sumanto.pal@accolitedigital.com

1.] 1st question from slide

Code:

```
import java.util.*;
public class handsOnExercise1 {
   // Test a: -5+8*6
   public static void q1() {
       System.out.println(-5+8*6);
   }
   // Test b: (55+9)%9
   public static void q2() {
       System.out.println((55+9)%9);
   }
   // Test c: 20+ -3*5/8
   public static void q3() {
       System.out.println(20+ -3*5/8);
   }
   // Test d: 5+15/3*2-8%3
   public static void q4() {
       System.out.println(5+15/3*2-8%3);
   }
   public static void main(String[] args) {
       q1();
       q2();
       q3();
       q4();
   }
}
```

Output:

```
sumanto@17bce0764:~/Desktop/xxxxx/JavaAssignment$ javac handsOnExercise1.java
sumanto@17bce0764:~/Desktop/xxxxx/JavaAssignment$ java handsOnExercise1
43
1
19
13
```

2.] 2st question from slide

Code:

```
public class slidesExercise2 {
   public static void main(String[] args) {
       String p1="", p2="";
       for (int i = 0; i < 6; i++)
           p1 = p1 + "* ";
       for (int i = 0; i < 35; i++)
           p1 = p1 + "=";
       p1 = p1 + "\n ";
       for (int i = 0; i < 5; i++)
           p1 = p1 + "* ";
       p1 += " ";
       for (int i = 0; i < 35; i++)
           p1 = p1 + "=";
       for (int i = 0; i < 45; i++)
           p2 = p2 + "=";
       for (int i = 0; i < 4; i++) {
           System.out.println(p1);
       }
       p2+="==";
       for (int i = 0; i < 8; i++) {
           System.out.println(p2);
       }
   }
}
```

Output:

```
sumanto@17bce0764:~/Desktop/xxxxx/JavaAssignment$ javac slidesExercise2.java
sumanto@17bce0764:~/Desktop/xxxxx/JavaAssignment$ java slidesExercise2
* * * * * * ===================================
 * * * * *  ===================================
* * * * * * ===================================
 * * * * *  ===================================
* * * * * * ===================================
 * * * * *  ===================================
* * * * * * ===================================
 * * * * *  ===================================
=============================================
=============================================
=============================================
=============================================
=============================================
=============================================
=============================================
=============================================
```

3.] Hand On question1 from assignment

Write a Java program to replace each substring of a given string that matches the given regular expression with the given replacement.

Code:

```
public class handOn1 {
   public static void main(String[] args) {
       String input = "hello I'm a senior java dev using for job";
       System.out.println("Original text ==> "+ input);
       System.out.print("New Text ==> "+input.replaceAll("(java|job|senior)",
"[$1]"));
   }
}
```

Output:

```
sumanto@17bce0764:~/Desktop/xxxxx/JavaAssignment$ java handOn1
Original text ==> hello I'm a senior java dev using for job
New Text ==> hello I'm a [senior] [java] dev using for [job]sum
```

4.] Hand On question2 from assignment

Write a Java program to get a reverse order view of the keys contained in a given map

Code:

```
import java.util.TreeMap;
public class q4 {
   public static void main(String args[]) {
       TreeMap<String, String> tree_map1 = new TreeMap<String, String>();
       tree_map1.put("P2", "java");
       tree_map1.put("P1", "javascript");
       tree_map1.put("P4", "php");
       tree_map1.put("P3", "go");
       System.out.println("Old " + tree_map1);
       System.out.println("Reversed Order " + tree_map1.descendingKeySet());
   }
}
```

Output:

```
sumanto@17bce0764:~/Desktop/xxxxx/JavaAssignment$ javac q4.java
sumanto@17bce0764:~/Desktop/xxxxx/JavaAssignment$ java q4
Old {P1=javascript, P2=java, P3=go, P4=php}
Reversed Order [P4, P3, P2, P1]
```

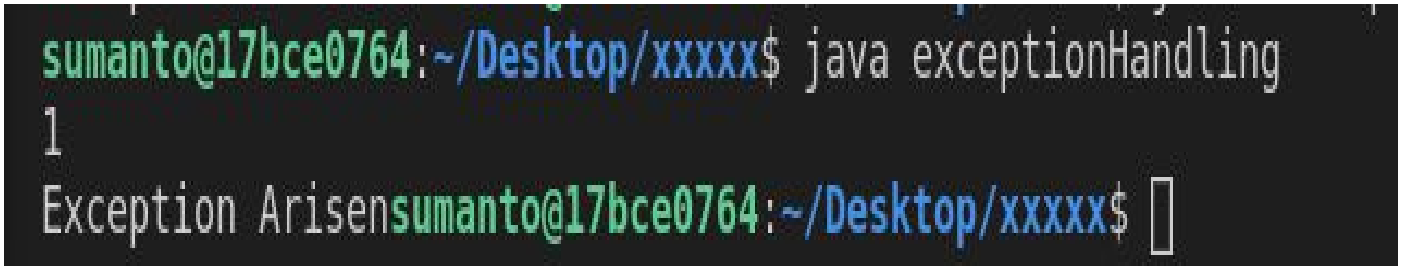
5.] Hand On question3 from assignment

Write your own unchecked Exception and throw it from you counter programme which counts 1 to 100.When you get Prime no while counting then throw this Exception and catch this to print you exception message.

Code:

```
public class exceptionHandling {
   public static Boolean prime(int num) {
       for (int i = 2; i  <= num; i++)
           if (num % i == 0)
               return true;
       return false;
   }
   static void counter() throws Exception
   {
       for (int i = 1; i <= 100; i++) {
           if (prime(i))
               throw new Exception("Exception Arisen");
           System.out.println(i);
       }
   }
   public static void main(String args[]) {
       try {
           counter();
       } catch (Exception e) {
           System.out.print(e.getMessage());
       }
   }
}
```

Output:



6.] Hand On question4 from assignment

Write a programme to serialize 3 fields out of 5 and deserialize it. Use UUID to prvent object mutation.

Code:

```
import java.io.*;
import java.util.*;
public class serialisation {
   public static void main(String[] args) {
       Data d1 = new Data(13, "Virat Kohli", 19134);
       System.out.println("\n====Serializing====\n");
```

```
        FileOutputStream file = new FileOutputStream("x.txt");
        ObjectOutputStream out = new ObjectOutputStream(file);
        out.writeObject(d1);
        out.close();
        file.close();
        System.out.println("Data before deserialization:\n ");
        System.out.println("id: " + d1.id);
        System.out.println("cricketer: " + d1.name);
        System.out.println("runs: " + d1.runs);
        System.out.println("wickets: " + d1.wickets);
        d1 = null;
        System.out.println("\n====Deserializing====\n");
        file = new FileInputStream("x.txt");
        ObjectInputStream in = new ObjectInputStream(file);
        d1 = (Data) in.readObject();
        in.close();
        file.close();
        System.out.println("\n\nData after deserialization:\n ");
        System.out.println("id: [NOT SERIALIZED] " + d1.id);
        System.out.println("cricketer: [NOT SERIALIZED] " + d1.name);
        System.out.println("runs: " + d1.runs);
        System.out.println("wickets: " + d1.wickets);
    }
}
class Data implements Serializable {

    transient String id;
    transient String name;
    int runs, wickets;

    public Data(int wickets, String name, int runs) {
        UUID serialversionUID = UUID.randomUUID();
        this.id = serialversionUID + "";
        this.wickets = wickets;
        this.runs = runs;
        this.name = name;
    }
}
```

Output:

```
sumanto@17bce0764:~/Desktop/xxxxx$ javac serialisation.java
sumanto@17bce0764:~/Desktop/xxxxx$ java serialisation

====Serializing====

Data before deserialization:

id: 4c29f52e-c430-40c2-bf79-e92be84cec74
cricketer: Virat Kohli
runs: 19134
wickets: 13

====Deserializing====



Data after deserialization:

id: [NOT SERIALIZED] null
cricketer: [NOT SERIALIZED] null
runs: 19134
wickets: 13
```