ASSIGNMENT NO.2

Note: Consider the following before starting the assignment:

- A **static field** declared inside a class is called a **class-level variable**. To access this variable, use the class name and the dot operator (e.g., `Integer.MAX_VALUE`).
- A **static method** defined inside a class is called a **class-level method**. To access this method, use the class name and the dot operator (e.g., `Integer.parseInt()`).
- When accessing static members within the same class, you do not need to use the class name.

1. Working with `java.lang.Boolean`

a. Explore the Java API documentation for `java.lang.Boolean` and observe its modifiers and super types.

There are a three types of modifiers of Boolean.

1. static Boolean : Which have a primitive value is false.
2. static Boolean : Which have a primitive value is true.
3. Static Class<Boolean> : The class object representing the primitive type 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;

public class Assign3_2 {

public static void main(String[] args) {

    Boolean b=true;

   // String s1 = Boolean.toString(  b ) ;

   String s1 = String.valueOf( true );

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

  }

}
```

Output :

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
Boolean value is:true
```

c. Declare a method-local variable `strStatus` of type `String` with the value `"true"` and convert it to a `boolean` using the `parseBoolean` method. (Hint: Use `Boolean.parseBoolean(String)`).

```
String strStatus = "true";
        boolean status = Boolean.parseBoolean(strStatus);
        System.out.println("Boolean value: " + status);
```

Output

```
PROBLEMS 4  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
Boolean value: true
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

d. 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"`).

```
String strStatus = "1";
         boolean status = "1".equals(strStatus);
        System.out.println("Boolean value: " + status);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
Boolean value: true
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

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)`).

```
boolean status = true;
        Boolean wrapperValue = Boolean.valueOf(status);
        System.out.println("Wrapper class value: " + wrapperValue);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
Wrapper class value: true
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

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)`).

```
String strStatus = "true";
        Boolean wrapperValue = Boolean.valueOf(strStatus);
        System.out.println("Wrapper class value: " + wrapperValue);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
Wrapper class value: true
```

g. Experiment with converting a `boolean` value into other primitive types or vice versa and observe the results.

```
import java.lang.Boolean;
public class Assign3_2 {
    public static void main(String[] args) {
       // Q.b) Boolean b=true;
       // String s1 = Boolean.toString(  b ) ;
       String s1 = String.valueOf( true );
        System.out.println("Boolean value is:" +s1);

       // Q.c
        String strStatus = "true";
        boolean status = Boolean.parseBoolean(strStatus);
        System.out.println("Boolean value: " + status);
       // Q.d)
        //String strStatus = "1";
        status = "1".equals(strStatus);
        System.out.println("Boolean value: " + status);
       // Q.e
          status = true;
         Boolean wrapperValue = Boolean.valueOf(status);
```

```
        System.out.println("Wrapper class value: " + wrapperValue);
       // Q.f)
      strStatus = "true";
        wrapperValue = Boolean.valueOf(strStatus);
        System.out.println("Wrapper class value: " + wrapperValue);


    }
    }
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
Boolean value is:true
Boolean value: true
Boolean value: false
Wrapper class value: true
Wrapper class value: true
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

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`).

```
public class Assign3_2 {
    public static void main(String[] args) {

                byte b = Byte.BYTES;
                System.out.println("bytes value :" + b);


    }
}
```

```
S F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
S F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
ytes value :1
```

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`).

```
byte b = Byte.MIN_VALUE;
        byte c = Byte.MAX_VALUE;
        System.out.println(" bytes min value :" + b);
    System.out.println(" bytes max value :" + c);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
 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)`).

```
byte number =7;
            String str = Byte.toString(number);
            System.out.println(" String value :" +str);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
 String value :7
```

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)`).

```
String s = "100";
        byte b = Byte.parseByte(s);
        System.out.println(" String value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
 String value :100
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

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`).

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
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.Byte.parseByte(Byte.java:193)
        at java.base/java.lang.Byte.parseByte(Byte.java:219)
        at Assign3_2.main(Assign3_2.java:18)
```

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)`).

```
byte Number = 100;
byte b = Byte.valueOf(Number);
System.out.println(" String value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
 String value :100
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

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)`).

```
String strNumber = "30";
byte b = Byte.valueOf(strNumber);
System.out.println(" String value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_2.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_2
 String value :30
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

3. Working with `java.lang.Short`

a. Explore the Java API documentation for `java.lang.Short` and observe its modifiers and super types.

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

```
byte b = Short.BYTES;
System.out.println(" String value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_3.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_3
 String value :2
```

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`).

```
Short b = Short.MIN_VALUE;
Short c = Short.MAX_VALUE;
System.out.println(" Short Min value :" +b);
System.out.println(" Short Max value :" +c);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_3.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_3
 Short Min value :-32768
 Short Max value :32767
```

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)`).

```
    Short b = 2;
    String str = Short.toString(b);
    System.out.println(" String value :" +str);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_3.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_3
 String value :2
```

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)`).

```
    String str = "70";
    Short s = Short.parseShort(str);
    System.out.println(" Short  value :" +s);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_3.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_3
 Short  value :70
```

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`).

```
        String str = "Ab12Cd3";
        Short s = Short.parseShort(str);
        System.out.println(" Short value :" +s);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_3.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_3
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 Assign3_3.main(Assign3_3.java:23)
```

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)`).

```
        Short s= 55;
         Short s1 = Short.valueOf(s);
         System.out.println(" Short value :" +s1);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_3.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_3
 Short value :55
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

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)`).

```
         Short s= 60;
        Short s1 = Short.valueOf(s);
        System.out.println(" Short value :" +s1);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_3.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_3
Short value :60
```

4. Working with `java.lang.Integer`

a. Explore the Java API documentation for `java.lang.Integer` and observe its modifiers and super types.

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

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

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_4.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_4
 Integer value :4
```

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`).

```
        int b = Integer.MIN_VALUE;
        int C = Integer.MAX_VALUE;
        System.out.println(" Integer value :" +b);
        System.out.println(" Integer value :" +C);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_4.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_4
 Integer value :-2147483648
 Integer value :2147483647
```

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)`).

```
      int b = 10;
      String str = Integer.toString(b);
      System.out.println(" String value :" +str);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_4.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_4
 String value :10
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

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)`).

```
        String strNumber = "100243";
         int b = Integer.parseInt(strNumber);
         System.out.println(" Integer value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_4.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_4
 Integer value :100243
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

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`).

```
            String strNumber = "12345";
            int b = Integer.parseInt(strNumber);
            System.out.println(" Integer value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_4.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_4
 Integer value :12345
```

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)`).

```
int Number = 147852;
int b = Integer.valueOf(+Number);
System.out.println(" Integer value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_4.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_4
 Integer value :147852
```

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)`).

```
    String  strNumber = "75230";
    int b = Integer.valueOf(strNumber);
    System.out.println(" Integer value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_4.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_4
 Integer value :75230
```

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)`).

```
            int a =10;
            int b =20;
            int result = Integer.sum(10,20);
            System.out.println(" Integer value :" +result);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_4.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_4
 Integer value :30
```

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)`).

```
    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);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_4.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_4
 Integer  MINIMUM value :10
 Integer  MAXMIUM value :20
```

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)`).

```
        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);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_4.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_4
 Integer  BINARY value :111
 Integer  OCTAL value :7
 Integer  HEXADECIMAL value :7
```

5. Working with `java.lang.Long`

a. Explore the Java API documentation for `java.lang.Long` and observe its modifiers and super types.

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

```
       byte b = Long.BYTES;
        System.out.println(" Long value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_5.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_5
 Long value :8
```

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`).

```
        long b = Long.MIN_VALUE;
        long c = Long.MAX_VALUE;
        System.out.println(" Long Min value :" +b);
        System.out.println(" Long MAX value :" +c);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_5.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_5
 Long Min value :-9223372036854775808
 Long MAX value :9223372036854775807
```

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)`).

```
    long b = 10;
    String str = Long.toString(b);
    System.out.println(" Long String value :" +str);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_5.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_5
 Long Min value :-9223372036854775808
 Long MAX value :9223372036854775807
```

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)`).

```
String str = "20";
long b = Long.parseLong(str);
System.out.println(" String to long value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_5.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_5
 String to long value :20
```

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`).

```
        String str = "Ab12Cd3";
        long b = Long.parseLong(str);
        System.out.println(" String to long value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_5.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_5
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.Long.parseLong(Long.java:711)
        at java.base/java.lang.Long.parseLong(Long.java:836)
        at Assign3_5.main(Assign3_5.java:22)
```

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)`).

```
long b = 100;
long b1 = Long.valueOf(b);
System.out.println(" wrapper class value :" +b1);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_5.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_5
 wrapper class value :100
```

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)`).

```
    String strNumber = "2000";
    long b1 = Long.valueOf(strNumber);
    System.out.println(" wrapper class value :" +b1);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_5.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_5
 wrapper class value :2000
```

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)`).

```
        long b = 1123;
        long b1 = 9845;
        long result = Long.sum(1123,9845);
        System.out.println(" long class sum :" +result);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_5.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_5
 long class sum :10968
```

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)`).

```
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);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_5.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_5
 long min value :1122
 long max value :5566
```

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)`).

```
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);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_5.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_5
 long Binary value :111
 long Octal value :7
 long Hexadecimal value :7
```

l. Experiment with converting a `long` value into other primitive types or vice versa and observe the results.

6. Working with `java.lang.Float`

a. Explore the Java API documentation for `java.lang.Float` and observe its modifiers and super types.

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

```
byte b = Float.BYTES;
System.out.println(" Long value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_6.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_6
 Long value :4
```

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`).

```
      float minValue = Float.MIN_VALUE;
        float maxValue = Float.MAX_VALUE;
        System.out.println("Minimum float value: " + minValue);
        System.out.println("Maximum float value: " + maxValue);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_6.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_6
Minimum float value: 1.4E-45
Maximum float value: 3.4028235E38
```

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)`).

```
      float b = 12.6f;
        String str = Float.toString(b);
        System.out.println(" float to string value :" +str);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_6.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_6
 float to string value :12.6
```

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)`).

```
        String str = "2021";
        float b = Float.parseFloat(str);
        System.out.println(" string to float value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_6.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_6
 string to float value :2021.0
```

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`).

```
String str = "Ab12Cd3";
 float b = Float.parseFloat(str);
 //System.out.println(" float Min value :" +b);
  System.out.println(" string to float value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_6.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_6
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 Assign3_6.main(Assign3_6.java:21)
```

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)`).

```
      float b = 150f;
        float b1 = Float.valueOf(b);
        System.out.println(" float to float value :" +b1);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_6.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_6
 float to float value :150.0
```

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)`).

```
      String strNumber = "80";
        float b1 = Float.valueOf(strNumber);
        System.out.println(" String to float value :" +b1);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_6
 String to float value :80.0
```

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)`).

```
 float b = 112.3f;
  float b1 = 984.5f;
```

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

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_6.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_6
 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)`).

```
      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);
```

```
S F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_6.java
S F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_6
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).

```
      float b = -25.0f;
        double squre = Math.sqrt(25.0);
        System.out.println(" Max float value :" +squre);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_6.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_6
 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).

```
 float b = 0.0f;
 float b1 = 0.0f;
float div = b/b1;
 System.out.println(" divide float value :" +div);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_6.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_6
 divide float value :NaN
```

m. Experiment with converting a `float` value into other primitive types or vice versa and observe the results.

7. Working with `java.lang.Double`

a. Explore the Java API documentation for `java.lang.Double` and observe its modifiers and super types.

```
      int bytes = Double.BYTES;
      System.out.println("Number of bytes used to represent a double
value: " + bytes);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
Number of bytes used to represent a double value: 8
```

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

```
    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);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
Minimum value of a double: 4.9E-324
Maximum value of a double: 1.7976931348623157E308
```

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`).

```
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);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
Minimum value of a double: 4.9E-324
Maximum value of a double: 1.7976931348623157E308
```

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)`).

```
double number = 123.456;

    String numberAsString = Double.toString(number);

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

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
The double value as a String: 123.456
```

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)`).

```
    String strNumber = "123.456";

    double number = Double.parseDouble(strNumber);

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

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
The String value as a double: 123.456
```

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`).

```
      String str = "Ab12Cd3";
    double b = Double.parseDouble(str);
    System.out.println(" string to Double value :" +b);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
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 Assign3_7.main(Assign3_7.java:35)
```

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)`).

```
    double number = 152.30;
    Double numberWrapper = Double.valueOf(number);
    System.out.println("The double value as a Double object: " +
    numberWrapper);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
The double value as a Double object: 152.3
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

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)`).

```
    String strNumber = "123.456";
    Double numberWrapper = Double.valueOf(strNumber);
    System.out.println("The String value as a Double object: " +
    numberWrapper);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
The String value as a Double object: 123.456
```

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)`).

```
    double num1 = 112.3;
    double num2 = 984.5;
    double sum = Double.sum(num1, num2);
    System.out.println("The sum is: " + sum);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
The sum is: 1096.8
```

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)`).

```
    double num1 = 100.2;
    double num2 = 200.6;
    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);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
The minimum value is: 100.2
The maximum value is: 200.6
```

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

```
    double num = -25.0;
    double sqrtValue = Math.sqrt(num);
    System.out.println("The square root is: " + sqrtValue);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
The square root is: NaN
```

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

```
    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);
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_7.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_7
The result of dividing 0.0 by 0.0 is: NaN
```

m. Experiment with converting a `double` value into other primitive types or vice versa and observe the results.

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()`).

```
public class Assign3_8 {
    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));
    }
}
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\Assign3_8.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java Assign3_8
Using Wrapper Class toString Methods:
boolean: true
char: A
byte: 10
short: 100
int: 1000
long: 10000
float: 10.5
double: 100.123

Using String valueOf Methods:
boolean: true
char: A
byte: 10
short: 100
int: 1000
long: 10000
float: 10.5
double: 100.123
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```

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).

```java
public 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);
    }
}
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\PrimitiveDefaultValues.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java PrimitiveDefaultValues
Instance Variables:
boolean: false
char: 0
byte: 0
short: 0
int: 0
long: 0
float: 0.0
double: 0.0

Static Variables:
boolean: false
char: 0
byte: 0
short: 0
int: 0
long: 0
float: 0.0
double: 0.0
```

10. Arithmetic Operations with Command Line Input

ASSIGNMENT NO.2

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.*;
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();
    }
}
```

```
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> javac .\ArithmeticOperation.java
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java ArithmeticOperation
Enter the first integer: 12
Enter the second integer: 20
Enter an arithmetic operator (+, -, *, /): +
Result: 32.00
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment> java ArithmeticOperation
Enter the first integer: 20
Enter the second integer: 20
Enter an arithmetic operator (+, -, *, /): *
Result: 400.00
PS F:\C-DAC Kharghar\OOP_JAVA\Lab_Assignment>
```