

JD 521 Formative Assessment 2

CONTENTS

Questions:.....	2
Question 1.....	2
studentMarksReport:	2
Student	3
Question 1 Results:.....	4
Question 2.....	7
DragonKiller	7
Question 2 Results:.....	9

QUESTIONS:

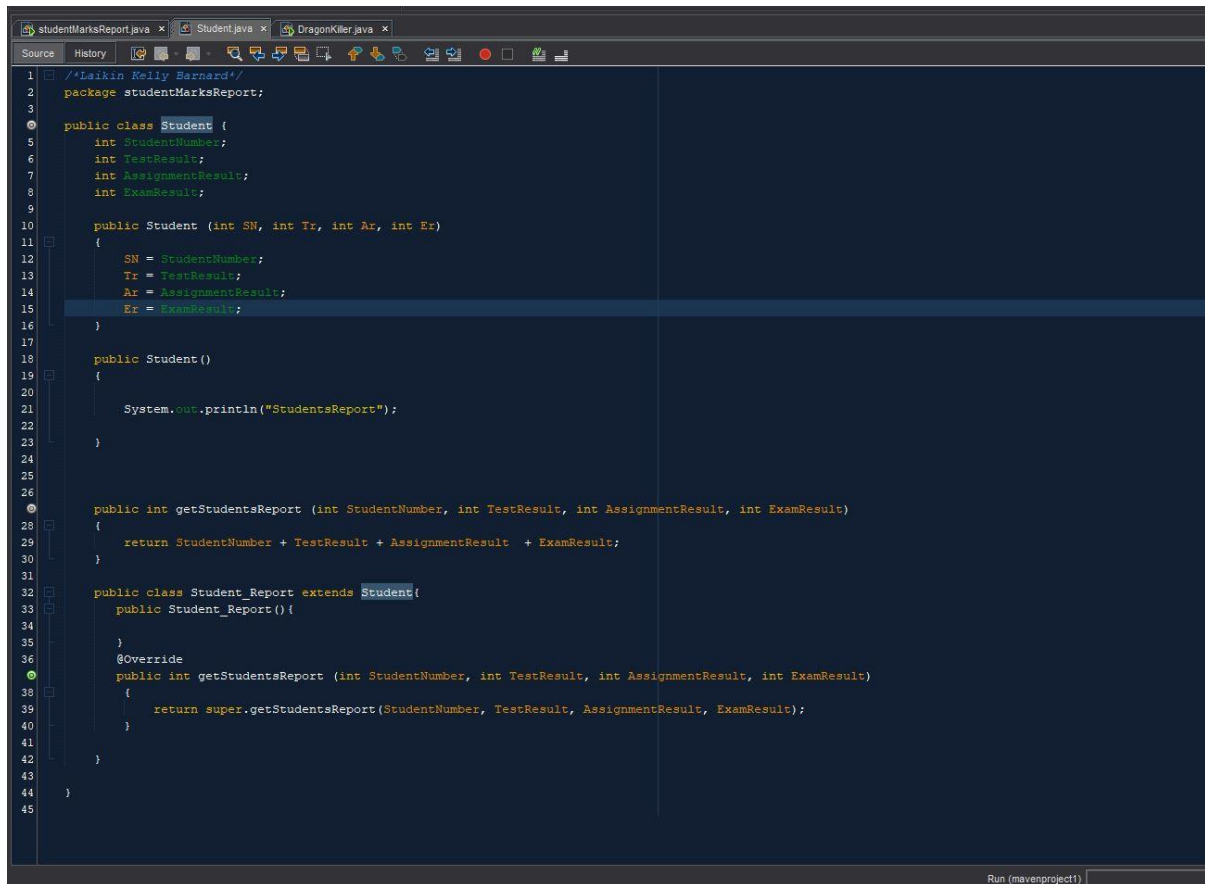
QUESTION 1

STUDENTMARKSREPORT:

```
1 // Leithin Bernard 4955 //
2 package studentMarksReport;
3 import java.util.Scanner;
4 import javax.swing.JOptionPane;
5
6 public class studentMarksReport {
7
8     public static void main(String[] args) {
9         Scanner validate = new Scanner(System.in);
10         Student st = new Student();
11         st.StudentNumber = 6955;
12         st.AssignmentResult = 25;
13         st.ExamResult = 50;
14
15         double TResults = 0;
16         double AResults = 0;
17         double EResults = 0;
18         int SNumber = 0;
19
20         /*-----*/
21         while(SNumber < 1 || SNumber > 9000){
22             try{
23                 String StudentNumber = JOptionPane.showInputDialog(null,
24                     "Student Number",
25                     "Enter your Student Number",
26                     JOptionPane.QUESTION_MESSAGE);
27             }
28             SNumber = Integer.parseInt(StudentNumber);
29         }
30         catch (Exception e){
31             System.out.println("Please enter 4 digit student number");
32         }
33     }
34
35     /*-----*/
36     while(TResults < 1 || TResults > 100){
37         try{
38             String TestResults = JOptionPane.showInputDialog(null,
39                 "Test Results ",
40                 "Enter your Test Results",
41                 JOptionPane.QUESTION_MESSAGE);
42             TResults = Double.parseDouble(TestResults);
43         }
44         catch (Exception e){
45             System.out.println("Please enter ");
46         }
47     }
48 }
```

```
47         catch (Exception e){
48             System.out.println("Please enter ");
49         }
50     }
51
52     /*-----*/
53     while(AResults < 1 || AResults > 100){
54         try{
55             String AssignmentResult = JOptionPane.showInputDialog(null,
56                 "Assignment Result",
57                 "Enter your Assignment Result",
58                 JOptionPane.QUESTION_MESSAGE);
59             AResults = Double.parseDouble(AssignmentResult);
60         }
61         catch (Exception e){
62             System.out.println("Please enter 4 digit student number");
63         }
64     }
65
66     /*-----*/
67     while(EResults < 1 || EResults > 100){
68         try{
69             String ExamResult = JOptionPane.showInputDialog(null,
70                 "Exam Result ",
71                 "Enter your Exam Result",
72                 JOptionPane.QUESTION_MESSAGE);
73             EResults = Double.parseDouble(ExamResult);
74         }
75         catch (Exception e){
76             System.out.println("Please enter 4 digit student number");
77         }
78     }
79
80     /*-----*/
81     double TotalScore = (((TResults + AResults) * 0.25) + (EResults * 0.5));
82     System.out.println(TotalScore);
83
84     /*-----*/
85     JOptionPane.showMessageDialog(null,
86         ("Student: " + SNumber + " Final Report: " + TotalScore),
87         "Student Result",
88         JOptionPane.PLAIN_MESSAGE);
89 }
90
91
92 }
```

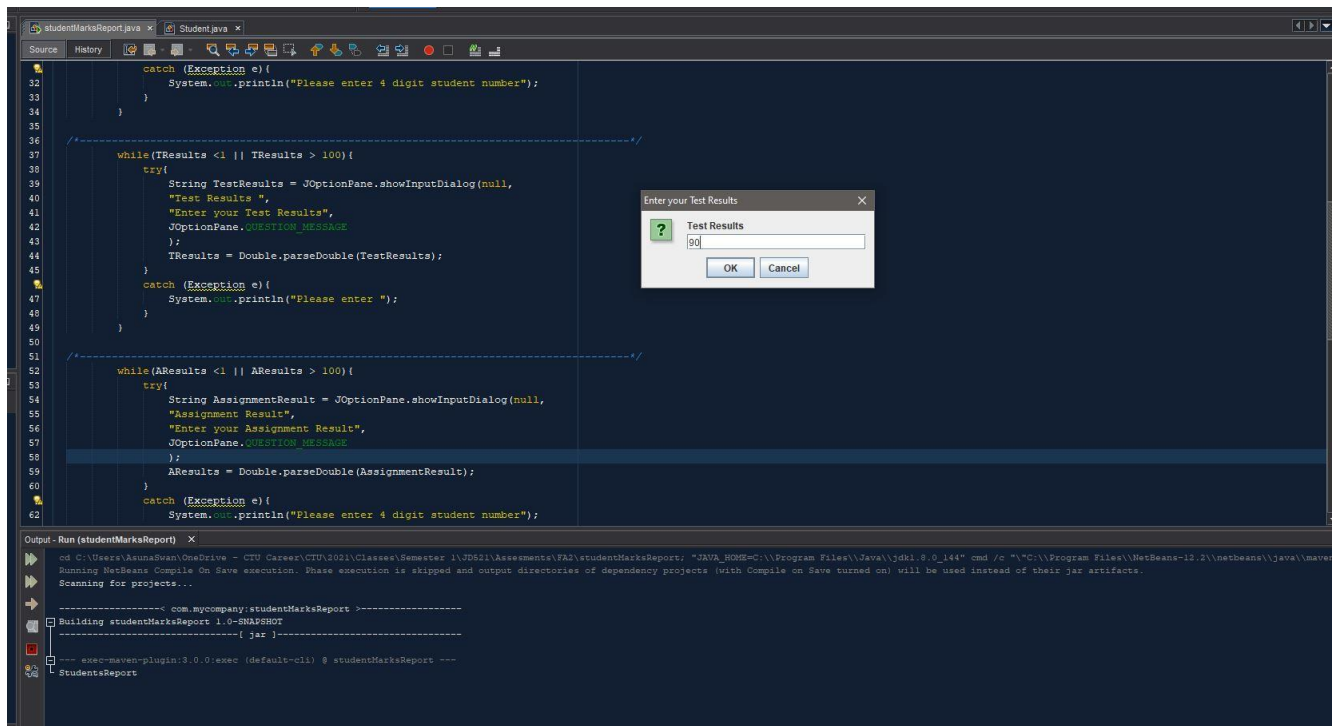
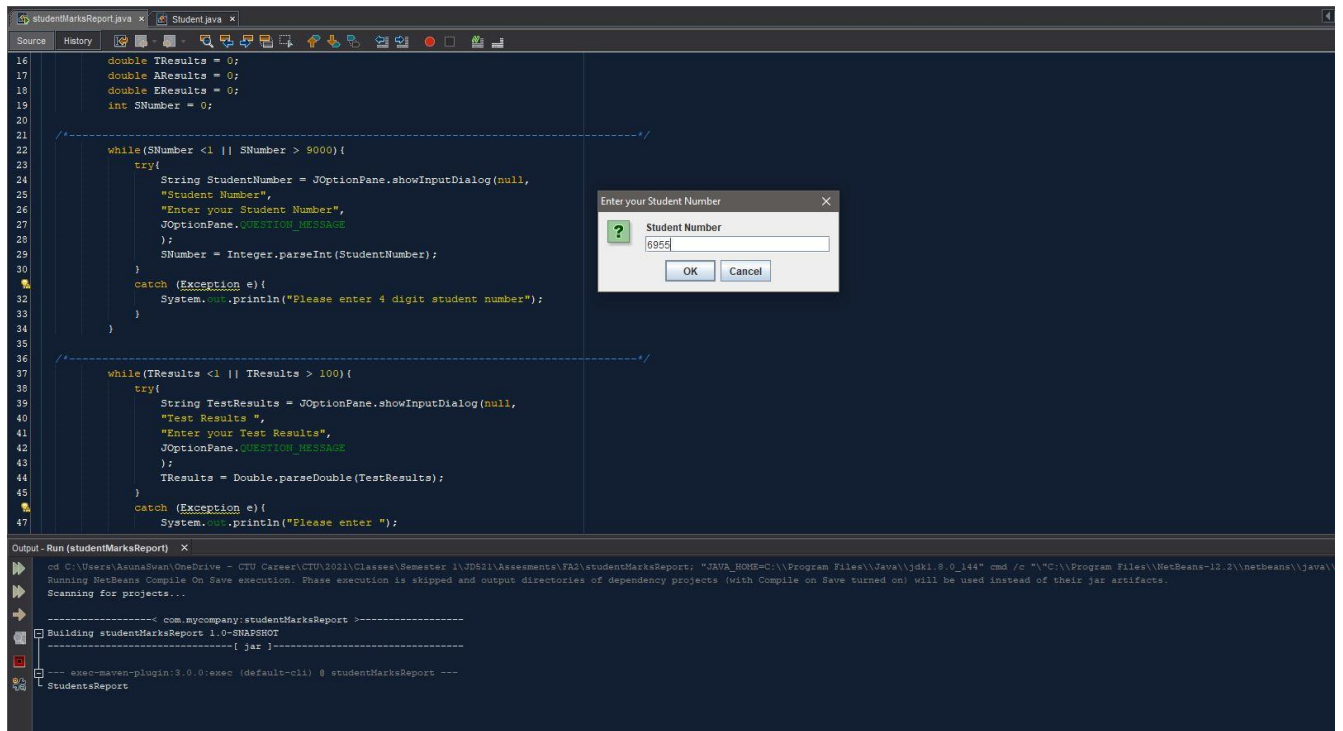
STUDENT

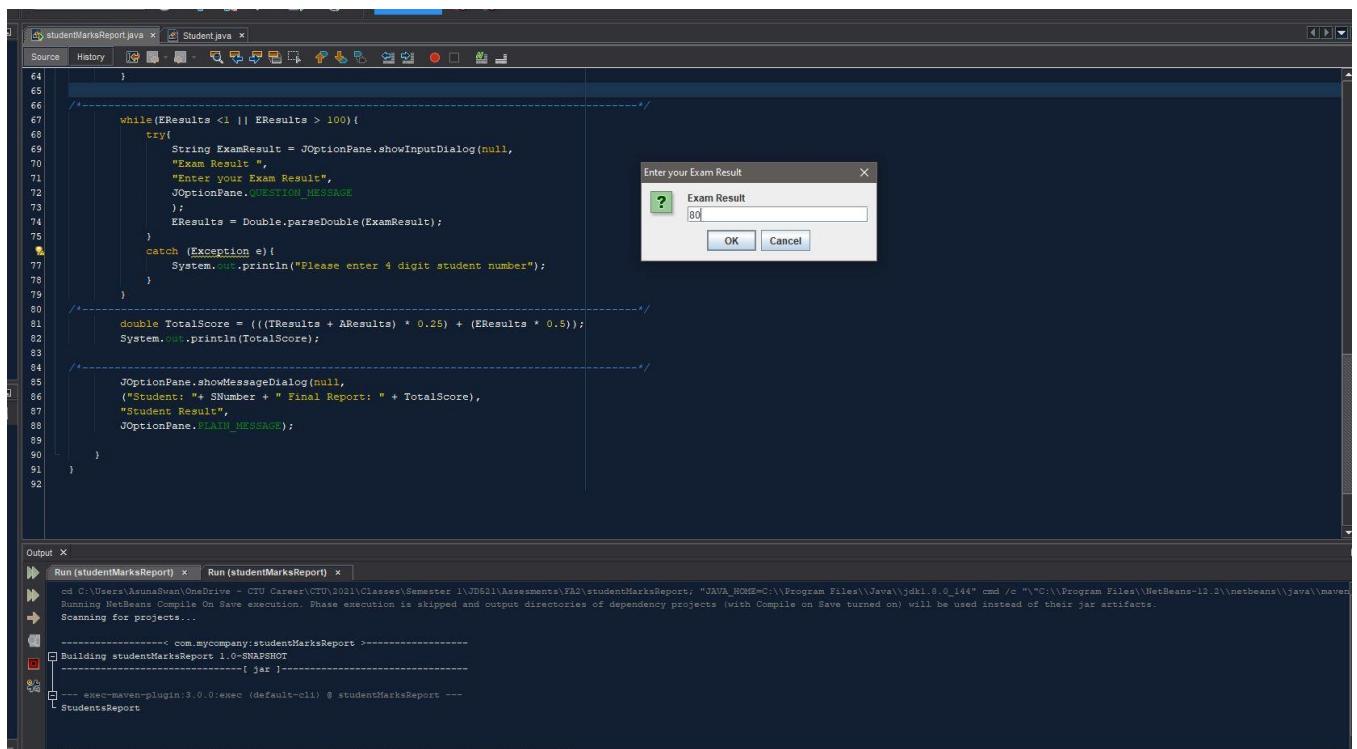
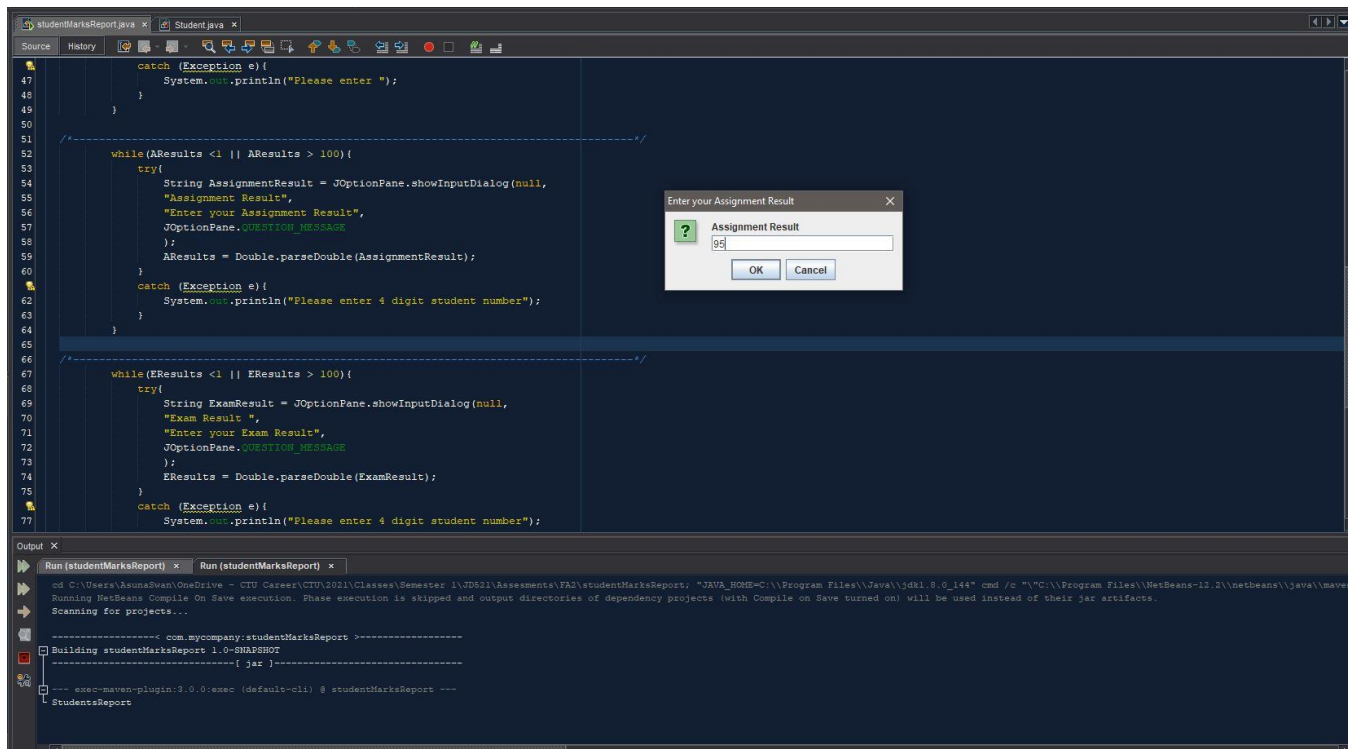


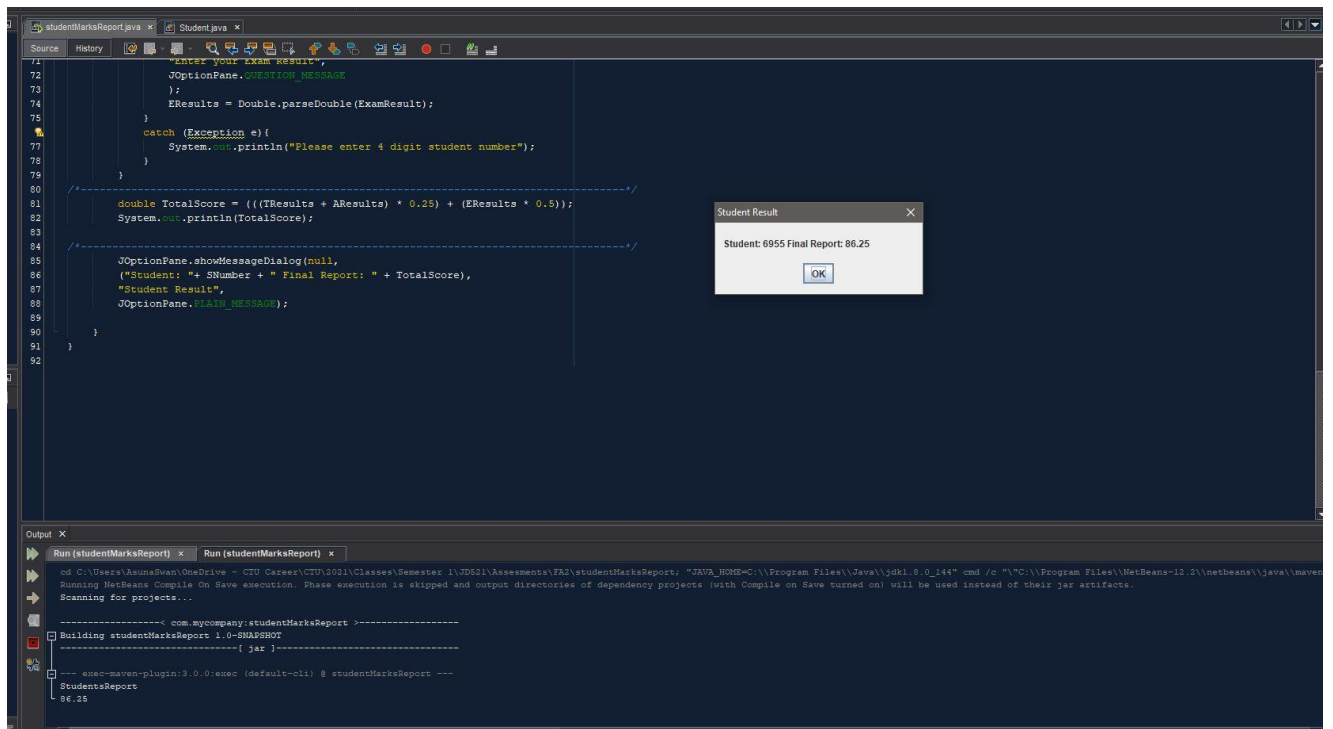
```
1  /*Leikin Kelly Bernard*/
2  package studentMarksReport;
3
4  public class Student {
5      int StudentNumber;
6      int TestResult;
7      int AssignmentResult;
8      int ExamResult;
9
10     public Student (int SN, int Tr, int Ar, int Er)
11     {
12         SN = StudentNumber;
13         Tr = TestResult;
14         Ar = AssignmentResult;
15         Er = ExamResult;
16     }
17
18     public Student ()
19     {
20         System.out.println("StudentsReport");
21     }
22
23
24
25
26
27
28     public int getStudentsReport (int StudentNumber, int TestResult, int AssignmentResult, int ExamResult)
29     {
30         return StudentNumber + TestResult + AssignmentResult + ExamResult;
31     }
32
33     public class Student_Report extends Student{
34         public Student_Report () {
35
36         }
37         @Override
38         public int getStudentsReport (int StudentNumber, int TestResult, int AssignmentResult, int ExamResult)
39         {
40             return super.getStudentsReport(StudentNumber, TestResult, AssignmentResult, ExamResult);
41         }
42     }
43
44 }
45
```

Run (mavenproject1)

QUESTION 1 RESULTS:







QUESTION 2

DRAGONKILLER

```
DragonKiller.java x
Source History
1 // Leikin Bernard 68551/
2 package DragonKiller;
3 import java.util.*;
4
5 public class DragonKiller {
6     public static void main(String[] args) {
7         // TODO code application logic here
8         DragonKiller dragonObj = new DragonKiller();
9         dragonObj.NameSurname = RemoveSpace(dragonObj.NameSurname);
10        int Count = CountDrag(dragonObj.NameSurname);
11        int[] arrayDragon = new int[Count];
12        arrayDragon = arrayDragon(Count);
13        arrayShow(arrayDragon);
14        System.out.println(Count);
15        arrayDragon = insertionSort(arrayDragon);
16        arrayShow(arrayDragon);
17        int MakeZero = findDragon(arrayDragon, getKey());
18        arrayDragon = ReplaceZero(arrayDragon, MakeZero);
19        arrayShow(arrayDragon);
20    }
21
22    private static int[] insertionSort(int[] arrayDragon) {
23        int n = arrayDragon.length;
24        for (int i = 1; i < n; ++i) {
25            int key = arrayDragon[i];
26            int j = i - 1;
27
28            while (j >= 0 && arrayDragon[j] > key) {
29                arrayDragon[j + 1] = arrayDragon[j];
30                j = j - 1;
31            }
32            arrayDragon[j + 1] = key;
33        }
34        return arrayDragon;
35    }
36
37    private static int getKey() {
38        Scanner dragScanner = new Scanner(System.in);
39        System.out.println("Select a number to replace");
40        String inputString = dragScanner.nextLine();
41        dragScanner.close();
42        return Integer.parseInt(inputString);
43    }
44
45    private static int[] ReplaceZero(int[] arrayDragon, int MakeZero) {
46        arrayDragon[MakeZero] = 0;
47    }
48}
```

```
DragonKiller.java x
Source History
46 private static int[] ReplaceZero(int[] arrayDragon, int MakeZero) {
47     arrayDragon[MakeZero] = 0;
48     return arrayDragon;
49 }
50
51 }
52
53 private static int findDragon(int[] arrayDragon, int key) {
54     int l = 0, r = arrayDragon.length - 1;
55     while (l <= r) {
56         int m = l + (r - l) / 2;
57         // Check if x is present at mid
58         if (arrayDragon[m] == key)
59             return m;
60         // If x greater, ignore left half
61         if (arrayDragon[m] < key)
62             l = m + 1;
63         // If x is smaller, ignore right half
64         else
65             r = m - 1;
66     }
67     // If we reach here, then element was
68     // not present
69     return -1;
70 }
71
72 String NameSurname;
73
74 private static int [] arrayDragon(int Length){
75     int[] arrayDragon = new int[Length];
76     Random rand = new Random();
77
78     for(int i=0; i< Length; i++){
79         int ran = rand.nextInt(40) + 10;
80         if (ran % 2 != 0){
81             arrayDragon[i] = ran;
82         }
83         else{
84             i--;
85         }
86     }
87     return arrayDragon;
88 }
89
90 private static void arrayShow(int[] arrayDragon){
91
92     for(int i=0; i< arrayDragon.length; i++){
93         System.out.println("Index " + i + " = " + arrayDragon[i] + ",");
94     }
95 }
```



```
DragonKiller.java
Source History
82
83     }
84     else{
85         i--;
86     }
87     return arrayDragon;
88 }
89
90 private static void arrayShow(int[] arrayDragon){
91
92     for(int i=0; i< arrayDragon.length; i++){
93         System.out.println("Index " + i + " = " +arrayDragon[i] + "," );
94     }
95
96 }
97
98 private String getNameSurname(){
99     Scanner DragScanner = new Scanner(System.in);
100     System.out.println("Enter Name and Surname");
101     String inputString = DragScanner.nextLine();
102     return inputString;
103 }
104
105 DragonKiller(){
106     NameSurname = getNameSurname();
107 }
108
109 private static int CountDrag(String NameSurname){
110     return NameSurname.length();
111 }
112
113 private static String RemoveSpace(String NameSurname){
114     String returnNameSurname = NameSurname.replace(" ", "");
115     return NameSurname;
116 }
117
118
119
120 }
```

QUESTION 2 RESULTS:

```
Source History
109 private static int CountDrag(String NameSurname){
110     return NameSurname.length();
111 }
112
113 private static String RemoveSpace(String NameSurname){
114     String returnNameSurname = NameSurname.replace(" ", "");
115     return NameSurname;
116 }
117
118
119
120

Output - Run (DragonKiller) X
Building DragonKiller 1.0-SNAPSHOT
[ jax ]-----
--- exec-maven-plugin:3.0.0:exec (default-cli) @ DragonKiller ---
Enter Name and Surname
Laikis Bernard
Index 0 = 38,
Index 1 = 39,
Index 2 = 19,
Index 3 = 49,
Index 4 = 29,
Index 5 = 17,
Index 6 = 11,
Index 7 = 31,
Index 8 = 49,
Index 9 = 37,
Index 10 = 31,
Index 11 = 16,
Index 12 = 16,
Index 13 = 31,
14
Index 0 = 11,
Index 1 = 19,
Index 2 = 16,
Index 3 = 16,
Index 4 = 17,
Index 5 = 29,
Index 6 = 31,
Index 7 = 31,
Index 8 = 31,
Index 9 = 36,
Index 10 = 37,
Index 11 = 39,
Index 12 = 49,
Index 13 = 49,
Select a number to replace
|
```

```
Source History
109 private static int CountDrag(String NameSurname){
110     return NameSurname.length();
111 }
112
113 private static String RemoveSpace(String NameSurname){
114     String returnNameSurname = NameSurname.replace(" ", "");
115     return NameSurname;
116 }
117
118
119
120

Output - Run (DragonKiller) X
Index 0 = 11,
Index 1 = 19,
Index 2 = 16,
Index 3 = 16,
Index 4 = 17,
Index 5 = 29,
Index 6 = 31,
Index 7 = 31,
Index 8 = 31,
Index 9 = 36,
Index 10 = 37,
Index 11 = 39,
Index 12 = 49,
Index 13 = 49,
Select a number to replace
21
Index 0 = 11,
Index 1 = 19,
Index 2 = 16,
Index 3 = 16,
Index 4 = 17,
Index 5 = 29,
Index 6 = 0,
Index 7 = 31,
Index 8 = 31,
Index 9 = 36,
Index 10 = 37,
Index 11 = 39,
Index 12 = 49,
Index 13 = 49,
-----
BUILD SUCCESS
-----
Total time: 01:01 min
Finished at: 2021-02-28T19:49:42+02:00
|
```