

ASSIGNMENT NO . 04

1. Working with `java.lang.Boolean`

a. Explore the [Java API documentation for `java.lang.Boolean`](#) and observe its modifiers and super types.

The Boolean is a primitive data type, its default value is false.

It has two modifiers:

1. `public`
2. `final`

Object is the super class of all the java classes. Boolean directly extends `java.lang.Object`

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)`).

CODE:

```
public class P1{  
  
    public static void main(String[] args) {  
  
        boolean status = true;  
  
        String statusString = Boolean.toString(status);  
  
        System.out.println(statusString);  
  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P1.java  
PS D:\dac\OOPJ\Day3\Practice> java P1  
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)`).

CODE:

```
public class P1{

    public static void main(String[] args) {

        String strStatus = "true";

        boolean status = Boolean.parseBoolean(strStatus);

        System.out.println(status);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P1.java
PS D:\dac\OOPJ\Day3\Practice> java P1
true
```

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"`).

CODE:

```
public class P1{

    public static void main(String[] args) {

        String strStatus = "0";

        boolean status = Boolean.parseBoolean(strStatus);

        System.out.println(status);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P1.java
PS D:\dac\OOPJ\Day3\Practice> java P1
false
```

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)`).

CODE:

```
public class P1{

    public static void main(String[] args) {

        boolean status = true;

        Boolean wapStatus = Boolean.valueOf(status);

        System.out.println(wapStatus);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P1.java
PS D:\dac\OOPJ\Day3\Practice> java P1
true
```

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)`).

CODE:

```
public class P1{

    public static void main(String[] args) {

        String strString = "true";

        boolean status = Boolean.valueOf(strString);

        System.out.println(status);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P1.java
PS D:\dac\OOPJ\Day3\Practice> java P1
true
```

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

CODE:

Boolean to Integer

```
public class P1{

    public static void main(String[] args) {

        boolean status = false;

        int iStatus = status ? 1 : 0 ;

        System.out.println(iStatus);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P1.java
PS D:\dac\OOPJ\Day3\Practice> java P1
0
```

Int to Boolean

```
public class P1{

    public static void main(String[] args) {

        int iStatus = 1;

        boolean status = (iStatus == 1);

        System.out.println(status);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P1.java
PS D:\dac\OOPJ\Day3\Practice> java P1
true
```

Boolean to Double

```
public class P1{

    public static void main(String[] args) {

        boolean status = true;

        double dStatus = status ? 1.0 : 0.0 ;

        System.out.println(dStatus);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P1.java
PS D:\dac\OOPJ\Day3\Practice> java P1
1.0
```

Double to Boolean

CODE:

```
public class P1{

    public static void main(String[] args) {

        double dStatus = 1.0;

        boolean status = (dStatus == 1.0);

        System.out.println(status);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P1.java
PS D:\dac\OOPJ\Day3\Practice> java P1
true
```

2. Working with `java.lang.Byte`

- a. Explore the [Java API documentation for `java.lang.Byte`](#) and observe its modifiers and super types.

The byte is a primitive data type, its default value is zero.

It has two modifiers:

1. public
2. final

Object is the super class of all the java classes. Byte directly extends `java.lang.Number`.

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

CODE:

```
public class P2 {

    public static void main(String[] args) {

        int b = Byte.BYTES;

        System.out.println(b);

    }
```

```
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P2.java
PS D:\dac\OOPJ\Day3\Practice> java P2
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`).

CODE:

```
public class P2 {

    public static void main(String[] args) {

        byte min = Byte.MIN_VALUE;

        byte max = Byte.MAX_VALUE;

        System.out.println("Minimum value is: "+min);

        System.out.println("Maximun value is: "+max);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P2.java
PS D:\dac\OOPJ\Day3\Practice> java P2
Minimum value is: -128
Maximun value is: 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)`).

CODE:

```
public class P2 {

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

```

        byte number = 5;

        String sNum = Byte.toString(number);

        System.out.println(sNum);

    }

}

```

OUTPUT:

```

Maximum value is: 127
PS D:\dac\OOPJ\Day3\Practice> javac P2.java
PS D:\dac\OOPJ\Day3\Practice> java P2
5

```

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)`).

CODE:

```

public class P2 {

    public static void main(String[] args) {

        String strNumber = "5";

        byte number = Byte.parseByte(strNumber);

        System.out.println(number);

    }

}

```

OUTPUT:

```

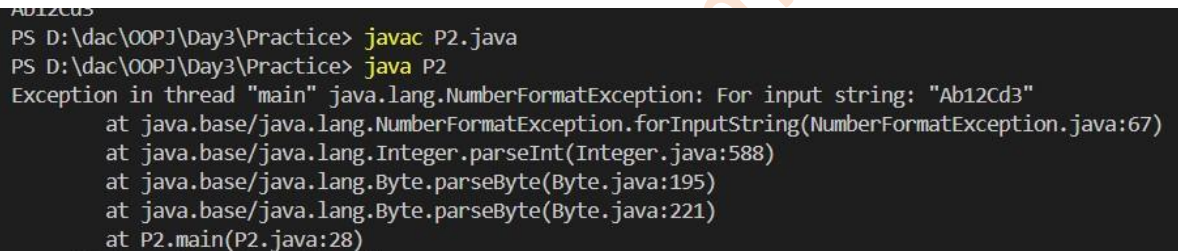
PS D:\dac\OOPJ\Day3\Practice> javac P2.java
PS D:\dac\OOPJ\Day3\Practice> java P2
5

```

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`).

CODE:

```
public class P2 {  
  
    public static void main(String[] args) {  
  
        String strNumber = "Ab12Cd3";  
  
        byte number = Byte.parseByte(strNumber);  
  
        System.out.println(number);  
  
    }  
}
```

OUTPUT:

The screenshot shows a terminal window with the following text:

```
Ab12Cd3  
PS D:\dac\OOPJ\Day3\Practice> javac P2.java  
PS D:\dac\OOPJ\Day3\Practice> java P2  
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:588)  
    at java.base/java.lang.Byte.parseByte(Byte.java:195)  
    at java.base/java.lang.Byte.parseByte(Byte.java:221)  
    at P2.main(P2.java:28)
```

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)`).

CODE:

```
public class P2 {  
  
    public static void main(String[] args) {  
  
        byte number = 5;  
  
        Byte num1 = Byte.valueOf(number);  
  
        System.out.println(num1);  
  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P2.java
PS D:\dac\OOPJ\Day3\Practice> java P2
5
```

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)`).

CODE:

```
public class P2 {

    public static void main(String[] args) {

        String strNumber = "5";

        byte number = Byte.valueOf(strNumber);

        System.out.println(number);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P2.java
PS D:\dac\OOPJ\Day3\Practice> java P2
5
```

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

CODE:

```
public class P2 {

    public static void main(String[] args) {

        byte number = 10;

        int intNumber = (int) number;
```

```
System.out.println("Byte to int: " + intNumber);

short shortNumber = (short) number;

System.out.println("Byte to short: " + shortNumber);

long longNumber = (long) number;

System.out.println("Byte to long: " + longNumber);

float floatNumber = (float) number;

System.out.println("Byte to float: " + floatNumber);

double doubleNumber = (double) number;

System.out.println("Byte to double: " + doubleNumber);
```

```
int iNumber = 20;

byte bInt = (byte) iNumber;

System.out.println("Int to byte: " + bInt);

short sNumber = 20;

byte bShort = (byte) sNumber;

System.out.println("Short to byte: " + bShort);

long lNumber = 20;

byte bLong = (byte) lNumber;

System.out.println("Long to byte: " + bLong);

float fNumber = 20.0f;

byte bFloat = (byte) fNumber;

System.out.println("Int to byte: " + bFloat);

double dNumber = 20.0;

byte bDouble = (byte) dNumber;
```

```
        System.out.println("Int to byte: " + bDouble);  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P2.java  
PS D:\dac\OOPJ\Day3\Practice> java P2  
Byte to int: 10  
Byte to short: 10  
Byte to long: 10  
Byte to float: 10.0  
Byte to double: 10.0  
Int to byte: 20  
Short to byte: 20  
Long to byte: 20  
Int to byte: 20  
Int to byte: 20
```

3. Working with `java.lang.Short`

a. Explore the [Java API documentation for `java.lang.Short`](#) and observe its modifiers and super types.

The short is a primitive data type, its default value is zero.

It has two modifiers:

1. public
2. final

Object is the super class of all the java classes. Byte directly extends `java.lang.Number`.

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

CODE:

```

public class P3 {

    public static void main(String[] args) {

        int number = Short.BYTES;

        System.out.println(number);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P3.java
PS D:\dac\OOPJ\Day3\Practice> java P3
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`).

CODE:

```

public class P3 {

    public static void main(String[] args) {

        short min = Short.MIN_VALUE;

        short max = Short.MAX_VALUE;

        System.out.println("Minimum value is: "+min);

        System.out.println("Maximun value is: "+max);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P3.java
PS D:\dac\OOPJ\Day3\Practice> java P3
Minimum value is: -32768
Maximun value is: 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)`).

CODE:

```
public class P3 {  
  
    public static void main(String[] args) {  
  
        short number = 10;  
  
        String sNum = Short.toString(number);  
  
        System.out.println(sNum);  
  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P3.java  
PS D:\dac\OOPJ\Day3\Practice> java P3  
10
```

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)`).

CODE:

```
public class P3 {  
  
    public static void main(String[] args) {  
  
        String strNumber = "10";  
  
        short number = Short.parseShort(strNumber);  
  
        System.out.println(number);  
  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P3.java
PS D:\dac\OOPJ\Day3\Practice> java P3
10
```

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`).

CODE:

```
public class P3 {

    public static void main(String[] args) {

        String strNumber = "Ab12Cd3";

        short number = Short.parseShort(strNumber);

        System.out.println(number);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P3.java
PS D:\dac\OOPJ\Day3\Practice> java P3
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:588)
    at java.base/java.lang.Short.parseShort(Short.java:138)
    at java.base/java.lang.Short.parseShort(Short.java:164)
    at P3.main(P3.java:25)
```

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)`).

CODE:

```
public class P3 {

    public static void main(String[] args) {

        short number = 10;

        short num1 = Short.valueOf(number);

    }

}
```

```
        System.out.println(num1);
    }
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P3.java
PS D:\dac\OOPJ\Day3\Practice> java P3
10
```

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)`).

CODE:

```
public class P3 {

    public static void main(String[] args) {

        String strNumber = "10";

        short number = Short.valueOf(strNumber);

        System.out.println(number);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P3.java
PS D:\dac\OOPJ\Day3\Practice> java P3
10
```

i. Experiment with converting a short value into other primitive types or vice versa and observe the results.

CODE:

```
public class P3 {

    public static void main(String[] args) {

        short number = 10;
```



```
int intNumber = (int) number;

System.out.println("Short to int: " + intNumber);

byte shortNumber = (byte) number;

System.out.println("Short to Byte: " + shortNumber);

long longNumber = (long) number;

System.out.println("Short to long: " + longNumber);

float floatNumber = (float) number;

System.out.println("Short to float: " + floatNumber);

double doubleNumber = (double) number;

System.out.println("Short to double: " + doubleNumber);
```

```
int iNumber = 20;

short bInt = (short) iNumber;

System.out.println("Int to Short: " + bInt);

byte sNumber = 20;

short bShort = (short) sNumber;

System.out.println("Byte to Short: " + bShort);

long lNumber = 20;

short bLong = (short) lNumber;

System.out.println("Long to Short: " + bLong);

float fNumber = 20.0f;

short bFloat = (short) fNumber;

System.out.println("Int to Short: " + bFloat);

double dNumber = 20.0;
```

```

        short bDouble = (short) dNumber;

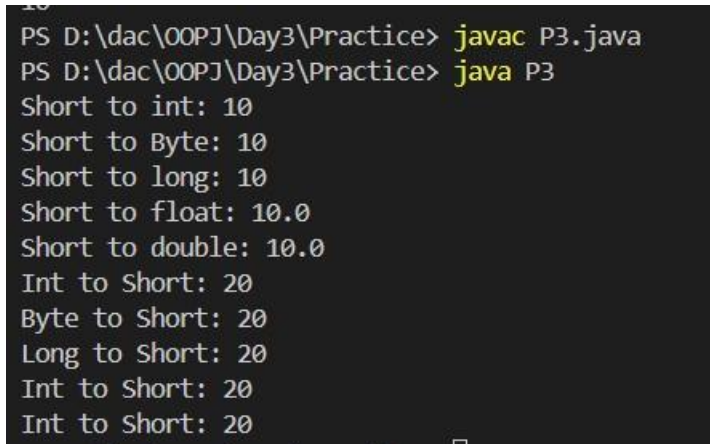
        System.out.println("Int to Short: " + bDouble);

    }

}

```

OUTPUT:



```

PS D:\dac\OOPJ\Day3\Practice> javac P3.java
PS D:\dac\OOPJ\Day3\Practice> java P3
Short to int: 10
Short to Byte: 10
Short to long: 10
Short to float: 10.0
Short to double: 10.0
Int to Short: 20
Byte to Short: 20
Long to Short: 20
Int to Short: 20
Int to Short: 20

```

4. Working with `java.lang.Integer`

- Explore the [Java API documentation for `java.lang.Integer`](#) and observe its modifiers and super types.
- Write a program to test how many bytes are used to represent an `int` value using the `BYTES` field. (Hint: Use `Integer.BYTES`).

CODE:

```

public class P4 {

    public static void main(String[] args) {

        int bytes = Integer.BYTES;

        System.out.println(bytes);

    }

}

```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P4.java
PS D:\dac\OOPJ\Day3\Practice> java P4
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`).

CODE:

```
public class P4 {

    public static void main(String[] args) {

        int min = Integer.MIN_VALUE;

        int max = Integer.MAX_VALUE;

        System.out.println("Minimum value is: "+min);

        System.out.println("Maximun value is: "+max);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P4.java
PS D:\dac\OOPJ\Day3\Practice> java P4
Minimum value is: -2147483648
Maximun value is: 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)`).

CODE:

```
public class P4 {

    public static void main(String[] args) {

        int number = 10;

        String sNum = Integer.toString(number);

        System.out.println(sNum);

    }

}
```

```
}  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P4.java  
PS D:\dac\OOPJ\Day3\Practice> java P4  
10
```

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)`).

CODE:

```
public class P4 {  
  
    public static void main(String[] args) {  
  
        String strNumber = "10";  
  
        int number = Integer.parseInt(strNumber);  
  
        System.out.println(number);  
  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P4.java  
PS D:\dac\OOPJ\Day3\Practice> java P4  
10
```

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`).

CODE:

```
public class P4 {  
  
    public static void main(String[] args) {  
  
        String strNumber = "Ab12Cd3";
```

```

        int number = Integer.parseInt(strNumber);

        System.out.println(number);

    }

}

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P4.java
PS D:\dac\OOPJ\Day3\Practice> java P4
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:588)
    at java.base/java.lang.Integer.parseInt(Integer.java:685)
    at P4.main(P4.java:26)

```

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)`).

CODE:

```

public class P4 {

    public static void main(String[] args) {

        int number = 10;

        int num1 = Integer.valueOf(number);

        System.out.println(num1);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P4.java
PS D:\dac\OOPJ\Day3\Practice> java P4
10

```

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)`).

CODE:

```
public class P4 {  
  
    public static void main(String[] args) {  
  
        String strNumber = "10";  
  
        int number = Integer.valueOf(strNumber);  
  
        System.out.println(number);  
  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P4.java  
PS D:\dac\OOPJ\Day3\Practice> java P4  
10
```

- 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)`).

CODE:

```
public class P4 {  
  
    public static void main(String[] args) {  
  
        int n1 = 10;  
  
        int n2 = 20;  
  
        int sum = Integer.sum(n1, n2);  
  
        System.out.println(sum);  
  
    }  
}
```

OUTPUT:

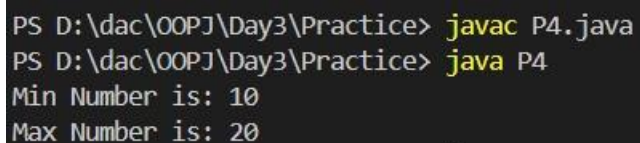
```
PS D:\dac\OOPJ\Day3\Practice> javac P4.java  
PS D:\dac\OOPJ\Day3\Practice> java P4  
30
```

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)`).

CODE:

```
public class P4 {  
  
    public static void main(String[] args) {  
  
        int n1 = 10;  
  
        int n2 = 20;  
  
        int min = Integer.min(n1, n2);  
  
        int max = Integer.max(n2, n2);  
  
        System.out.println("Min Number is: " + min );  
  
        System.out.println("Max Number is: " + max );  
  
    }  
}
```

OUTPUT:



```
PS D:\dac\OOPJ\Day3\Practice> javac P4.java  
PS D:\dac\OOPJ\Day3\Practice> java P4  
Min Number is: 10  
Max Number is: 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)`).

CODE:

```
public class P4 {  
  
    public static void main(String[] args) {  
  
        int num = 7;  
  
        String bin = Integer.toBinaryString(num);
```

```

String oct = Integer.toOctalString(num);

String hex = Integer.toHexString(num);

System.out.println("Int to Binary : " + bin);

System.out.println("Int to Octal : " + oct);

System.out.println("Int to Hexadecimal : " + hex);

}

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P4.java
PS D:\dac\OOPJ\Day3\Practice> java P4
Int to Binary : 111
Int to Octal : 7
Int to Hexadecimal : 7

```

I. Experiment with converting an `int` value into other primitive types or vice versa and observe the results.

CODE:

```

public class P4 {

    public static void main(String[] args) {

        int number = 100;

        short shortNumber = (short) number;

        System.out.println("Int to Short: " + shortNumber);

        byte byteNumber = (byte) number;

        System.out.println("Int to Byte: " + byteNumber);

        long longNumber = (long) number;

        System.out.println("Int to long: " + longNumber);

        float floatNumber = (float) number;

        System.out.println("Int to float: " + floatNumber);
    }
}

```



```
double doubleNumber = (double) number;  
  
System.out.println("Int to double: " + doubleNumber);
```

```
int iNumber = 200;  
  
int iShort = (int) iNumber;  
  
System.out.println("Short to Int: " + iShort);  
  
byte sNumber = 20;  
  
int iByte = (int) sNumber;  
  
System.out.println("Byte to Int: " + iByte);  
  
long lNumber = 200;  
  
int iLong = (int) lNumber;  
  
System.out.println("Long to Int: " + iLong);  
  
float fNumber = 200.0f;  
  
int iFloat = (int) fNumber;  
  
System.out.println("Float to Int: " + iFloat);  
  
double dNumber = 200.0;  
  
int iDouble = (int) dNumber;  
  
System.out.println("Double to Int: " + iDouble);  
}  
}
```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P4.java
PS D:\dac\OOPJ\Day3\Practice> java P4
Int to Short: 100
Int to Byte: 100
Int to long: 100
Int to float: 100.0
Int to double: 100.0
Short to Int: 200
Byte to Int: 20
Long to Int: 200
Float to Int: 200
Double to Int: 200

```

5. Working with `java.lang.Long`

- Explore the [Java API documentation for `java.lang.Long`](#) and observe its modifiers and super types.
- Write a program to test how many bytes are used to represent a `long` value using the `BYTES` field. (Hint: Use `Long.BYTES`).

CODE:

```

public class P5 {

    public static void main(String[] args) {

        long num = Long.BYTES;

        System.out.println(num);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P5.java
PS D:\dac\OOPJ\Day3\Practice> java P5
8

```

- 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`).

CODE:

```

public class P5 {

```

```

public static void main(String[] args) {

    long min = Long.MIN_VALUE;

    long max = Long.MAX_VALUE;

    System.out.println("Minimum value is: "+min);

    System.out.println("Maximum vaue is: "+max);

}

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P5.java
PS D:\dac\OOPJ\Day3\Practice> java P5
Minimum value is: -9223372036854775808
Maximum vaue is: 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)`).

CODE:

```

public class P5 {

    public static void main(String[] args) {

        long number = 1000;

        String strString = Long.toString(number);

        System.out.println(strString);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P5.java
PS D:\dac\OOPJ\Day3\Practice> java P5
1000

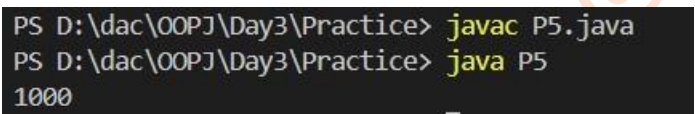
```

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)`).

CODE:

```
public class P5 {  
  
    public static void main(String[] args) {  
  
        String strString = "1000";  
  
        long number = Long.parseLong(strString);  
  
        System.out.println(number);  
  
    }  
}
```

OUTPUT:



```
PS D:\dac\OOPJ\Day3\Practice> javac P5.java  
PS D:\dac\OOPJ\Day3\Practice> java P5  
1000
```

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`).

CODE :

```
public class P5 {  
  
    public static void main(String[] args) {  
  
        String strNumber = "Ab12Cd3";  
  
        long number = Long.parseLong(strNumber);  
  
        System.out.println(number);  
  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P5.java
PS D:\dac\OOPJ\Day3\Practice> java P5
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:618)
    at java.base/java.lang.Long.parseLong(Long.java:722)
    at P5.main(P5.java:24)
```

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)`).

CODE:

```
public class P5 {

    public static void main(String[] args) {

        long number = 1000;

        long n1 = Long.valueOf(number);

        System.out.println(n1);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P5.java
PS D:\dac\OOPJ\Day3\Practice> java P5
1000
```

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)`).

CODE:

```
public class P5 {

    public static void main(String[] args) {

        String strString = "1000";

        long number = Long.valueOf(strString);

        System.out.println(number);

    }

}
```

```
}  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P5.java  
PS D:\dac\OOPJ\Day3\Practice> java P5  
1000
```

- 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)`).

CODE:

```
public class P5 {  
  
    public static void main(String[] args) {  
  
        long n1 = 1123;  
  
        long n2 = 9845;  
  
        long sum = Long.sum(n1, n2);  
  
        System.out.println(sum);  
  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P5.java  
PS D:\dac\OOPJ\Day3\Practice> java P5  
Sum is: 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)`).

CODE:

```
public class P5 {  
  
    public static void main(String[] args) {  
  
        long n1 = 1122;
```

```

    long n2 = 5566;

    long min = Long.min(n1, n2);

    long max = Long.max(n1, n2);

    System.out.println("Minimum value is: "+min);

    System.out.println("Maximum value is: "+max);

}

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P5.java
PS D:\dac\OOPJ\Day3\Practice> java P5
Minimum value is: 1122
Maximum value is: 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.toString(long)`, `Long.toOctalString(long)`, and `Long.toHexString(long)`).

CODE:

```

public class P5 {

    public static void main(String[] args) {

        long num = 7;

        String bin = Long.toString(num);

        String oct = Long.toOctalString(num);

        String hex = Long.toHexString(num);

        System.out.println("Long to Binary : " + bin);

        System.out.println("Long to Octal : " + oct);

        System.out.println("Long to Hexadecimal : " + hex);

    }

}

```

```
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P5.java
PS D:\dac\OOPJ\Day3\Practice> java P5
Long to Binary : 111
Long to Octal : 7
Long to Hexadecimal : 7
```

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

CODE: public class P5 {

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

```
        long number = 1000;
```

```
        short shortNumber = (short) number;
```

```
        System.out.println("Long to Short: " + shortNumber);
```

```
        byte byteNumber = (byte) number;
```

```
        System.out.println("Long to Byte: " + byteNumber);
```

```
        long longNumber = (long) number;
```

```
        System.out.println("Long to long: " + longNumber);
```

```
        float floatNumber = (float) number;
```

```
        System.out.println("Long to float: " + floatNumber);
```

```
        double doubleNumber = (double) number;
```

```
        System.out.println("Long to double: " + doubleNumber);
```

```
    }

    short sNumber = 2000;
```

```
    long iShort = (long) sNumber;
```

```
    System.out.println("Short to Long: " + iShort);
```

```
    byte bNumber = 20;
```



```

        long iByte = (long) bNumber;

        System.out.println("Byte to Long: " + iByte);

        int iNumber = 2000;

        long iInt = (long) iNumber;

        System.out.println("Int to Long: " + iInt);

        float fNumber = 2000.0f;

        long iFloat = (long) fNumber;

        System.out.println("Float to Long: " + iFloat);

        double dNumber = 2000.0;

        long iDouble = (long) dNumber;

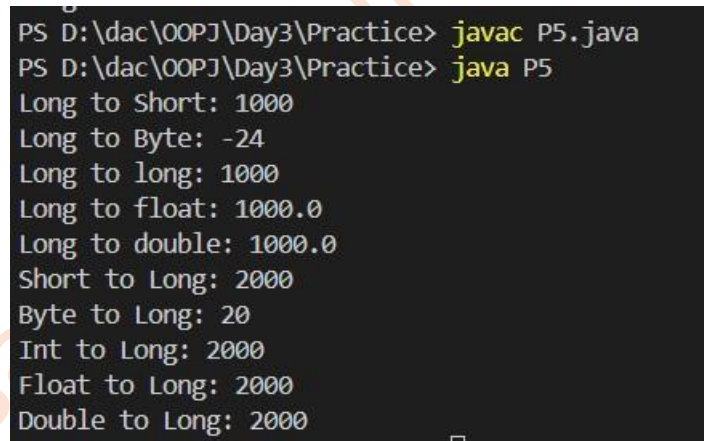
        System.out.println("Double to Long: " + iDouble);

    }

}

```

OUTPUT:



```

PS D:\dac\OOPJ\Day3\Practice> javac P5.java
PS D:\dac\OOPJ\Day3\Practice> java P5
Long to Short: 1000
Long to Byte: -24
Long to long: 1000
Long to float: 1000.0
Long to double: 1000.0
Short to Long: 2000
Byte to Long: 20
Int to Long: 2000
Float to Long: 2000
Double to Long: 2000

```

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`).

CODE:

```
public class P6 {  
  
    public static void main(String[] args) {  
  
        float num = Float.BYTES;  
  
        System.out.println(num);  
  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P6.java  
PS D:\dac\OOPJ\Day3\Practice> java P6  
4.0
```

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`).

CODE:

```
public class P6 {  
  
    public static void main(String[] args) {  
  
        float min = Float.MIN_VALUE;  
  
        float max = Float.MAX_VALUE;  
  
        System.out.println("Minimum value : "+ min);  
  
        System.out.println("Maximum value : "+max);  
  
    }  
}
```

OUTPUT:

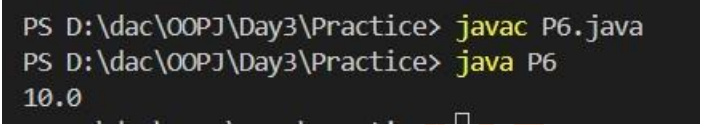
```
PS D:\dac\OOPJ\Day3\Practice> javac P6.java  
PS D:\dac\OOPJ\Day3\Practice> java P6  
Minimum value : 1.4E-45  
Maximum value : 3.4028235E38
```

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)`).

CODE:

```
public class P6 {  
  
    public static void main(String[] args) {  
  
        float number = 10.0f;  
  
        String strString = Float.toString(number);  
  
        System.out.println(strString);  
  
    }  
}
```

OUTPUT:



```
PS D:\dac\OOPJ\Day3\Practice> javac P6.java  
PS D:\dac\OOPJ\Day3\Practice> java P6  
10.0
```

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)`).

CODE:

```
public class P6 {  
  
    public static void main(String[] args) {  
  
        String strString = "10";  
  
        float number = Float.parseFloat(strString);  
  
        System.out.println(number);  
  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P6.java
PS D:\dac\OOPJ\Day3\Practice> java P6
10.0
```

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`).

CODE:

```
public class P6 {

    public static void main(String[] args) {

        String strNumber = "Ab12Cd3";

        float number = Float.parseFloat(strNumber);

        System.out.println(number);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P6.java
PS D:\dac\OOPJ\Day3\Practice> java P6
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:564)
    at P6.main(P6.java:25)
```

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)`).

CODE:

```
public class P6 {

    public static void main(String[] args) {

        float number = 10.0f;

        float n1 = Float.valueOf(number);

        System.out.println(n1);

    }

}
```

```
}  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P6.java  
PS D:\dac\OOPJ\Day3\Practice> java P6  
10.0
```

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)`).

CODE:

```
public class P6 {  
  
    public static void main(String[] args) {  
  
        String strString = "10";  
  
        float number = Float.valueOf(strString);  
  
        System.out.println(number);  
  
    }  
  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P6.java  
PS D:\dac\OOPJ\Day3\Practice> java P6  
10.0
```

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)`).

CODE:

```
public class P6 {  
  
    public static void main(String[] args) {  
  
        float n1 = 112.3f;  
  
        float n2 = 984.5f;
```

```

        float sum = Float.sum(n1, n2);

        System.out.println("Sum is: "+sum);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P6.java
PS D:\dac\OOPJ\Day3\Practice> java P6
Sum is: 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)`).

CODE:

```

public class P6 {

    public static void main(String[] args) {

        float n1 = 112.2f;

        float n2 = 556.6f;

        float min = Float.min(n1, n2);

        float max = Float.max(n1, n2);

        System.out.println("Minimum value is: "+min);

        System.out.println("Maximum value is: "+max);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P6.java
PS D:\dac\OOPJ\Day3\Practice> java P6
Minimum value is: 112.2
Maximum value is: 556.6

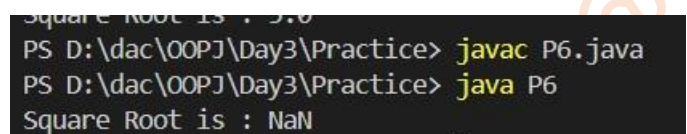
```

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

CODE:

```
public class P6 {  
  
    public static void main(String[] args) {  
  
        float num = -25.0f;  
  
        double sqrt = Math.sqrt(num);  
  
        System.out.println("Square Root is : "+sqrt);  
  
    }  
}
```

OUTPUT:



```
PS D:\dac\OOPJ\Day3\Practice> javac P6.java  
PS D:\dac\OOPJ\Day3\Practice> java P6  
Square Root is : NaN
```

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

CODE:

```
public class P6 {  
  
    public static void main(String[] args) {  
  
        float n1 = 0.0f;  
  
        float n2 = 0.0f;  
  
        float div = n1 / n2 ;  
  
        System.out.println(div);  
  
    }  
}
```

OUTPUT:

```
Square root is NaN
PS D:\dac\OOPJ\Day3\Practice> javac P6.java
PS D:\dac\OOPJ\Day3\Practice> java P6
NaN
```

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

CODE:

```
public class P6 {

    public static void main(String[] args) {

        float number = 100.0f;

        short shortNumber = (short) number;

        System.out.println("Float to Short: " + shortNumber);

        byte byteNumber = (byte) number;

        System.out.println("Float to Byte: " + byteNumber);

        long longNumber = (long) number;

        System.out.println("Float to long: " + longNumber);

        double doubleNumber = (double) number;

        System.out.println("Float to double: " + doubleNumber);

        short sNumber = 2000;

        float fShort = (float) sNumber;

        System.out.println("Float to Long: " + fShort);

        byte bNumber = 20;

        float fByte = (float) bNumber;

        System.out.println("Byte to Float: " + fByte);

        int iNumber = 2000;

        float fInt = (float) iNumber;

        System.out.println("Int to Float: " + fInt);
```



```

    long lNumber = 2000;

    float fLong = (float) lNumber;

    System.out.println("Long to Float " + fLong);

    double dNumber = 2000.0;

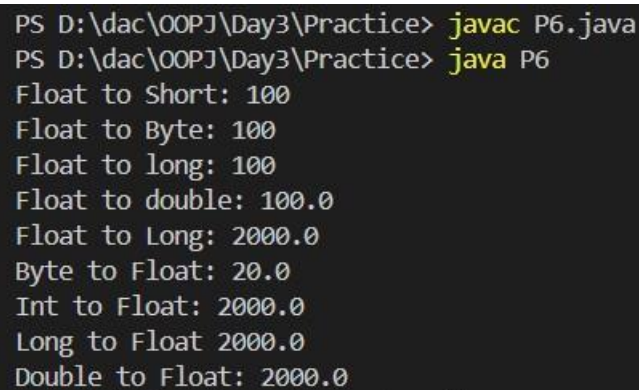
    float fDouble = (float) dNumber;

    System.out.println("Double to Float: " + fDouble);

}
}

```

OUTPUT:



```

PS D:\dac\OOPJ\Day3\Practice> javac P6.java
PS D:\dac\OOPJ\Day3\Practice> java P6
Float to Short: 100
Float to Byte: 100
Float to long: 100
Float to double: 100.0
Float to Long: 2000.0
Byte to Float: 20.0
Int to Float: 2000.0
Long to Float 2000.0
Double to Float: 2000.0

```

7. Working with `java.lang.Double`

- Explore the [Java API documentation for `java.lang.Double`](#) and observe its modifiers and super types.
- Write a program to test how many bytes are used to represent a `double` value using the `BYTES` field. (Hint: Use `Double.BYTES`).

CODE:

```

public class P7 {

    public static void main(String[] args) {

    }

}

```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P7.java
PS D:\dac\OOPJ\Day3\Practice> java P7
8.0
```

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`).

CODE:

```
public class P7 {

    public static void main(String[] args) {

        double min = Double.MIN_VALUE;

        double max = Double.MAX_VALUE;

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

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

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P7.java
PS D:\dac\OOPJ\Day3\Practice> java P7
Minimum value : 4.9E-324
Maximum value : 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)`).

CODE:

```
public class P7 {

    public static void main(String[] args) {

        double number = 10;
```

```

        String strString = Double.toString(number);

        System.out.println(strString);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P7.java
PS D:\dac\OOPJ\Day3\Practice> java P7
10.0

```

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)`).

CODE:

```

public class P7 {

    public static void main(String[] args) {

        String strString = "10";

        double number = Double.parseDouble(strString);

        System.out.println(number);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P7.java
PS D:\dac\OOPJ\Day3\Practice> java P7
10.0

```

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`).

CODE:

```

public class P7 {

```

```

public static void main(String[] args) {

    String strNumber = "Ab12Cd3";

    double number = Double.parseDouble(strNumber);

    System.out.println(number);

}

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P7.java
PS D:\dac\OOPJ\Day3\Practice> java P7
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:938)
    at P7.main(P7.java:28)

```

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)`).

CODE:

```

public class P7 {

    public static void main(String[] args) {

        double number = 10;

        double n1 = Double.valueOf(number);

        System.out.println(n1);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P7.java
PS D:\dac\OOPJ\Day3\Practice> java P7
10.0

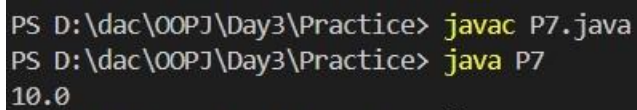
```

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)`).

CODE:

```
public class P7 {  
  
    public static void main(String[] args) {  
  
        double number = 10;  
  
        double n1 = Double.valueOf(number);  
  
        System.out.println(n1);  
  
    }  
}
```

OUTPUT:



```
PS D:\dac\OOPJ\Day3\Practice> javac P7.java  
PS D:\dac\OOPJ\Day3\Practice> java P7  
10.0
```

- 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)`).

CODE:

```
public class P7 {  
  
    public static void main(String[] args) {  
  
        double n1 = 112.3;  
  
        double n2 = 984.5;  
  
        double sum = Double.sum(n1, n2);  
  
        System.out.println("Sum is: "+sum);  
  
    }  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P7.java
PS D:\dac\OOPJ\Day3\Practice> java P7
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)`).

CODE:

```
public class P7 {

    public static void main(String[] args) {

        double n1 = 112.2f;

        double n2 = 556.6f;

        double min = Double.min(n1, n2);

        double max = Double.max(n1, n2);

        System.out.println("Minimum value is: "+min);

        System.out.println("Maximum value is: "+max);

    }

}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P7.java
PS D:\dac\OOPJ\Day3\Practice> java P7
Minimum value is: 112.19999694824219
Maximum value is: 556.5999755859375
```

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

CODE:

```
public class P7 {

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

```

double num = -25.0;

double sqrt = Math.sqrt(num);

System.out.println("Square Root is : "+sqrt);

}

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P7.java
PS D:\dac\OOPJ\Day3\Practice> java P7
Square Root is : NaN

```

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

CODE:

```

public class P7 {

    public static void main(String[] args) {

        double n1 = 0.0;

        double n2 = 0.0;

        double div = n1 / n2 ;

        System.out.println(div);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P7.java
PS D:\dac\OOPJ\Day3\Practice> java P7
NaN

```

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

CODE:

```
public class P7 {  
  
    public static void main(String[] args) {  
  
        double number = 100;  
  
        short shortNumber = (short) number;  
  
        System.out.println("Double to Short: " + shortNumber);  
  
        byte byteNumber = (byte) number;  
  
        System.out.println("Double to Byte: " + byteNumber);  
  
        long longNumber = (long) number;  
  
        System.out.println("Double to long: " + longNumber);  
  
        float floatNumber = (float) number;  
  
        System.out.println("Double to Float: " + floatNumber);  
  
        short sNumber = 2000;  
  
        double dShort = (double) sNumber;  
  
        System.out.println("Short to Double: " + dShort);  
  
        byte bNumber = 20;  
  
        double dByte = (double) bNumber;  
  
        System.out.println("Byte to Double: " + dByte);  
  
        int iNumber = 2000;  
  
        double dInt = (double) iNumber;  
  
        System.out.println("Int to Double: " + dInt);  
  
        long lNumber = 2000;  
  
        double dLong = (double) lNumber;  
  
        System.out.println("Long to Double: " + dLong);  
  
        float fNumber = 2000.0f;
```



```

        double dFloat = (double) fNumber;

        System.out.println("Float to Double: " + dFloat);

    }

}

```

OUTPUT:

```

PS D:\dac\OOPJ\Day3\Practice> javac P7.java
PS D:\dac\OOPJ\Day3\Practice> java P7
Double to Short: 100
Double to Byte: 100
Double to long: 100
Double to Float: 100.0
Short to Double: 2000.0
Byte to Double: 20.0
Int to Double: 2000.0
Long to Double: 2000.0
Float to Double: 2000.0

```

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()`).

CODE:

```

import java.util.Scanner;

public class P8 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter an integer value:");

        int intValue = sc.nextInt();

        System.out.println("Enter a long value:");

        long longValue = sc.nextLong();
    }
}

```

```
System.out.println("Enter a float value:");

float floatValue = sc.nextFloat();

System.out.println("Enter a double value:");

double doubleValue = sc.nextDouble();

System.out.println("Enter a byte value:");

byte byteValue = sc.nextByte();

System.out.println("Enter a short value:");

short shortValue = sc.nextShort();

System.out.println("Enter a boolean value (true/false):");

boolean booleanValue = sc.nextBoolean();

System.out.println("Enter a character value:");

char charValue = sc.next().charAt(0);

System.out.println("\nConverting using toString() method:");

System.out.println("Integer to String: " + Integer.toString(intValue));

System.out.println("Long to String: " + Long.toString(longValue));

System.out.println("Float to String: " + Float.toString(floatValue));

System.out.println("Double to String: " + Double.toString(doubleValue));

System.out.println("Byte to String: " + Byte.toString(byteValue));

System.out.println("Short to String: " + Short.toString(shortValue));

System.out.println("Boolean to String: " + Boolean.toString(booleanValue));

System.out.println("Char to String: " + Character.toString(charValue));


System.out.println("\nConverting using String.valueOf() method:");

System.out.println("Integer to String: " + String.valueOf(intValue));
```

```
System.out.println("Long to String: " + String.valueOf(longValue));  
System.out.println("Float to String: " + String.valueOf(floatValue));  
System.out.println("Double to String: " + String.valueOf(doubleValue));  
System.out.println("Byte to String: " + String.valueOf(byteValue));  
System.out.println("Short to String: " + String.valueOf(shortValue));  
System.out.println("Boolean to String: " + String.valueOf(booleanValue));  
System.out.println("Char to String: " + String.valueOf(charValue));  
  
sc.close();  
  
}  
  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P8.java
PS D:\dac\OOPJ\Day3\Practice> java P8
Enter an integer value:
35
Enter a long value:
4567
Enter a float value:
55.76
Enter a double value:
76.3
Enter a byte value:
24
Enter a short value:
13
Enter a boolean value (true/false):
true
Enter a character value:
j
java

Converting using toString() method:
Integer to String: 35
Long to String: 4567
Float to String: 55.76
Double to String: 76.3
Byte to String: 24
Short to String: 13
Boolean to String: true
Char to String: j

Converting using String.valueOf() method:
Integer to String: 35
Long to String: 4567
Float to String: 55.76
Double to String: 76.3
Byte to String: 24
Short to String: 13
Boolean to String: true
Char to String: j
```

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).

CODE:

```
public class P9 {
```

```
int i;
```

```
byte b;
```

```
char c;
```

```
boolean o;
```

```
long l;
```

```
short s;
```

```
float f;
```

```
double d;
```

```
public void displayDefault(){
```

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

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

```
    System.out.println("Character: "+c);
```

```
    System.out.println("Boolean: "+o);
```

```
    System.out.println("Long: "+l);
```

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

```
    System.out.println("Float: "+f);
```

```
    System.out.println("Double: "+d);
```

```
}
```

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

```
    P9 defaults = new P9();
```

```
    defaults.displayDefault();
```

```
}  
  
}
```

OUTPUT:

```
PS D:\dac\OOPJ\Day3\Practice> javac P9.java  
PS D:\dac\OOPJ\Day3\Practice> java P9  
Integer: 0  
Byte: 0  
Character:  
Boolean: false  
Long: 0  
Short: 0  
Float: 0.0  
Double: 0.0
```

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).

CODE:

```
import java.util.Scanner;  
public class P10 {  
    public static void main(String[] args) {  
  
        Scanner sc = new Scanner(System.in);  
  
        System.out.println("Enter first number: ");  
        int n1 = sc.nextInt();  
  
        System.out.println("Enter second number: ");  
        int n2 = sc.nextInt();  
  
        System.out.println("1.+ 2.- 3.* 4./");  
        System.out.println("Enter Your Choice");  
  
        int choice = sc.nextInt();  
  
        switch (choice) {  
            case 1:  
                int add=n1+n2;
```

```

        System.out.println("Addition is: "+add);
        break;
    case 2:
        int sub=n1-n2;
        System.out.println("Substraction is: "+sub);
        break;
    case 3:
        int mul=n1*n2;
        System.out.println("Multiplication is: "+mul);
        break;
    case 4:
        if(n2!=0){
            int div=n1/n2;
            System.out.println("Division is: "+div);
        }
        break;

    default:
        System.out.println("Enter Valid Choice");
    }
    sc.close();
}
}

```

OUTPUT:

```

PS D:\dac\00PJ\Day3\Practice> javac P10.java
PS D:\dac\00PJ\Day3\Practice> java P10
Enter first number:
12
Enter second number:
23
1.+ 2.- 3.* 4./
Enter Your Choice
1
Addition is: 35

```