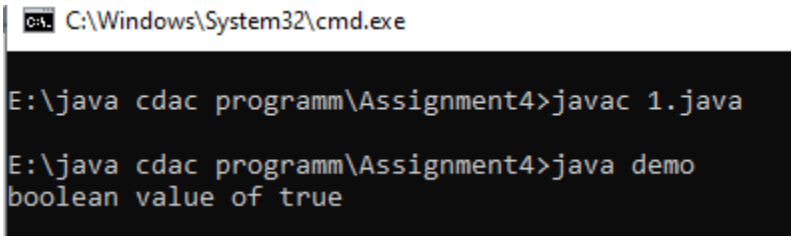
Q.1 Working with java.lang.Boolean

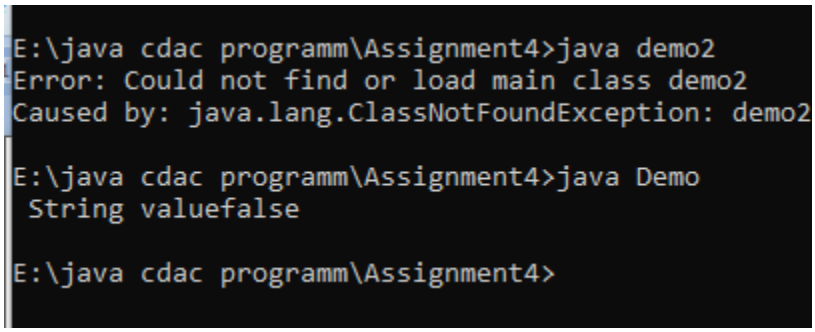
B) Declare a method-local variable status of type boolean with the value true and convert it to a String using the toString method. (Hint: Use Boolean.toString(Boolean)).

```
import java.lang.Boolean;
class demo{
 public static void main(String args[]){
    boolean status = true;
      String statusString = Boolean.toString(status);


      // Print out the result
      System.out.println("The boolean value as a String is: " + statusString);
                }
                }

}}
```



c. Declare a method-local variable strStatus of type String with the value "1" or "0" and attempt to convert it to a boolean. (Hint: parseBoolean method will not work as expected with "1" or "0").

```
import java.lang.String;
 class Demo{
        public static void main(String args[]){
                Boolean Status = true;
Boolean b = Boolean.valueOf(Status);
 System.out.println(" Boolean value : "  + b);         }}
```



e. Declare a method-local variable status of type boolean with the value true and convert it to the corresponding wrapper class using Boolean.valueOf(). (Hint: Use Boolean.valueOf(boolean)).

```
import java.lang.String;
 class Demo{
        public static void main(String args[]){
                String strStatus = "true";
Boolean b = Boolean.valueOf(strStatus);
 System.out.println(" Boolean value : "  + b);       }}
```

```
E:\java cdac programm\Assignment4>javac 1.java

E:\java cdac programm\Assignment4>java Demo
 Boolean value : true

E:\java cdac programm\Assignment4>
```

f. Declare a method-local variable strStatus of type String with the value "true" and convert it to the corresponding wrapper class using Boolean.valueOf(). (Hint: Use Boolean.valueOf(String)).

```
class Demo{

        public static void main(String args[]){

                String strStatus = "true";

                //System.out.println("Enter the value");

Boolean b = Boolean.valueOf(strStatus);

 System.out.println(" Boolean value : "  + b);

 }

}
```

```
E:\java cdac programm\Assignment4>javac 1.java

E:\java cdac programm\Assignment4>java Demo
 Boolean value : true

E:\java cdac programm\Assignment4>
```

2. Working with **java.lang.Byte**

b. Write a program to test how many bytes are used to represent a byte value using the BYTES field. (Hint: Use Byte.BYTES).

```
class Demo{

   public static void main(String[] args) {

      byte b = byte.BYTES;

      System.out.println("bytes value :" + b);

   }
```

}

```
E:\java cdac programm\java core>javac 2.java

E:\java cdac programm\java core>java Demo
bytes value :1

E:\java cdac programm\java core>java Demo
bytes value :1

E:\java cdac programm\java core>javac 2.java

E:\java cdac programm\java core>java Demo
bytes value :4
```

Q.c Write a program to find the minimum and maximum values of byte using the MIN_VALUE and MAX_VALUE fields. (Hint: Use Byte.MIN_VALUE and Byte.MAX_VALUE).

```
class Demo{

    public static void main(String[] args) {

        byte b = Byte.MIN_VALUE;

      byte c = Byte.MAX_VALUE;

        System.out.println(" bytes min value :" + b);

           System.out.println(" bytes max value :" + c);

           }}
```

```
E:\java cdac programm\java core>javac 2.java

E:\java cdac programm\java core>java Demo
 bytes min value :-128
 bytes max value :127
```

d. Declare a method-local variable number of type byte with some value and convert it to a String using the toString method. (Hint: Use Byte.toString(byte)).

```
class Demo{

   public static void main(String[] args) {

      byte number = 6;

                String str = Byte.toString(number);

      System.out.println(" String value :" +str);

        }}
```

```
E:\java cdac programm\java core>javac 2.java

E:\java cdac programm\java core>java Demo
 String value :6
```

e. Declare a method-local variable strNumber of type String with some value and convert it to a byte value using the parseByte method. (Hint: Use Byte.parseByte(String)).

```
class Demo{

   public static void main(String[] args) {

      String s = "100";

                byte b = Byte.parseByte(s);

      System.out.println(" String value :" +b);

        }}
```

```
E:\java cdac programm\java core>javac 2.java

E:\java cdac programm\java core>java Demo
 String value :100
```

f. Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a byte value. (Hint: parseByte method will throw a NumberFormatException).

g. Declare a method-local variable number of type byte with some value and convert it to the corresponding wrapper class using Byte.valueOf(). (Hint: Use Byte.valueOf(byte)).

```
class Demo{

   public static void main(String[] args) {

      byte Number = 100;

                byte b = Byte.valueOf(Number);

      System.out.println(" String value :" +b);

        }}
```

```
E:\java cdac programm\java core>javac 2.java

E:\java cdac programm\java core>java Demo
 String value :100
```

h. Declare a method-local variable strNumber of type String with some byte value and convert it to the corresponding wrapper class using Byte.valueOf(). (Hint: Use Byte.valueOf(String)).

```
class Demo{

   public static void main(String[] args) {

      String strNumber = "10";

                byte b = Byte.valueOf(strNumber);

      System.out.println(" String value :" +b);

        }}
```

```
E:\java cdac programm\java core>java Demo
 String value :10

E:\java cdac programm\java core>
```

3. Working with **java.lang.Short**

b. Write a program to test how many bytes are used to represent a short value using the BYTES field. (Hint: Use Short.BYTES).

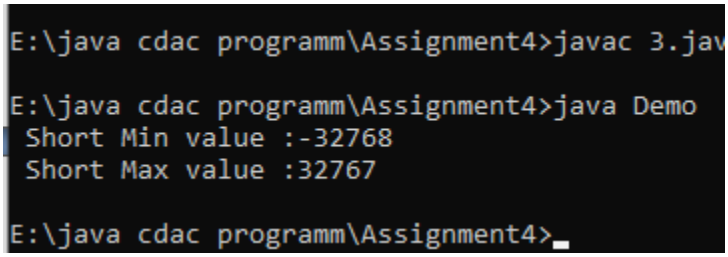
```
class Demo{

  public static void main(String[] args) {

                byte b = Short.BYTES;

      System.out.println(" String value :" +b);

}]
```

```
E:\java cdac programm\Assignment4>java Demo
 Bytes value :2

E:\java cdac programm\Assignment4>
```

c. Write a program to find the minimum and maximum values of short using the MIN_VALUE and MAX_VALUE fields. (Hint: Use Short.MIN_VALUE and Short.MAX_VALUE).

```
class Demo{

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

```
        Short b = Short.MIN_VALUE;

        Short c = Short.MAX_VALUE;

    System.out.println(" Short Min value :" +b);

        System.out.println(" Short Max value :" +c);

     }}
```



d. Declare a method-local variable number of type short with some value and convert it to a String using the toString method. (Hint: Use Short.toString(short)).

```
class Demo{

   public static void main(String[] args) {

        Short b = 2;

        String str = Short.toString(b);

    System.out.println(" String value :" +str);
```

}}

```
E:\java cdac programm\Assignment4>javac 3.java

E:\java cdac programm\Assignment4>java Demo
 String value :2

E:\java cdac programm\Assignment4>
```

e. Declare a method-local variable strNumber of type String with some value and convert it to a short value using the parseShort method. (Hint: Use Short.parseShort(String)).

```
class Demo{

        public static void main(String[] args) {

        String str = "50";

        Short s = Short.parseShort(str);

    System.out.println(" Short  value :" +s);

}

}
```

```
E:\java cdac programm\Assignment4>javac 3.java

E:\java cdac programm\Assignment4>java Demo
 Short value :50
```

f. Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a short value. (Hint: parseShort method will throw a NumberFormatException).

```
class Demo{

   public static void main(String[] args) {

                String str = "Ab12Cd3";

                Short s = Short.parseShort(str);

      System.out.println(" Short value :" +s);

}

}
```

```
  location: class String
1 error

E:\java cdac programm\Assignment4>javac 3.java

E:\java cdac programm\Assignment4>java Demo
Exception in thread "main" java.lang.NumberFormatException: For input string: "Ab12Cd3"
        at java.base/java.lang.NumberFormatException.forInputString(NumberFormatException.java:67)
        at java.base/java.lang.Integer.parseInt(Integer.java:668)
        at java.base/java.lang.Short.parseShort(Short.java:137)
        at java.base/java.lang.Short.parseShort(Short.java:163)
        at Demo.main(3.java:31)

E:\java cdac programm\Assignment4>
```

g. Declare a method-local variable number of type short with some value and convert it to the corresponding wrapper class using Short.valueOf(). (Hint: Use Short.valueOf(short)).

```
class Demo{

   public static void main(String[] args) {

                Short s= 55;

                Short s1 = Short.valueOf(s);

      System.out.println(" Short value :" +s1);

}

}
```

```
Microsoft Windows [Version 10.0.19045.4780]
(c) Microsoft Corporation. All rights reserved.

E:\java cdac programm\Assignment4>javac 3.java

E:\java cdac programm\Assignment4>java Demo
 Short value :55

E:\java cdac programm\Assignment4>
```

h. Declare a method-local variable strNumber of type String with some short value and convert it to the corresponding wrapper class using Short.valueOf(). (Hint: Use Short.valueOf(String)).

```
class Demo{

   public static void main(String[] args) {

                String strNumber = "40";

                Short s1 = Short.valueOf(strNumber);
```

```
        System.out.println(" Short value :" +s1);

}}
```

```
E:\java cdac programm\Assignment4>javac 3.java

E:\java cdac programm\Assignment4>java Demo
 Short value :40

E:\java cdac programm\Assignment4>javac 3.java
```

4. Working with **java.lang.Integer**

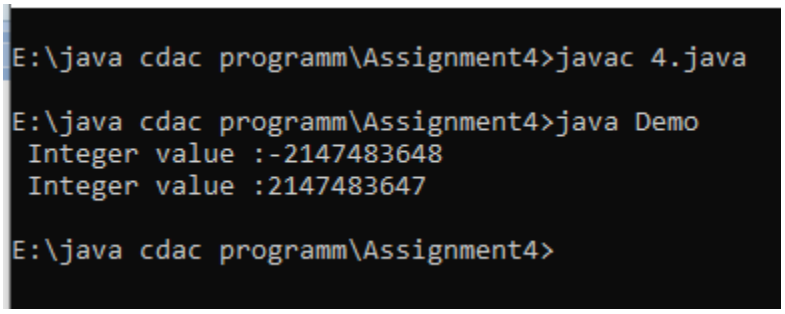
b. Write a program to test how many bytes are used to represent an int value using the BYTES field. (Hint: Use Integer.BYTES).

```
class Demo{

   public static void main(String[] args) {

                byte b = Integer.BYTES;

       System.out.println(" Integer value :" +b);

}}
```

```
E:\java cdac programm\Assignment4>javac 4.java

E:\java cdac programm\Assignment4>java Demo
 String value :4

E:\java cdac programm\Assignment4>
```

c. Write a program to find the minimum and maximum values of int using the MIN_VALUE and MAX_VALUE fields. (Hint: Use Integer.MIN_VALUE and Integer.MAX_VALUE).

```
class Demo{

   public static void main(String[] args) {

                int b = Integer.MIN_VALUE;

                int C = Integer.MAX_VALUE;

      System.out.println(" Integer value :" +b);

                System.out.println(" Integer value :" +C);

}}
```



d. Declare a method-local variable number of type int with some value and convert it to a String using the toString method. (Hint: Use Integer.toString(int)).

```
class Demo{

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

```
        int b = 10;

        String str = Integer.toString(b);

    System.out.println(" String value :" +str);

}}
```

```
E:\java cdac programm\Assignment4>javac 4.java

E:\java cdac programm\Assignment4>java Demo
 String value :10

E:\java cdac programm\Assignment4>_
```

e. Declare a method-local variable strNumber of type String with some value and convert it to an int value using the parseInt method. (Hint: Use Integer.parseInt(String)).

```
class Demo{

   public static void main(String[] args) {

        String strNumber = "12345";

        int b = Integer.parseInt(strNumber);

    System.out.println(" Integer value :" +b);

}}
```

```
E:\java cdac programm\Assignment4>javac 4.java

E:\java cdac programm\Assignment4>java Demo
 Integer value :12345

E:\java cdac programm\Assignment4>
```

f. Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to an int value. (Hint: parseInt method will throw a NumberFormatException).

```
class Demo{

   public static void main(String[] args) {

                String strNumber = "12345";

                int b = Integer.parseInt(strNumber);

      System.out.println(" Integer value :" +b);

}}
```

```
E:\java cdac programm\Assignment4>javac 4.java

E:\java cdac programm\Assignment4>java Demo
 Integer value :12345

E:\java cdac programm\Assignment4>
```

g. Declare a method-local variable number of type int with some value and convert it to the corresponding wrapper class using Integer.valueOf(). (Hint: Use Integer.valueOf(int)).

```
class Demo{

   public static void main(String[] args) {

         int Number = 12345;

         int b = Integer.valueOf(+Number);

          System.out.println(" Integer value :" +b);

}}
```

```
E:\java cdac programm\Assignment4>javac 4.java

E:\java cdac programm\Assignment4>java Demo
 Integer value :12345

E:\java cdac programm\Assignment4>
```

h. Declare a method-local variable strNumber of type String with some integer value and convert it to the corresponding wrapper class using Integer.valueOf(). (Hint: Use Integer.valueOf(String)).

```
class Demo{

   public static void main(String[] args) {

                String  strNumber = "12345";

                int b = Integer.valueOf(strNumber);

      System.out.println(" Integer value :" +b);
```

}}

```
E:\java cdac programm\Assignment4>javac 4.java

E:\java cdac programm\Assignment4>java Demo
 Integer value :12345

E:\java cdac programm\Assignment4>
```

i. Declare two integer variables with values 10 and 20, and add them using a method from the Integer class. (Hint: Use Integer.sum(int, int)).

```
class Demo{

   public static void main(String[] args) {

                int a =10;

                int b =20;

                int result = Integer.sum(10,20);

      System.out.println(" Integer value :" +result);

}}
```

```
E:\java cdac programm\Assignment4>javac 4.java

E:\java cdac programm\Assignment4>java Demo
 Integer value :30

E:\java cdac programm\Assignment4>
```

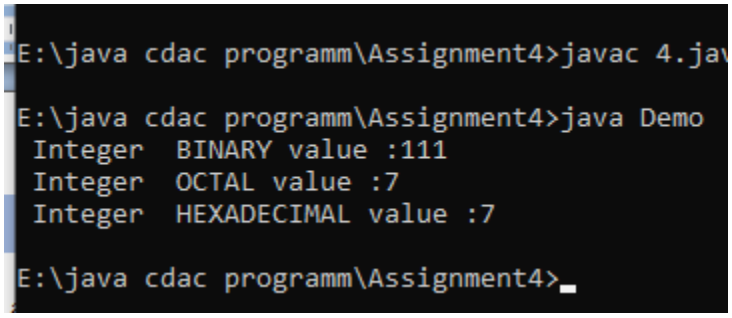
j. Declare two integer variables with values 10 and 20, and find the minimum and maximum values using the Integer class. (Hint: Use Integer.min(int, int) and Integer.max(int, int)).

```
class Demo{

   public static void main(String[] args) {

                int a =10;

                int b =20;

                int Min = Integer.min(10,20);

                int Max = Integer.max(10,20);

      System.out.println(" Integer MINIMUM value :" +Min);

                System.out.println(" Integer  MAXMIUM value :" +Max);

}}
```

```
E:\java cdac programm\Assignment4>java Demo
 Integer  MINIMUM value :10
 Integer  MAXMIUM value :20

E:\java cdac programm\Assignment4>
```

k. Declare an integer variable with the value 7. Convert it to binary, octal, and hexadecimal strings using methods from the Integer class. (Hint: Use Integer.toBinaryString(int), Integer.toOctalString(int), and Integer.toHexString(int)).

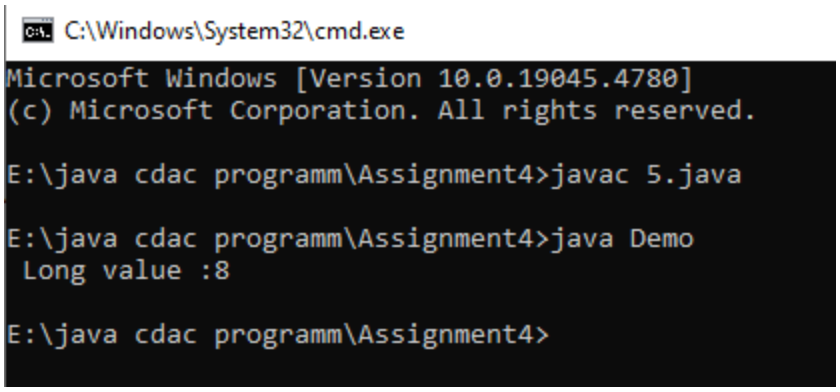
```
class Demo{

    public static void main(String[] args) {

                int number =7;

                String binary = Integer.toBinaryString(number);

                String octal = Integer.toOctalString(number);

                String hexa = Integer.toHexString(number);

        System.out.println(" Integer BINARY value :" +binary);

                System.out.println(" Integer OCTAL value :" +octal);

                System.out.println(" Integer  HEXADECIMAL value :" +hexa);

}}
```



5. **Working with** java.lang.Long

b. Write a program to test how many bytes are used to represent a long value using the BYTES field. (Hint: Use Long.BYTES).

```
class Demo{

   public static void main(String[] args) {

                byte b = Long.BYTES;

      System.out.println(" Long value :" +b);

}}
```



c. Write a program to find the minimum and maximum values of long using the MIN_VALUE and MAX_VALUE fields. (Hint: Use Long.MIN_VALUE and Long.MAX_VALUE).

```
class Demo{

   public static void main(String[] args) {

                long b = Long.MIN_VALUE;

                long c = Long.MAX_VALUE;

      System.out.println(" Long Min value :" +b);
```

```
System.out.println(" Long  MAX value :" +c);

}}
```

```
E:\java cdac programm\Assignment4>javac 5.java

E:\java cdac programm\Assignment4>java Demo
 Long Min value :-9223372036854775808
 Long  MAX value :9223372036854775807

E:\java cdac programm\Assignment4>
```

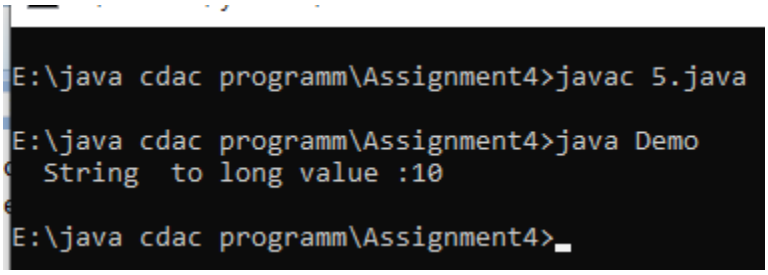
d. Declare a method-local variable number of type long with some value and convert it to a String using the toString method. (Hint: Use Long.toString(long)).

```
class Demo{

   public static void main(String[] args) {

            long b = 10;

            String str = Long.toString(b);

            System.out.println(" Long  String  value :" +str);

}}
```

```
E:\java cdac programm\Assignment4>java Demo
 Long  String  value :10

E:\java cdac programm\Assignment4>
```

e. Declare a method-local variable strNumber of type String with some value and convert it to a long value using the parseLong method. (Hint: Use Long.parseLong(String)).

```
class Demo{

   public static void main(String[] args) {

                String str = "10";

                long b = Long.parseLong(str);

                System.out.println("  String  to long value :" +b);

}}
```



f. Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a long value. (Hint: parseLong method will throw a NumberFormatException).

```
class Demo{

   public static void main(String[] args) {

                String str = "Ab12Cd3";
```

```
        long b = Long.parseLong(str);

        System.out.println("  String  to long value :" +b);

}}
```

```
E:\java cdac programm\Assignment4>javac 5.java

E:\java cdac programm\Assignment4>java Demo
Exception in thread "main" java.lang.NumberFormatException: For input string: "Ab12Cd3"
        at java.base/java.lang.NumberFormatException.forInputString(NumberFormatException.java
        at java.base/java.lang.Long.parseLong(Long.java:711)
        at java.base/java.lang.Long.parseLong(Long.java:836)
        at Demo.main(5.java:22)

E:\java cdac programm\Assignment4>
```

g. Declare a method-local variable number of type long with some value and convert it to the corresponding wrapper class using Long.valueOf(). (Hint: Use Long.valueOf(long)).

```
class Demo{

   public static void main(String[] args) {

        long b  = 100;

        long b1 = Long.valueOf(b);

        System.out.println("  wrapper class value :" +b1);

}}
```

```
E:\java cdac programm\Assignment4>javac 5.java

E:\java cdac programm\Assignment4>java Demo
  wrapper class value :100

E:\java cdac programm\Assignment4>
```

h. Declare a method-local variable strNumber of type String with some long value and convert it to the corresponding wrapper class using Long.valueOf(). (Hint: Use Long.valueOf(String)).

```
class Demo{

   public static void main(String[] args) {

                String strNumber = "1000";

                long b1 = Long.valueOf(strNumber);

                System.out.println("  wrapper class value :" +b1);

}}
```

```
E:\java cdac programm\Assignment4>javac 5.java

E:\java cdac programm\Assignment4>java Demo
  wrapper class value :1000

E:\java cdac programm\Assignment4>
```

i. Declare two long variables with values 1123 and 9845, and add them using a method from the Long class. (Hint: Use Long.sum(long, long)).

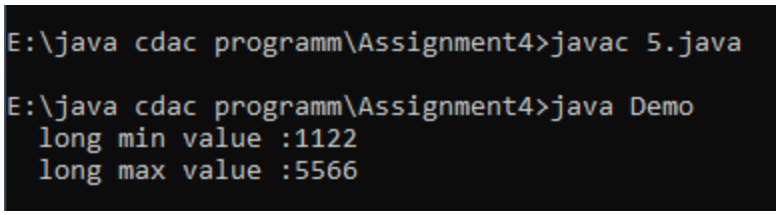
```
class Demo{

   public static void main(String[] args) {

                long b = 1123;

                long b1 = 9845;

                long result = Long.sum(1123,9845);

                System.out.println("  long class sum  :" +result);

}}
```

```
Microsoft Windows [Version 10.0.19045.4780]
(c) Microsoft Corporation. All rights reserved.

E:\java cdac programm\Assignment4>javac 5.java

E:\java cdac programm\Assignment4>java Demo
  long class sum  :10968

E:\java cdac programm\Assignment4>
```

j. Declare two long variables with values 1122 and 5566, and find the minimum and maximum values using the Long class. (Hint: Use Long.min(long, long) and Long.max(long, long)).

```
class Demo{

   public static void main(String[] args) {

                long b = 1122;

                long b1 = 5566;
```

```
                long MIN = Long.min(1122,5566);

                long Max = Long.max(1122,5566);

                System.out.println(" long min value :" +MIN);

System.out.println(" long max value :" +Max);

}}

class Demo{

   public static void main(String[] args) {

                long number = 7;

                String binary = Long.toBinaryString(number);

                String octal = Long.toOctalString(number);

                String hexadecimal = Long.toHexString(number);

                System.out.println("  long Binary value :" +binary);

System.out.println(" long Octal value :" +octal);

System.out.println("  long Hexadecimal value :" +hexadecimal);

}}
```



k. Declare a long variable with the value 7. Convert it to binary, octal, and hexadecimal strings using methods from the Long class. (Hint: Use Long.toBinaryString(long),Long.toOctalString(long), and Long.toHexString(long)).

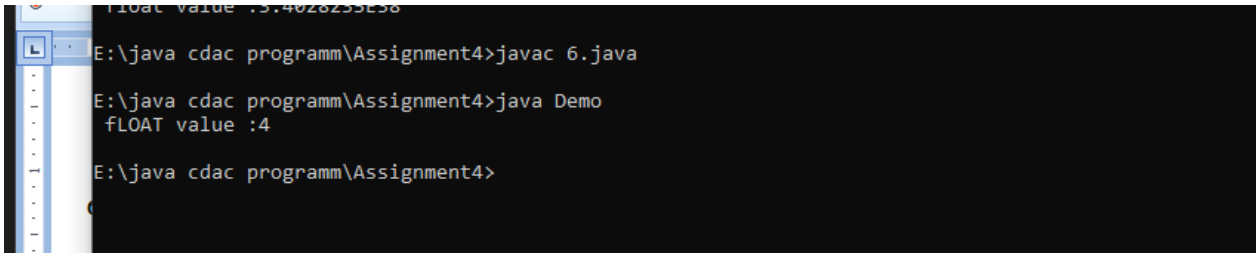
```
class Demo{

    public static void main(String[] args) {

                long number = 7;

                String binary = Long.toBinaryString(number);

                String octal = Long.toOctalString(number);

                String hexadecimal = Long.toHexString(number);

                System.out.println("  long Binary value :" +binary);

System.out.println(" long Octal value :" +octal);

System.out.println("  long Hexadecimal value :" +hexadecimal);

}}
```

```
E:\java cdac programm\Assignment4>java Demo
  long Binary value :111
  long Octal value :7
  long Hexadecimal value :7

E:\java cdac programm\Assignment4>_
```

6. Working with **java.lang.Float**

b. Write a program to test how many bytes are used to represent a float value using the BYTES field. (Hint: Use Float.BYTES).

```
class Demo{

   public static void main(String[] args) {

                byte b = Float.BYTES;

      System.out.println(" Long value :" +b);

}}
```



c. Write a program to find the minimum and maximum values of float using the MIN_VALUE and MAX_VALUE fields. (Hint: Use Float.MIN_VALUE and Float.MAX_VALUE).

```
E:\java cdac programm\Assignment4>javac 6.java

E:\java cdac programm\Assignment4>java Demo
 float Min value :1.4E-45
 float Max value :3.4028235E38

E:\java cdac programm\Assignment4>
```

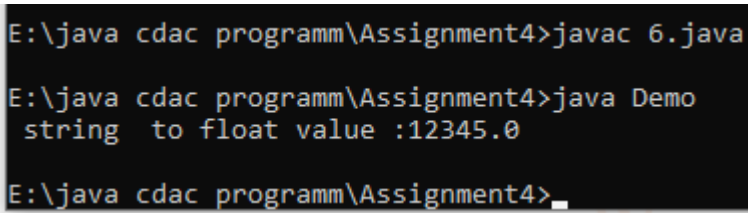
d. Declare a method-local variable number of type float with some value and convert it to a String using the toString method. (Hint: Use Float.toString(float)).

```
class Demo{

   public static void main(String[] args) {

                float b = 12.6f;

                String str = Float.toString(b);

                System.out.println(" float to string value :" +str);

}}
```

```
E:\java cdac programm\Assignment4>javac 6.java

E:\java cdac programm\Assignment4>java Demo
 float to string value :12.6

E:\java cdac programm\Assignment4>
```

e. Declare a method-local variable strNumber of type String with some value and convert it to a float value using the parseFloat method. (Hint: Use Float.parseFloat(String)).

```
class Demo{

   public static void main(String[] args) {

                String str = "12345";

                float b = Float.parseFloat(str);

      //System.out.println(" float Min value :" +b);

                System.out.println(" string  to float value :" +b);

}}
```



f. Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a float value. (Hint: parseFloat method will throw a NumberFormatException).

```
class Demo{

   public static void main(String[] args) {

                String str = "Ab12Cd3";
```

```
        float b = Float.parseFloat(str);

    //System.out.println(" float Min value :" +b);

        System.out.println(" string  to float value :" +b);

}}
```

```
E:\java cdac programm\Assignment4>javac 6.java

E:\java cdac programm\Assignment4>java Demo
Exception in thread "main" java.lang.NumberFormatException: For input string: "Ab12Cd3"
        at java.base/jdk.internal.math.FloatingDecimal.readJavaFormatString(FloatingDecimal.java:2054)
        at java.base/jdk.internal.math.FloatingDecimal.parseFloat(FloatingDecimal.java:122)
        at java.base/java.lang.Float.parseFloat(Float.java:476)
        at Demo.main(6.java:30)
```

g. Declare a method-local variable number of type float with some value and convert it to the corresponding wrapper class using Float.valueOf(). (Hint: Use Float.valueOf(float)).

```
class Demo{

   public static void main(String[] args) {

        float b = 100f;

        float b1 = Float.valueOf(b);

        System.out.println(" float to float value :" +b1);

}}
```

```
E:\java cdac programm\Assignment4>javac 6.java

E:\java cdac programm\Assignment4>java Demo
 float to float value :100.0
```

h. Declare a method-local variable strNumber of type String with some float value and convert it to the corresponding wrapper class using Float.valueOf(). (Hint: Use Float.valueOf(String)).

```
class Demo{

   public static void main(String[] args) {

                String strNumber = "100";

                float b1 = Float.valueOf(strNumber);

                System.out.println(" String to float value :" +b1);

}}
```

```
E:\java cdac programm\Assignment4>javac 6.java

E:\java cdac programm\Assignment4>java Demo
 String to float value :100.0
```

i. Declare two float variables with values 112.3 and 984.5, and add them using a method from the Float class. (Hint: Use Float.sum(float, float)).

```
class Demo{
   public static void main(String[] args) {
            float b = 112.3f;
```

```
        float b1 = 984.5f;
        float result = Float.sum(b, b1);
        System.out.println(" Sum of  float value :" +result);
    }}
```

```
E:\java cdac programm\Assignment4>javac 6.java

E:\java cdac programm\Assignment4>java Demo
 Sum of  float value :1096.8
```

j. Declare two float variables with values 112.2 and 556.6, and find the minimum and maximum values using the Float class. (Hint: Use Float.min(float, float) and Float.max(float, float)).

```
class Demo{

   public static void main(String[] args) {

        float b = 112.2f;

        float b1 = 556.6f;

        float Min = Float.min(112.2f,556.6f);

        float Max = Float.max(112.2f,556.6f);

        System.out.println(" Min float value :" +Min);

        System.out.println(" Max float value :" +Max);

}}
```

```
E:\java cdac programm\Assignment4>javac 6.java

E:\java cdac programm\Assignment4>java Demo
 Min float value :112.2
 Max float value :556.6
```

k. Declare a float variable with the value -25.0f. Find the square root of this value. (Hint: Use Math.sqrt() method).

```
class Demo{

    public static void main(String[] args) {

                float b = -25.0f;

                double squre = Math.sqrt(25.0);

                System.out.println(" Max float value :" +squre);

}}
```

```
E:\java cdac programm\Assignment4>javac 6.java

E:\java cdac programm\Assignment4>java Demo
 Max float value :5.0
```

l. Declare two float variables with the same value, 0.0f, and divide them. (Hint: Observe the result and any special floating-point behavior).

```
class Demo{

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

```
        float b = 0.0f;

        float b1 = 0.0f;

        float div = b/b1;

        System.out.println(" divide float value :" +div);

}}
```

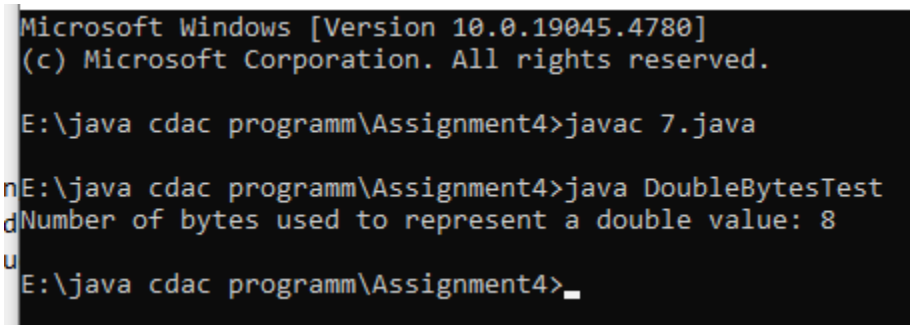
```
E:\java cdac programm\Assignment4>javac 6.java

E:\java cdac programm\Assignment4>java Demo
 divide float value :NaN

E:\java cdac programm\Assignment4>
```

7. **Working with** java.lang.Double

b. Write a program to test how many bytes are used to represent a double value using the BYTES field. (Hint: Use Double.BYTES).

```
public class DoubleBytesTest {

   public static void main(String[] args) {

      int bytes = Double.BYTES;

      System.out.println("Number of bytes used to represent a double value: " + bytes);

   }
```

}



c. Write a program to find the minimum and maximum values of double using the MIN_VALUE and MAX_VALUE fields. (Hint: Use Double.MIN_VALUE and Double.MAX_VALUE).

```
public class DoubleRangeTest {

    public static void main(String[] args) {

        double minValue = Double.MIN_VALUE;

        double maxValue = Double.MAX_VALUE;

        System.out.println("Minimum value of a double: " + minValue);

        System.out.println("Maximum value of a double: " + maxValue);

    }
```

```
E:\java cdac programm\Assignment4>javac 7.java

E:\java cdac programm\Assignment4>java DoubleRangeTest
Minimum value of a double: 4.9E-324
Maximum value of a double: 1.7976931348623157E308

E:\java cdac programm\Assignment4>
```
}

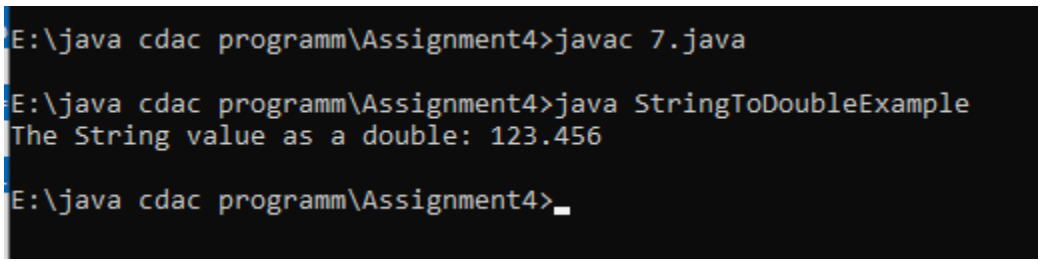
d. Declare a method-local variable number of type double with some value and convert it to a String using the toString method. (Hint: Use Double.toString(double)).

```
class DoubleToStringExample {

   public static void main(String[] args) {

      double number = 123.456;

      String numberAsString = Double.toString(number);

      System.out.println("The double value as a String: " + numberAsString);

   }

}
```

```
E:\java cdac programm\Assignment4>javac 7.java

E:\java cdac programm\Assignment4>java DoubleToStringExample
The double value as a String: 123.456

E:\java cdac programm\Assignment4>
```

e. Declare a method-local variable strNumber of type String with some value and convert it to a double value using the parseDouble method. (Hint: Use Double.parseDouble(String)).

```
public class StringToDoubleExample {

   public static void main(String[] args) {

      String strNumber = "123.456";

     double number = Double.parseDouble(strNumber);

    System.out.println("The String value as a double: " + number);

   }

}
```



f. Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a double value. (Hint: parseDouble method will throw a NumberFormatException).

```
class StringToDouble{
```

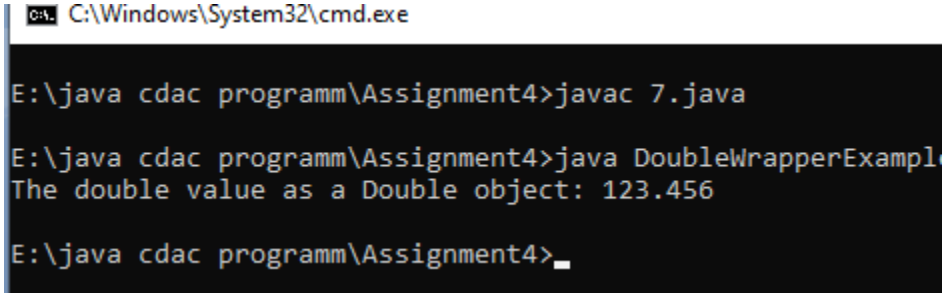
```
    public static void main(String[] args) {

        String str = "Ab12Cd3";

                double b = Double.parseDouble(str);

System.out.println(" string  to Double value :" +b);

    }

}
```

```
E:\java cdac programm\Assignment4>javac 7.java

E:\java cdac programm\Assignment4>java StringToDouble
Exception in thread "main" java.lang.NumberFormatException: For input string: "Ab12Cd3"
        at java.base/jdk.internal.math.FloatingDecimal.readJavaFormatString(FloatingDecimal.java:2054)
        at java.base/jdk.internal.math.FloatingDecimal.parseDouble(FloatingDecimal.java:110)
        at java.base/java.lang.Double.parseDouble(Double.java:651)
        at StringToDouble.main(7.java:39)

E:\java cdac programm\Assignment4>
```

g. Declare a method-local variable number of type double with some value and convert it to the corresponding wrapper class using Double.valueOf(). (Hint: Use Double.valueOf(double)).

```
class DoubleWrapperExample {

    public static void main(String[] args) {

        double number = 123.456;

        Double numberWrapper = Double.valueOf(number);
System.out.println("The double value as a Double object: " +
numberWrapper);
```

```
    }

}
```



h. Declare a method-local variable strNumber of type String with some double value and convert it to the corresponding wrapper class using Double.valueOf(). (Hint: Use Double.valueOf(String)).

```
class StringToDoubleWrapperExample {

    public static void main(String[] args) {

        String strNumber = "123.456";

        Double numberWrapper = Double.valueOf(strNumber);

        System.out.println("The String value as a Double object: " +
numberWrapper);

    }

}
```

```
E:\java cdac programm\Assignment4>java StringToDoubleWrapperExample
The String value as a Double object: 123.456

E:\java cdac programm\Assignment4>javac 7.java
```

i. Declare two double variables with values 112.3 and 984.5, and add them using a method from the Double class. (Hint: Use Double.sum(double, double)).

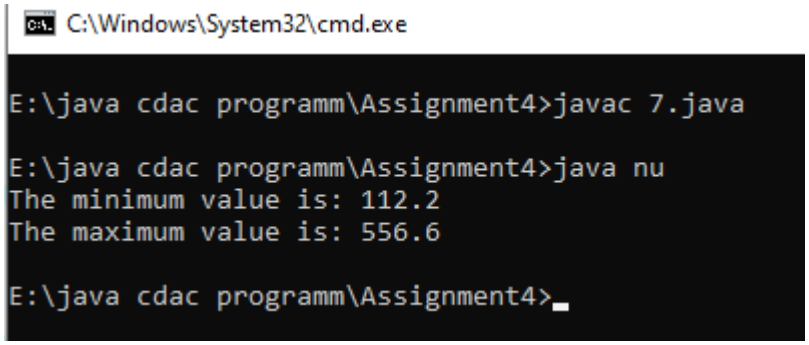
```
class SUM {
   public static void main(String[] args) {
      double num1 = 112.3;
      double num2 = 984.5;
      double sum = Double.sum(num1, num2);
      System.out.println("The sum is: " + sum);
   }
}
```

```
E:\java cdac programm\Assignment4>javac 7.java

E:\java cdac programm\Assignment4>java SUM
The sum is: 1096.8

E:\java cdac programm\Assignment4>
```

j. Declare two double variables with values 112.2 and 556.6, and find the minimum and maximum values using the Double class. (Hint: Use Double.min(double, double) and Double.max(double, double)).

```
class  nu {

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

```
        double num1 = 112.2;

        double num2 = 556.6;

        double minValue = Double.min(num1, num2);

double maxValue = Double.max(num1, num2);

        System.out.println("The minimum value is: " + minValue);

        System.out.println("The maximum value is: " + maxValue);

    }

}
```



k. Declare a double variable with the value -25.0. Find the square root of this value. (Hint: Use Math.sqrt() method).

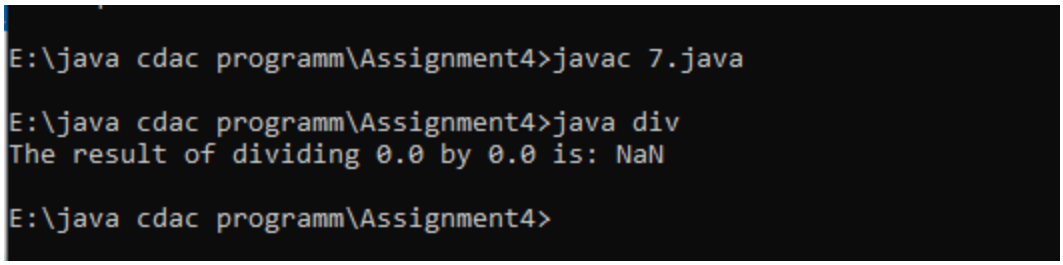
```
class square {

    public static void main(String[] args) {

        double num = -25.0;
```

```
        double sqrtValue = Math.sqrt(num);

        System.out.println("The square root is: " + sqrtValue);

    }

}
```

```
E:\java cdac programm\Assignment4>javac 7.java

E:\java cdac programm\Assignment4>java square
The square root is: NaN

E:\java cdac programm\Assignment4>
```

l. Declare two double variables with the same value, 0.0, and divide them. (Hint: Observe the result and any special floating-point behavior).

```
class div {

    public static void main(String[] args) {

        double num1 = 0.0;

        double num2 = 0.0;

       double result = num1 / num2;

        System.out.println("The result of dividing 0.0 by 0.0 is: " + result);

    }

}
```



8. Conversion between Primitive Types and Strings

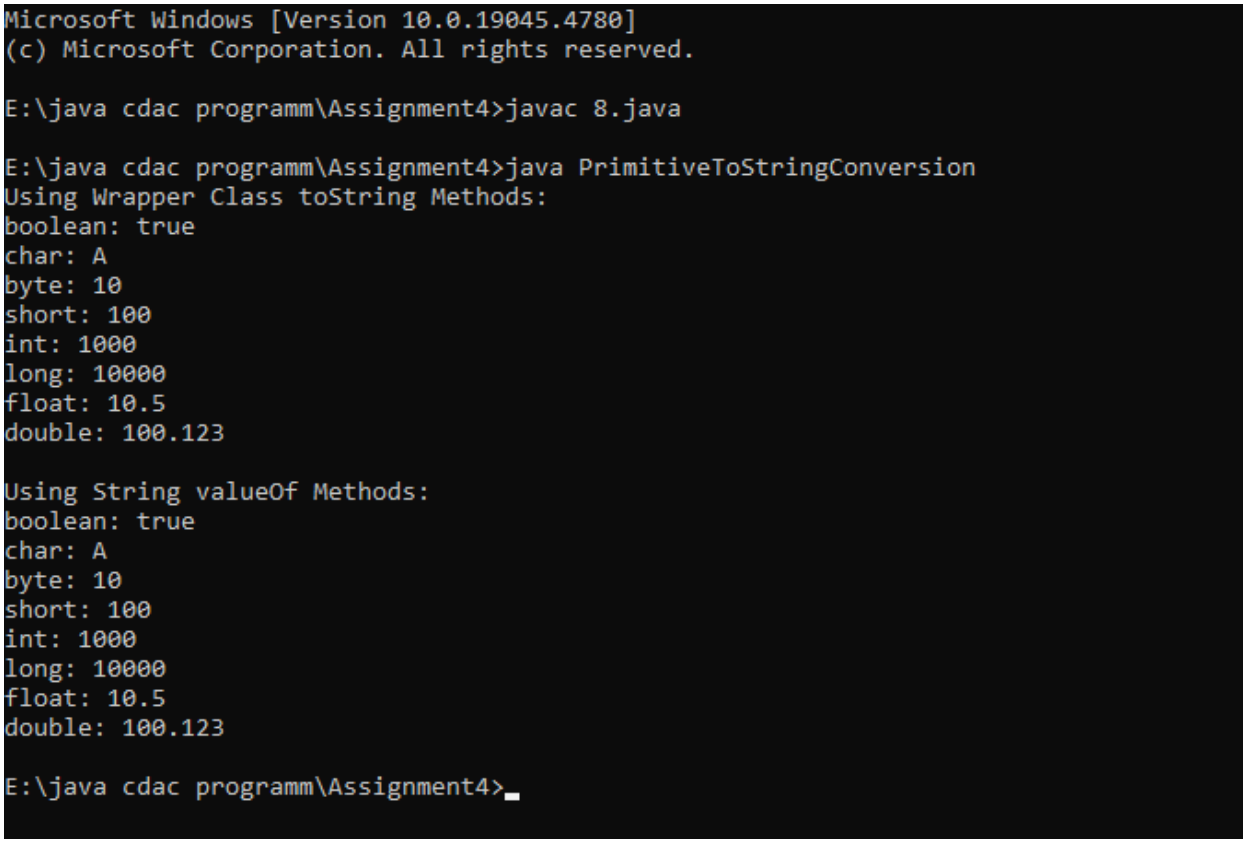
Initialize a variable of each primitive type with a user-defined value and convert it into String:

- First, use the toString method of the corresponding wrapper class. (e.g., Integer.toString()).
- Then, use the valueOf method of the String class. (e.g., String.valueOf()).

```
class PrimitiveToStringConversion {

   public static void main(String[] args) {

      boolean boolVal = true;

      char charVal = 'A';

      byte byteVal = 10;

      short shortVal = 100;

      int intVal = 1000;

      long longVal = 10000L;
```

```
float floatVal = 10.5f;

double doubleVal = 100.123;

System.out.println("Using Wrapper Class toString Methods:");

System.out.println("boolean: " + Boolean.toString(boolVal));

System.out.println("char: " + Character.toString(charVal));

System.out.println("byte: " + Byte.toString(byteVal));

System.out.println("short: " + Short.toString(shortVal));

System.out.println("int: " + Integer.toString(intVal));

System.out.println("long: " + Long.toString(longVal));

System.out.println("float: " + Float.toString(floatVal));

System.out.println("double: " + Double.toString(doubleVal));

System.out.println("\nUsing String valueOf Methods:");

System.out.println("boolean: " + String.valueOf(boolVal));

System.out.println("char: " + String.valueOf(charVal));

System.out.println("byte: " + String.valueOf(byteVal));

System.out.println("short: " + String.valueOf(shortVal));
```

```
        System.out.println("int: " + String.valueOf(intVal));

        System.out.println("long: " + String.valueOf(longVal));

        System.out.println("float: " + String.valueOf(floatVal));

        System.out.println("double: " + String.valueOf(doubleVal));

    }

}
```



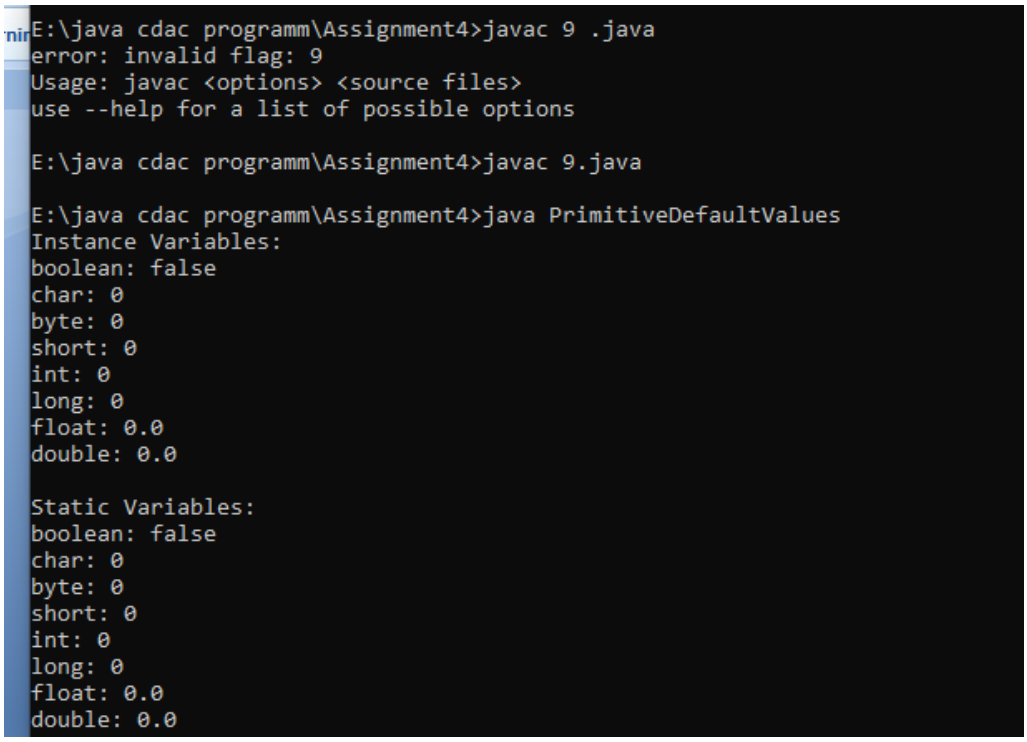
9. Default Values of Primitive Types

Declare variables of each primitive type as fields of a class and check their default values. (Note: Default values depend on whether the variables are instance variables or static variables).

```
class PrimitiveDefaultValues {

   boolean instanceBool;

   char instanceChar;

   byte instanceByte;

   short instanceShort;

   int instanceInt;

   long  instanceLong;

   float instanceFloat;

   double instanceDouble;

   static boolean staticBool;

   static char staticChar;

   static byte staticByte;

   static short staticShort;
```

```
    static int staticInt;

    static long staticLong;

    static float staticFloat;

    static double staticDouble;

    public static void main(String[] args) {

        PrimitiveDefaultValues obj = new PrimitiveDefaultValues();



        System.out.println("Instance Variables:");

        System.out.println("boolean: " + obj.instanceBool);

        System.out.println("char: " + (int) obj.instanceChar); // char defaults to
'\u0000', which is 0

        System.out.println("byte: " + obj.instanceByte);

        System.out.println("short: " + obj.instanceShort);

        System.out.println("int: " + obj.instanceInt);

        System.out.println("long: " + obj.instanceLong);

        System.out.println("float: " + obj.instanceFloat);

        System.out.println("double: " + obj.instanceDouble);
```

```
        System.out.println("\nStatic Variables:");

        System.out.println("boolean: " + staticBool);

        System.out.println("char: " + (int) staticChar); // char defaults to
'\u0000', which is 0

        System.out.println("byte: " + staticByte);

        System.out.println("short: " + staticShort);

        System.out.println("int: " + staticInt);

        System.out.println("long: " + staticLong);

        System.out.println("float: " + staticFloat);

        System.out.println("double: " + staticDouble);

    }

}
```



10. Arithmetic Operations with Command Line Input

Write a program that accepts two integers and an arithmetic operator (+, -, *, /) from the command line. Perform the specified arithmetic operation based on the operator provided. (Hint: Use switch-case for operations).

```
import java.util.Scanner;

public class ArithmeticOperation {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the first integer: ");

        int num1 = scanner.nextInt();
```

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

int num2 = scanner.nextInt();

System.out.print("Enter an arithmetic operator (+, -, *, /): ");

char operator = scanner.next().charAt(0);

double result = 0;

boolean validOperation = true;

switch (operator) {

   case '+':

      result = num1 + num2;

      break;

   case '-':

      result = num1 - num2;

      break;

   case '*':

      result = num1 * num2;

      break;
```

```
    case '/':
        if (num2 == 0) {
            System.out.println("Error: Division by zero is not allowed.");
            validOperation = false;
        } else {
            result = (double) num1 / num2;
        }
        break;
    default:
        System.out.println("Error: Invalid operator. Use +, -, *, or /.");
        validOperation = false;
        break;
}
if (validOperation) {
    System.out.printf("Result: %.2f%n", result);
}
```

```
        scanner.close();

    }

}
```