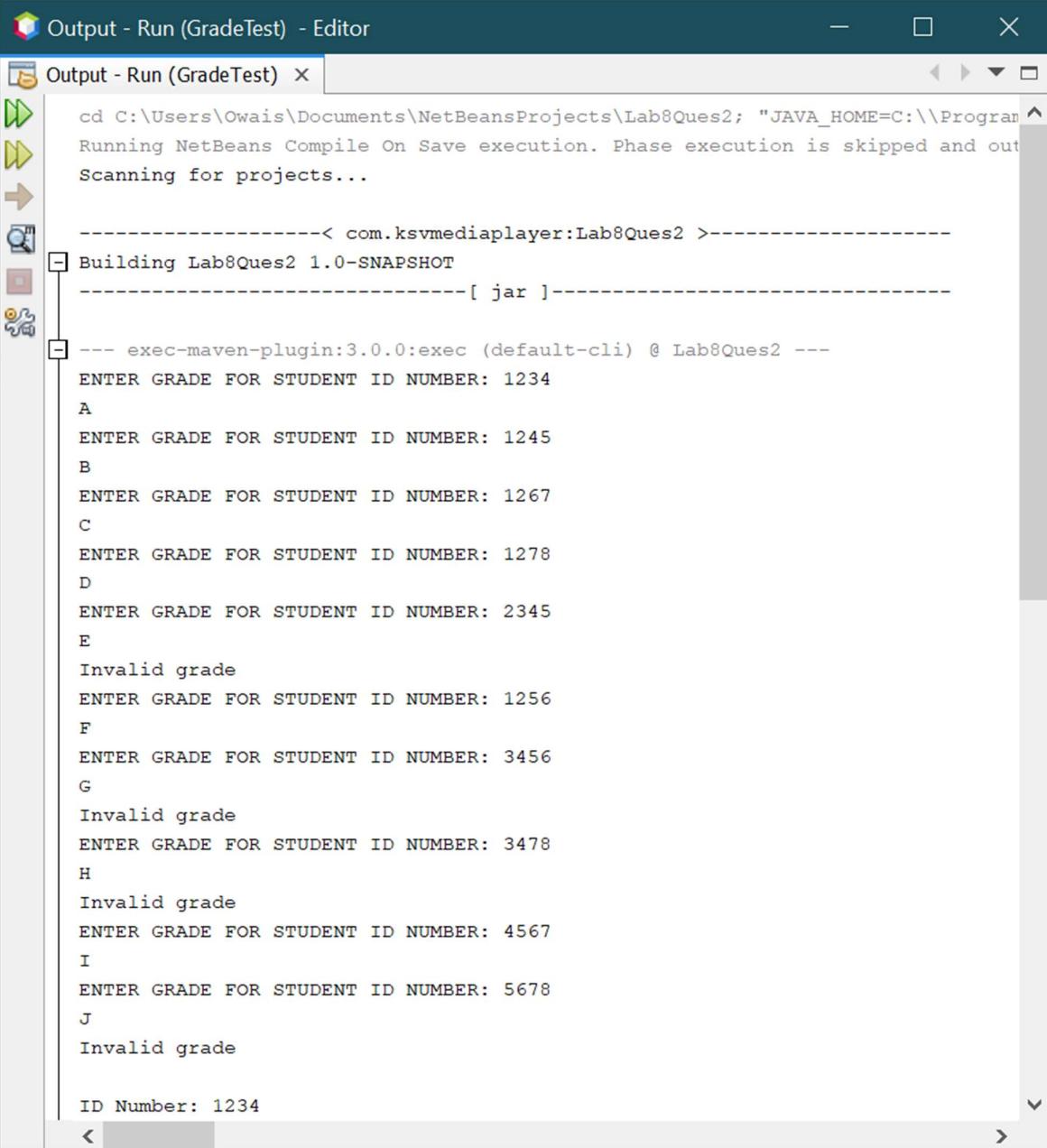
```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  import java.util.*;
4  public class GradeTest {
5      public static void main(String args[]) throws Exception {
6          Scanner input = new Scanner(System.in);
7          int[] ids = {1234, 1245, 1267, 1278, 2345,
8              1256, 3456, 3478, 4567, 5678
9          };
10         char[] grades = new char[10];
11         String gradeString = new String();
12         final int HIGHLIMIT = 100;
13         String inString, outString = "";
14         int flag = 0;
15         for (int x = 0; x < ids.length; ++x) {
16             System.out.println("ENTER GRADE FOR STUDENT ID NUMBER: "
17                 + ids[x]);
18             inString = input.next();
19             grades[x] = inString.charAt(0);
20             try
21             {
22                 flag = 0;
23                 for(int y = 0; y < GradeException.VALID_GRADES.length; ++y)
24                 {
25                     if(grades[x] == GradeException.VALID_GRADES[y])
26                         flag = 1;
27                 }
28                 if(flag == 0)
29                 {
30                     gradeString = "Invalid grade";
31                     throw(new GradeException(gradeString));
32                 }
33             }
34             catch(GradeException e)
35             {
36                 System.out.println(e.getMessage());
37                 grades[x] = 'I';
38             }
39         }
40         for (int x = 0; x < ids.length; ++x)
41             outString = outString + "\nID Number: "
42                 + ids[x] + "\nGrade: "
43                 + grades[x] + "\n";
44         System.out.println(outString);
45     }
46 }
47 }
```

ScoreTest.java –



```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  import java.util.*;
4  public class ScoreTest {
5      public static void main(String args[]) throws Exception {
6          Scanner input = new Scanner(System.in);
7          int[] ids = {1234, 2345, 3456, 4567, 5678};
8          int[] scores = {0, 0, 0, 0, 0};
9          String scoreString = new String();
10         final int HIGHLIMIT = 100;
11         String inString, outString = "";
12         for (int x = 0; x < ids.length; ++x) {
13             System.out.println("ENTER SCORE FOR STUDENT ID NUMBER: " + ids[x]);
14             inString = input.nextLine();
15             scores[x] = Integer.parseInt(inString);
16             try{
17                 if (scores[x]>HIGHLIMIT){
18                     throw (new ScoreException(scoreString));
19                 }
20             }
21             catch (ScoreException e){
22                 System.out.println("SCORE OVER 100");
23                 scores[x] = 0;
24             }
25         }
26     }
27     for (int x = 0; x < ids.length; ++x)
28         outString = outString + "\nID Number: "
29                     + ids[x] + "\nScore: "
30                     + scores[x] + "\n";
31     System.out.println(outString);
32 }
33 }
```

Output console (GradeTest.java) –



The screenshot shows the NetBeans IDE's Output window titled "Output - Run (GradeTest) - Editor". The window displays the command-line interface of a Java application named GradeTest. The application prompts the user to enter student ID numbers and their corresponding grades. The user inputs various grade letters (A through J) and some invalid entries like "ID Number: 1234". The application processes each input and prints a response. The background of the window is white, and the text is black. There are some gray UI elements like scroll bars and toolbars.

```
cd C:\Users\Owais\Documents\NetBeansProjects\Lab8Ques2; "JAVA_HOME=C:\\Program Files\\Java\\jdk-11.0.1\\bin" java -jar Lab8Ques2-1.0-SNAPSHOT.jar
Running NetBeans Compile On Save execution. Phase execution is skipped and output is not generated.
Scanning for projects...
-----< com.ksvmediaplayer:Lab8Ques2 >-----
Building Lab8Ques2 1.0-SNAPSHOT
-----[ jar ]-----
--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab8Ques2 ---
ENTER GRADE FOR STUDENT ID NUMBER: 1234
A
ENTER GRADE FOR STUDENT ID NUMBER: 1245
B
ENTER GRADE FOR STUDENT ID NUMBER: 1267
C
ENTER GRADE FOR STUDENT ID NUMBER: 1278
D
ENTER GRADE FOR STUDENT ID NUMBER: 2345
E
Invalid grade
ENTER GRADE FOR STUDENT ID NUMBER: 1256
F
ENTER GRADE FOR STUDENT ID NUMBER: 3456
G
Invalid grade
ENTER GRADE FOR STUDENT ID NUMBER: 3478
H
Invalid grade
ENTER GRADE FOR STUDENT ID NUMBER: 4567
I
ENTER GRADE FOR STUDENT ID NUMBER: 5678
J
Invalid grade

ID Number: 1234
```

Output - Run (GradeTest) - Editor

Output - Run (GradeTest) X

The screenshot shows a software window titled "Output - Run (GradeTest) - Editor". On the left is a vertical toolbar with icons for file operations (New, Open, Save, Print, Find, Copy, Paste, Cut, Undo, Redo). A tab bar at the top contains "Output - Run (GradeTest) X". The main area displays a list of student records:

- ID Number: 1234
Grade: A
- ID Number: 1245
Grade: B
- ID Number: 1267
Grade: C
- ID Number: 1278
Grade: D
- ID Number: 2345
Grade: I
- ID Number: 1256
Grade: F
- ID Number: 3456
Grade: I
- ID Number: 3478
Grade: I
- ID Number: 4567
Grade: I
- ID Number: 5678
Grade: I

Below the records, there is a dashed line followed by the text "BUILD SUCCESS" in green, another dashed line, and then the build statistics:

Total time: 30.995 s
Finished at: 2021-12-14T00:16:57+05:30

Output console (ScoreTest.java) –

```
Output - Run (ScoreTest) - Editor
Output - Run (ScoreTest) X
cd C:\Users\Owais\Documents\NetBeansProjects\Lab8Ques2; "JAVA_HOME=C:\\\\Pr
Running NetBeans Compile On Save execution. Phase execution is skipped and
Scanning for projects...

-----< com.ksvmediaplayer:Lab8Ques2 >-----
Building Lab8Ques2 1.0-SNAPSHOT
-----[ jar ]-----
--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab8Ques2 ---
ENTER SCORE FOR STUDENT ID NUMBER: 1234
75
ENTER SCORE FOR STUDENT ID NUMBER: 2345
86
ENTER SCORE FOR STUDENT ID NUMBER: 3456
78
ENTER SCORE FOR STUDENT ID NUMBER: 4567
98
ENTER SCORE FOR STUDENT ID NUMBER: 5678
94

ID Number: 1234
Score: 75

ID Number: 2345
Score: 86

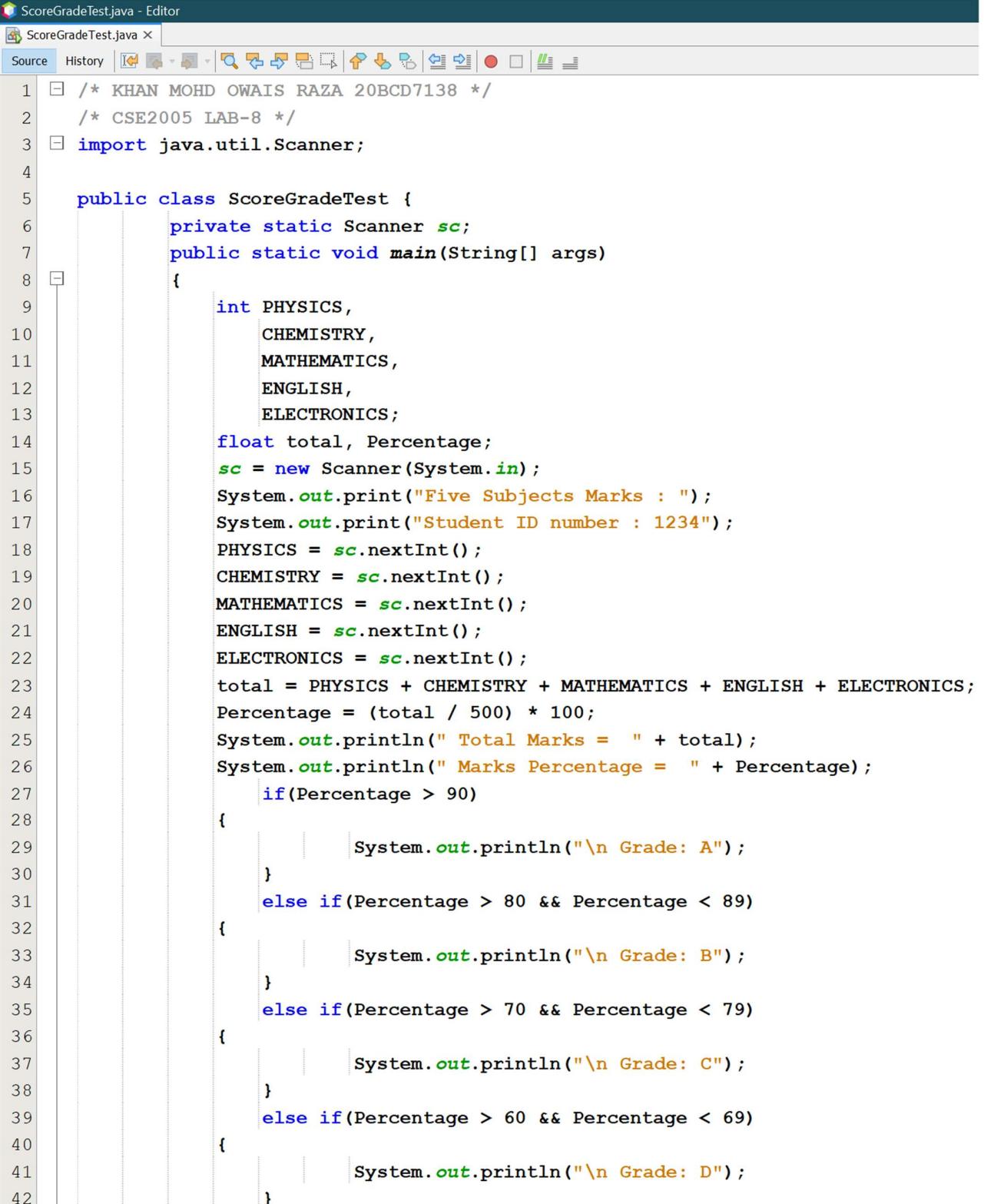
ID Number: 3456
Score: 78

ID Number: 4567
Score: 98

ID Number: 5678
Score: 94

-----
BUILD SUCCESS
-----
Total time: 16.570 s
Finished at: 2021-12-14T00:28:53+05:30
-----
```

ScoreGradeTest.java –



The screenshot shows a Java code editor window titled "ScoreGradeTest.java - Editor". The menu bar includes "File", "Edit", "View", "Tools", "Help", and "ScoreGradeTest.java". The toolbar contains icons for file operations like Open, Save, Print, and others. The code itself is a Java program that calculates the total marks and percentage of a student across five subjects (Physics, Chemistry, Mathematics, English, Electronics) and prints the grade based on the percentage.

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  import java.util.Scanner;
4
5  public class ScoreGradeTest {
6      private static Scanner sc;
7      public static void main(String[] args)
8      {
9          int PHYSICS,
10             CHEMISTRY,
11             MATHEMATICS,
12             ENGLISH,
13             ELECTRONICS;
14
15         float total, Percentage;
16         sc = new Scanner(System.in);
17         System.out.print("Five Subjects Marks : ");
18         System.out.print("Student ID number : 1234");
19         PHYSICS = sc.nextInt();
20         CHEMISTRY = sc.nextInt();
21         MATHEMATICS = sc.nextInt();
22         ENGLISH = sc.nextInt();
23         ELECTRONICS = sc.nextInt();
24         total = PHYSICS + CHEMISTRY + MATHEMATICS + ENGLISH + ELECTRONICS;
25         Percentage = (total / 500) * 100;
26         System.out.println(" Total Marks = " + total);
27         System.out.println(" Marks Percentage = " + Percentage);
28         if(Percentage > 90)
29         {
30             System.out.println("\n Grade: A");
31         }
32         else if(Percentage > 80 && Percentage < 89)
33         {
34             System.out.println("\n Grade: B");
35         }
36         else if(Percentage > 70 && Percentage < 79)
37         {
38             System.out.println("\n Grade: C");
39         }
40         else if(Percentage > 60 && Percentage < 69)
41         {
42             System.out.println("\n Grade: D");
43         }
44     }
45 }
```

```
43         else if(Percentage < 60)
44             {
45                 System.out.println("\n Grade: F");
46             }
47         else
48             {
49                 System.out.println("\n Fail");
50             }
51     }
52 }
```

Output Console (ScoreGradeTest.java) –

The screenshot shows the NetBeans IDE's Output window titled "Output - Run (ScoreGradeTest) - Editor". The window displays the command-line output of a Maven build for the project "Lab8Ques2". The output includes the Maven command, project details, build logs, and the execution results of the Java application.

```
cd C:\Users\Owais\Documents\NetBeansProjects\Lab8Ques2; "JAVA_HOME=C:\\Program
Running NetBeans Compile On Save execution. Phase execution is skipped and outp
Scanning for projects...

-----< com.ksvmediaplayer:Lab8Ques2 >-----
[-] Building Lab8Ques2 1.0-SNAPSHOT
-----[ jar ]-----

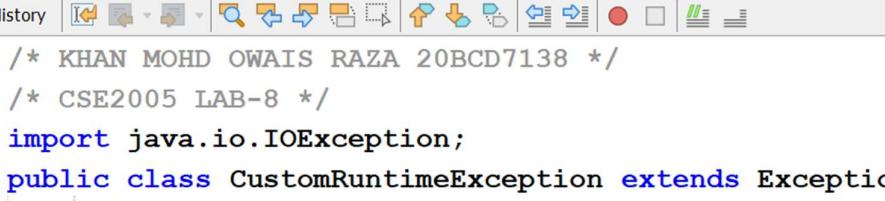
[-] --- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab8Ques2 ---
93
97
95
98
89
Five Subjects Marks : Student ID number : 1234 Total Marks = 472.0
Marks Percentage = 94.4

Grade: A

BUILD SUCCESS
-----
Total time: 23.261 s
Finished at: 2021-12-14T11:54:08+05:30
-----
```

Q.2] b) Convert Checked exceptions developed in (1b) and (2a) questions into Un-Checked exceptions. Write appropriate observations and differences that you identified after conversion.

Creating un-checked exceptions –



The screenshot shows a Java code editor window titled "CustomRuntimeException.java - Editor". The code is as follows:

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  import java.io.IOException;
4  public class CustomRuntimeException extends Exception {
5      public CustomRuntimeException(IOException ioe) {
6      }
7
8  }
```

The screenshot shows a Java code editor window titled "RuntimeException.java - Editor". The code implements a custom RuntimeException class that extends the built-in Exception class. It includes a constructor that takes a String parameter and calls the super constructor.

```
/* KHAN MOHD OWAIS RAZA 20BCD7138 */
/* CSE2005 LAB-8 */

public class RuntimeException extends Exception {
    public RuntimeException(String string) {
        super(string);
    }
}
```

Solution of Q.1] a) after creating un-checked exception –

CourseException.java –

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  public class CourseException extends Exception {
4      public CourseException(String s) {
5          super(s);
6      }
7 }
```

Course.java –

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8*/
3  public class Course {
4      String department;
5      int courseNumber;
6      double credits;
7      final int DEPT_LENGTH = 3;
8      final int LOW_NUM = 100;
9      final int HIGH_NUM = 499;
10     final double LOW_CREDITS = 0.5;
11     final double HIGH_CREDITS = 6;
12     public Course() {
13         department="";
14         courseNumber=0;
15         credits=0;
16     }
17
18     public Course(String dept, int num, double creditValue)
19             throws CourseException {
20         boolean isBad = false;
21         String message = "";
22         department = dept;
23         courseNumber=num;
24         credits = creditValue;
25         if (dept.length()!=DEPT_LENGTH) {
26             message += "The Course code number must be 3 numerals ";
27         }
28     }
}
```

```

29     if (num<LOW_NUM || num> HIGH_NUM) {
30         message+="Course number should be between 100 and 499";
31     }
32     if (creditValue<LOW_CREDITS|| creditValue> HIGH_CREDITS){
33         message+="Credit must be between 1 or 6";
34     }
35     if(!message.equals("")){
36         throw(new CourseException(message+"\nDepartment: "+department+
37                                     " Course number: " + courseNumber+
38                                     "Credits: "+credits));
39     }
40 }
41 }
42 }
43 }
44 }
45 public String toString() {
46     return "Department: " +department+
47             " Course number: " + courseNumber+
48             "Credits: "+credits;
49 }

```

ThrowCourseException.java –

CourseException.java x ThrowCourseException.java x Course.java x

Source History

```

1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  public class ThrowCourseException {
4      public static void main(String[] args) {
5          var courses = new Course[14];
6          System.out.println("WELCOME TO MID-TOWN HIGH JUNIOR COLLEGE, NEW YORK");
7          System.out.println("-----");
8          System.out.println("-----");
9          System.out.println("-----");
10         System.out.println("-----");
11         System.out.println("           Course Name           | Course Code ");
12         System.out.println("-----|-----");
13         System.out.println("Mathematics & Statistics   | MAT    ");
14         System.out.println("Physics                   | PHY    ");
15         System.out.println("Electronics Paper-1       | ELE1   ");
16         System.out.println("Electronics Paper-2       | ELE2   ");
17         System.out.println("Chemistry Part-1          | CHY1   ");
18         System.out.println("Chemistry Part-2          | CHY2   ");
19         System.out.println("English                   | ENG    ");
20         System.out.println("Environmental Studies      | EVS    ");
21         System.out.println("Engineering Mechanics (Statics) | EMS    ");
22         System.out.println("Engineering Mechanics (Dynamics) | EMD    ");
23         System.out.println("Quantum Mechanics          | QMEC   ");
24         System.out.println("Engineering Thermodynamics   | THM    ");
25         System.out.println("Computer Science            | CS     ");
26         System.out.println("Fluid Mechanics              | FME    ");
27         System.out.println("-----");
28         System.out.println("-----");

```

```
29     String[] crsDepartments = {"MAT", "PHY",
30             "ELE1", "ELE2",
31             "CHY1", "CHY2",
32             "ENG", "EVS",
33             "EMS", "EMD",
34             "QMEC", "THM",
35             "CS", "FME"};
36     int[] crsNumbers = {101, 204,
37                         306, 200,
38                         303, 601,
39                         112, 231,
40                         229, 402,
41                         403, 209,
42                         319, 200};
43     double[] crsCredits = {1, 2, 3, 4, 5, 6, 6, 5, 4, 3, 2, 1, 2, 3};
44     int x;
45     for(x = 0; x < courses.length; ++x)
46         courses[x] = new Course();
47     for(x = 0; x < courses.length; ++x)
48     {
49         try
50         {
51             courses[x] = new Course(crsDepartments[x],
52                                     crsNumbers[x], crsCredits[x]);
53         }
54         catch(CourseException error)
55         {
56             System.out.println("Course: \n" + x +
57                                 " Error: \n" + error.getMessage());
58         }
59     }
60     System.out.println("\nCourse values - ");
61     for(x = 0; x < courses.length; ++x)
62         System.out.println(courses[x].toString());
63     }
64 }
```

Output console –

```
Output - Run (ThrowCourseException) - Editor
Output - Run (ThrowCourseException) X

cd C:\Users\Owais\Documents\NetBeansProjects\Lab8; "JAVA_HOME=C:\\Program File.
Running NetBeans Compile On Save execution. Phase execution is skipped and out;
Scanning for projects...

-----< com.ksvmediaplayer:Lab8 >-----
Building Lab8 1.0-SNAPSHOT
-----[ jar ]-----
--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab8 ---
WELCOME TO MID-TOWN HIGH JUNIOR COLLEGE, NEW YORK
-----

-----|-----
Course Name | Course Code
-----|-----
Mathematics & Statistics | MAT
Physics | PHY
Electronics Paper-1 | ELE1
Electronics Paper-2 | ELE2
Chemistry Part-1 | CHY1
Chemistry Part-2 | CHY2
English | ENG
Environmental Studies | EVS
Engineering Mechanics (Statics) | EMS
Engineering Mechanics (Dynamics) | EMD
Quantum Mechanics | QMEC
Engineering Thermodynamics | THM
Computer Science | CS
Fluid Mechanics | FME
-----|-----


Course:
2 Error:
The Course code number must be 3 numerals
Department: ELE1 Course number:306Credits: 3.0
Course:
3 Error:
The Course code number must be 3 numerals
Department: ELE2 Course number:200Credits: 4.0
Course:
4 Error:
The Course code number must be 3 numerals
Department: CHY1 Course number:303Credits: 5.0
Course:
5 Error:
The Course code number must be 3 numerals Course number should be between 100 and 499
Department: CHY2 Course number:601Credits: 6.0
Course:
10 Error:
```

```
The Course code number must be 3 numerals
Department: QMEC Course number:403Credits: 2.0
Course:
12 Error:
The Course code number must be 3 numerals
Department: CS Course number:319Credits: 2.0

Course values -
Department: MAT Course number:101Credits: 1.0
Department: PHY Course number:204Credits: 2.0
Department: Course number:0Credits: 0.0
Department: Course number:0Credits: 0.0
Department: Course number:0Credits: 0.0
Department: Course number:0Credits: 0.0
Department: ENG Course number:112Credits: 6.0
Department: EVS Course number:231Credits: 5.0
Department: EMS Course number:229Credits: 4.0
Department: EMD Course number:402Credits: 3.0
Department: Course number:0Credits: 0.0
Department: THM Course number:209Credits: 1.0
Department: Course number:0Credits: 0.0
Department: FME Course number:200Credits: 3.0
-----
BUILD SUCCESS
-----
Total time: 1.482 s
Finished at: 2021-12-14T12:50:17+05:30
-----
```

Solution of Q.2] b) after creating un-checked exception –

RuntimeException.java - Editor

RuntimeException.java

Source History

```
1 /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2 /* CSE2005 LAB-8 */
3 public class RuntimeException extends Exception {
4     public RuntimeException(String string) {
5         super(string);
6     }
7 }
```

CustomRuntimeException.java - Editor

CustomRuntimeException.java

Source History

```
1 /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2 /* CSE2005 LAB-8 */
3 import java.io.IOException;
4 public class CustomRuntimeException extends Exception {
5     public CustomRuntimeException(IOException ioe) {
6     }
7 }
8 }
```

GradeException.java - Editor

GradeException.java X

Source History

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  public class GradeException extends Exception
4  {
5      public static final char[] VALID_GRADES =
6          {'A', 'B', 'C', 'D', 'F', 'I'};
7      public GradeException(String string)
8      {
9          super(string);
10     }
11 }
```

15:1 |INS|

ScoreException.java - Editor

ScoreException.java X

Source History

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  public class ScoreException extends Exception {
4      public ScoreException(String s) {
5          super(s);
6      }
7 }
```

9:1 |INS|

ScoreTest.java –

```
/* KHAN MOHD OWAIS RAZA 20BCD7138 */
/* CSE2005 LAB-8 */
import java.util.*;
public class ScoreTest {
    public static void main(String args[]) throws Exception {
        Scanner input = new Scanner(System.in);
        int[] ids = {1234, 2345, 3456, 4567, 5678};
        int[] scores = {0, 0, 0, 0, 0};
        String scoreString = new String();
        final int HIGHLIMIT = 100;
        String inString, outString = "";
        for (int x = 0; x < ids.length; ++x) {
            System.out.println("ENTER SCORE FOR STUDENT ID NUMBER: " + ids[x]);
            inString = input.next();
            scores[x] = Integer.parseInt(inString);
            try{
                if (scores[x]>HIGHLIMIT){
                    throw (new ScoreException(scoreString));
                }
            }
            catch (ScoreException e){
                System.out.println("SCORE OVER 100");
                scores[x] = 0;
            }
        }
        for (int x = 0; x < ids.length; ++x)
            outString = outString + "\nID Number: "
                        + ids[x] + "\nScore: "
                        +scores[x] + "\n";
        System.out.println(outString);
    }
}
```

GradeTest.java –

```
/* KHAN MOHD OWAIS RAZA 20BCD7138 */
/* CSE2005 LAB-8 */
import java.util.*;
public class GradeTest {
    public static void main(String args[]) throws Exception {
        Scanner input = new Scanner(System.in);
        int[] ids = {1234, 1245, 1267, 1278, 2345,
                     1256, 3456, 3478, 4567, 5678
                };
        char[] grades = new char[10];
        String gradeString = new String();
        final int HIGHLIMIT = 100;
        String inString, outString = "";
        int flag = 0;
        for (int x = 0; x < ids.length; ++x) {
            System.out.println("ENTER GRADE FOR STUDENT ID NUMBER: "
                               + ids[x]);
            inString = input.next();
            grades[x] = inString.charAt(0);
            try
            {
                flag = 0;
                for(int y = 0; y < GradeException.VALID_GRADES.length; ++y)
                {
                    if(grades[x] == GradeException.VALID_GRADES[y])
                        flag = 1;
                }
                if(flag == 0)
                {
                    gradeString = "Invalid grade";
                    throw(new GradeException(gradeString));
                }
            }
        }
    }
}
```

```

        catch(GradeException e)
        {
            System.out.println(e.getMessage());
            grades[x] = 'I';
        }
    }

    for (int x = 0; x < ids.length; ++x)
        outString = outString + "\nID Number: "
        + ids[x] + "\nGrade: "
        + grades[x] + "\n";
    System.out.println(outString);
}
}

```

ScoreGradeTest.java –

```

/* KHAN MOHD OWAIS RAZA 20BCD7138 */
/* CSE2005 LAB-8 */
import java.util.Scanner;
public class ScoreGradeTest {
    private static Scanner sc;
    public static void main(String[] args)
    {
        int PHYSICS,
            CHEMISTRY,
            MATHEMATICS,
            ENGLISH,
            ELECTRONICS;
        float total, Percentage;
        sc = new Scanner(System.in);
        System.out.print("Five Subjects Marks : ");
        System.out.print("Student ID number : 1234");
        PHYSICS = sc.nextInt();
        CHEMISTRY = sc.nextInt();

```

```
MATHEMATICS = sc.nextInt();
ENGLISH = sc.nextInt();
ELECTRONICS = sc.nextInt();
total = PHYSICS + CHEMISTRY + MATHEMATICS + ENGLISH + ELECTRONICS;
Percentage = (total / 500) * 100;
System.out.println(" Total Marks = " + total);
System.out.println(" Marks Percentage = " + Percentage);
if(Percentage > 90)
{
    System.out.println("\n Grade: A");
}
else if(Percentage > 80 && Percentage < 89)
{
    System.out.println("\n Grade: B");
}
else if(Percentage > 70 && Percentage < 79)
{
    System.out.println("\n Grade: C");
}
else if(Percentage > 60 && Percentage < 69)
{
    System.out.println("\n Grade: D");
}
else if(Percentage < 60)
{
    System.out.println("\n Grade: F");
}
else
{
    System.out.println("\n Fail");
}
}
```

Output Console (ScoreTest.java) –

```
Output - Run (ScoreTest) - Editor
Output - Run (ScoreTest) x

cd C:\Users\Owais\Documents\NetBeansProjects\Lab8Ques2; "JAVA_HOME=C:\\Program Files\\Java\\jdk-15.0.1\\bin" & "C:\\Program Files\\NetBeans 12.1\\java\\javac.exe" ScoreTest.java
Running NetBeans Compile On Save execution. Phase execution is skipped and only the output is shown.
Scanning for projects...

-----< com.ksvmediaplayer:Lab8Ques2 >-----
Building Lab8Ques2 1.0-SNAPSHOT
-----[ jar ]-----
--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab8Ques2 ---
ENTER SCORE FOR STUDENT ID NUMBER: 1234
87
ENTER SCORE FOR STUDENT ID NUMBER: 2345
98
ENTER SCORE FOR STUDENT ID NUMBER: 3456
78
ENTER SCORE FOR STUDENT ID NUMBER: 4567
96
ENTER SCORE FOR STUDENT ID NUMBER: 5678
95

ID Number: 1234
Score: 87

ID Number: 2345
Score: 98

ID Number: 3456
Score: 78

ID Number: 4567
Score: 96

ID Number: 5678
Score: 95

-----
BUILD SUCCESS
-----
Total time: 18.222 s
Finished at: 2021-12-14T14:19:51+05:30
-----
```

Output Console (GradeTest.java) -

```
Output - Run (GradeTest) - Editor
Output - Run (GradeTest) X
cd C:\Users\Owais\Documents\NetBeansProjects\Lab8Ques2; "JAVA_HOME=C:\\\\Pr
Running NetBeans Compile On Save execution. Phase execution is skipped an
Scanning for projects...

-----< com.csvmediaplayer:Lab8Ques2 >-----
Building Lab8Ques2 1.0-SNAPSHOT
-----[ jar ]-----
--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab8Ques2 ---
ENTER GRADE FOR STUDENT ID NUMBER: 1234
A
ENTER GRADE FOR STUDENT ID NUMBER: 1245
B
ENTER GRADE FOR STUDENT ID NUMBER: 1267
C
ENTER GRADE FOR STUDENT ID NUMBER: 1278
D
ENTER GRADE FOR STUDENT ID NUMBER: 2345
E
Invalid grade
ENTER GRADE FOR STUDENT ID NUMBER: 1256
F
ENTER GRADE FOR STUDENT ID NUMBER: 3456
G
Invalid grade
ENTER GRADE FOR STUDENT ID NUMBER: 3478
H
Invalid grade
ENTER GRADE FOR STUDENT ID NUMBER: 4567
I
ENTER GRADE FOR STUDENT ID NUMBER: 5678
J
Invalid grade

ID Number: 1234
Grade: A

ID Number: 1245
Grade: B

ID Number: 1267
Grade: C

ID Number: 1278
Grade: D

ID Number: 2345
Grade: I

ID Number: 1256
Grade: F
```

```
ID Number: 3456
```

```
Grade: I
```

```
ID Number: 3478
```

```
Grade: I
```

```
ID Number: 4567
```

```
Grade: I
```

```
ID Number: 5678
```

```
Grade: I
```

```
BUILD SUCCESS
```

```
Total time: 22.047 s
```

```
Finished at: 2021-12-14T14:37:10+05:30
```

```
<
```

Output Console (ScoreGradeTest.java) –

Output - Run (ScoreGradeTest) - Editor

Output - Run (ScoreGradeTest) X

cd C:\Users\Owais\Documents\NetBeansProjects\Lab8Ques2; "JAVA_HOME=C:\\\\Pro
Running NetBeans Compile On Save execution. Phase execution is skipped and
Scanning for projects...

-----< com.ksvmediaplayer:Lab8Ques2 >-----

[jar]

Building Lab8Ques2 1.0-SNAPSHOT

--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab8Ques2 ---

90
92
94
96
98

Five Subjects Marks : Student ID number : 1234 Total Marks = 470.0
Marks Percentage = 94.0

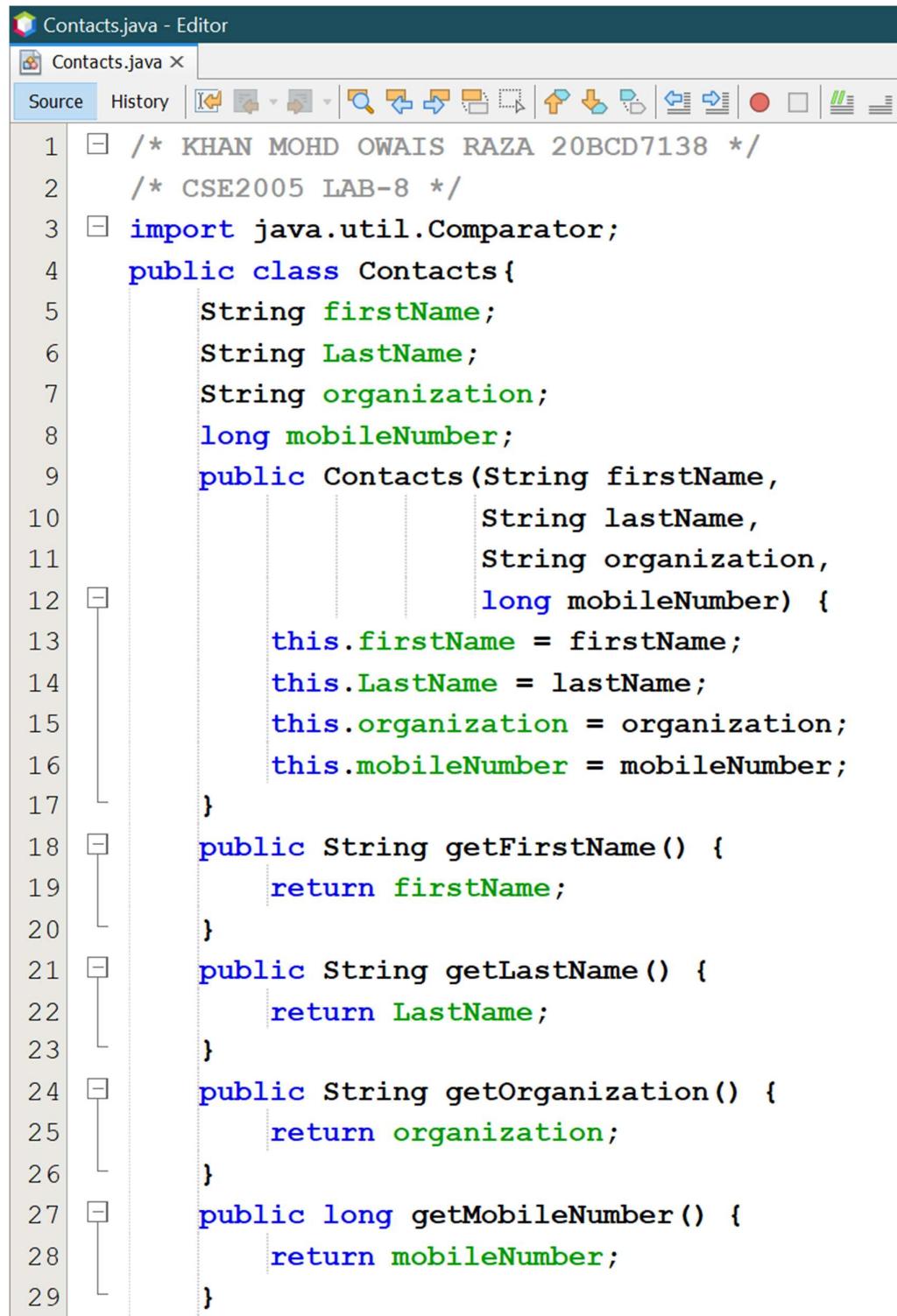
Grade: A

BUILD SUCCESS

Total time: 19.108 s
Finished at: 2021-12-14T14:43:30+05:30

Q.3] Develop a class “Contacts” with attributes first name, last name, organization, mobile number. Develop another class “ContactList” to hold all contacts of a user in the form of Contacts class objects and as an Array List. Develop a menu-based application to add, to delete, to update, and to display all contacts. During display of each contact object from the ArrayList the developer not able to access attributes. Find the way to display to all objects using Java predefined methods, in the order of insertions taken place into an ArrayList.

Contacts.java –



The screenshot shows a Java code editor window titled "Contacts.java - Editor". The file "Contacts.java" is open. The code defines a class "Contacts" with attributes: firstName, LastName, organization, and mobileNumber. It includes a constructor and four getter methods. The code is color-coded: comments are in grey, strings are in green, and keywords are in blue. The editor interface includes tabs for "Source" and "History", and a toolbar with various icons.

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  import java.util.Comparator;
4  public class Contacts{
5      String firstName;
6      String LastName;
7      String organization;
8      long mobileNumber;
9      public Contacts(String firstName,
10                      String lastName,
11                      String organization,
12                      long mobileNumber) {
13          this.firstName = firstName;
14          this.LastName = lastName;
15          this.organization = organization;
16          this.mobileNumber = mobileNumber;
17      }
18      public String getFirstName() {
19          return firstName;
20      }
21      public String getLastname() {
22          return LastName;
23      }
24      public String getOrganization() {
25          return organization;
26      }
27      public long getMobileNumber() {
28          return mobileNumber;
29      }
}
```

```
30     @Override
31     public String toString() {
32         return
33             "firstName='" + firstName + '\'' +
34             ", LastName='" + LastName + '\'' +
35             ", organization='" + organization + '\'' +
36             ", mobileNumber=" + mobileNumber;
37     }
38
39     class FirstNameSorter implements Comparator<Contacts>{
40         @Override
41         public int compare(Contacts o1, Contacts o2) {
42             return o1.getFirstName().compareTo(o2.getFirstName());
43         }
44     }

```

ContactsList.java –

The screenshot shows a Java code editor window titled "ContactsList.java - Editor". The code is a simple application for managing a list of contacts. It uses a Scanner to read input from the user and an ArrayList to store Contact objects. Each contact is defined by four fields: firstName, lastName, organization, and mobileNumber.

```
1  /* KHAN MOHD OWAIS RAZA 20BCD7138 */
2  /* CSE2005 LAB-8 */
3  import java.util.ArrayList;
4  import java.util.Collections;
5  import java.util.Comparator;
6  import java.util.Scanner;
7  public class ContactsList {
8      public static void main(String[] args) {
9          String firstName;
10         String lastName;
11         String organization;
12         long mobileNumber;
13         Scanner scanner = new Scanner(System.in);
14         ArrayList<Contacts> contactListArray = new ArrayList<>();
15         System.out.println("Enter the number of Contacts you want to enter");
16         int n = scanner.nextInt();
17         for(int i=1;i<=n;i++) {
18             {
19                 System.out.println("Enter the firstName of the Person "+i);
20                 firstName=scanner.next();
21                 System.out.println("Enter the LastName of the Person "+i);
22                 lastName = scanner.next();
23                 System.out.println("Enter the Organization Name "+i);
24                 organization = scanner.next();
25                 System.out.println("Enter the phone Number "+i);
26                 mobileNumber = scanner.nextLong();
27                 contactListArray.add(new Contacts(firstName,
28                                         lastName,
29                                         organization,
30                                         mobileNumber));
31             }
32         }
33     }
34 }
```

```

31         System.out.println();
32     }
33     System.out.println("Printing of the unsorted Passed ArrayList");
34     System.out.println(contactListArray);
35     System.out.println("Printing arrayList in Ascending order of firstName");
36     Collections.sort(contactListArray, new FirstNameSorter());
37     System.out.println(contactListArray);
38     System.out.println("Printing list in Descending order of lastName");
39     Comparator<Contacts> comparator
40         = Comparator.comparing(e -> e.getLastName());
41     contactListArray.sort(comparator.reversed());
42     System.out.println(contactListArray);
43 }
44 }
```

Output console –

```

-----< com.ksvmediaplayer:Lab8Ques3 >-----
Building Lab8Ques3 1.0-SNAPSHOT
-----[ jar ]-----

--- exec-maven-plugin:3.0.0:exec (default-cli) @ Lab8Ques3 ---
Enter the number of Contacts you want to enter
2
Enter the firstName of the Person 1
PERSON-A
Enter the LastName of the Person 1
ABC
Enter the Organization Name 1
ORG-1
Enter the phone Number 1
1234567890

Enter the firstName of the Person 2
PERSON-2
Enter the LastName of the Person 2
XYZ
Enter the Organization Name 2
ORG-2
Enter the phone Number 2
0987654321

Printing of the unsorted Passed ArrayList
[firstName='PERSON-A', LastName='ABC', organization='ORG-1', mobileNumber=1234567890, firstName='PERSON-2', LastName='XYZ', organization='ORG-2', mobileNumber=987654321]
Printing arrayList in Ascending order of firstName
[firstName='PERSON-2', LastName='XYZ', organization='ORG-2', mobileNumber=987654321, firstName='PERSON-A', LastName='ABC', organization='ORG-1', mobileNumber=1234567890]
Printing list in Descending order of lastName
[firstName='PERSON-2', LastName='XYZ', organization='ORG-2', mobileNumber=987654321, firstName='PERSON-A', LastName='ABC', organization='ORG-1', mobileNumber=1234567890]
-----
BUILD SUCCESS
-----
Total time: 01:28 min
Finished at: 2021-12-14T23:45:37+05:30
-----X-----X-----
```