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.

Explanation: Boolean is a class in java.lang(which is a root of all classes and also "cosmic superclass") which is most important classes and pre-imported in the java. The boolean datatype is not a class in it rather defined in wrapper class(java.lang.Boolean) and has a final keyword which signifies that it cannot have a child class.

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)`).

```
package in.Cdac.Boolean;

public class Q1_B {
	public static void main(String[] args)
	{
		boolean status = true;
		String str = Boolean.toString(status);
		System.out.println(str);
	}
}
```

```
<terminated> Q1_B [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe  (09-Sep-2024, 11:16:25 pm – 11:16:25 pm) [pid: 5552]
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)`).

```
package in.Cdac.Boolean;

public class Q1_C {
	public static void main(String[] args) {
```

```
		String strStatus = "true";
		boolean bool = Boolean.parseBoolean(strStatus);
		System.out.println(bool);
	}
}
```

```
<terminated> Q1_C [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe  (09-Sep-2024, 11:16:58 pm – 11:16:59 pm) [pid: 13140]
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"`).**

```
package in.Cdac.Boolean;

public class Q1_D {
	public static void main(String[] args) {
		String strStatus =" 0 ";
		boolean bool = Boolean.parseBoolean(strStatus);
		System.out.println(bool);
	}
}
```

```
<terminated> Q1_D [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe
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)`).**

```
package in.Cdac.Boolean;

public class Q1_E {
	public static void main(String[] args) {
		boolean status = true;
		boolean bool = Boolean.valueOf(status);
		System.out.println(bool);
	}
}
```

```
<terminated> Q1_E [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe (
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)`).

```
package in.Cdac.Boolean;

public class Q1_F {
	public static void main(String[] args) {
		String strStatus ="true";
		boolean bool = Boolean.valueOf(strStatus);
		System.out.println(bool);
	}
}
```

```
<terminated> Q1_F [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe
true
```

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

Explanation: Boolean.valueOf() is same as Boolean.toString() which is used to convert String a non-primitive datatype into a primitive datatype this process is called as unboxing. This would have not worked if the string was anything other than 'true' or 'false' and would have just returned as false.

2. Working with `java.lang.Byte`

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

Explanation: Byte is a class in java.lang(which is a root of all classes and also “cosmic superclass”) which is most important classes and pre-imported in the java. The byte datatype is not a class in it rather defined in wrapper class(java.lang.Byte) and has a final keyword which signifies that it cannot have a child class.

byte datatype----→java.lang.Byte (wrapper class)----→java.lang(superclass)

byte ranges from -------→ -128 to 127

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

```
package in.Cdac.Byte;

public class Q2_B {
	public static void main(String[] args)
```

```
	{
		byte b = Byte.BYTES;
		System.out.println(b);
	}
}
```

```
<terminated> Q2_B [Java Application] C:\Program Files\Java\jdk-18.0.2
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`).**

```
package in.Cdac.Byte;

public class Q2_C {
	public static void main(String[] args)
	{
		byte b = Byte.BYTES;
		System.out.println(Byte.MIN_VALUE);
		System.out.println(Byte.MAX_VALUE);
	}
}
```

```
<terminated> Q2_C [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.e
-128
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)`).**

```
package in.Cdac.Byte;

public class Q2_D {
	public static void byte1() {
		byte num = 45;
		String str = Byte.toString(num);
		System.out.println(str);
	}
	public static void main(String[] args) {
		byte b = Byte.BYTES;
		System.out.println(Byte.MIN_VALUE);
		System.out.println(Byte.MAX_VALUE);
		byte1();
	}
}
```

```
<terminated> Q2_D [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.e
-128
127
45
```

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)`).**

```
package in.Cdac.Byte;

public class Q2_E {
	public static void main(String[] args) {
		String strNumber = "5";
		byte b = Byte.parseByte(strNumber);
		System.out.println(b);
	}
}
```

```
<terminated> Q2_E [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\java
5
```

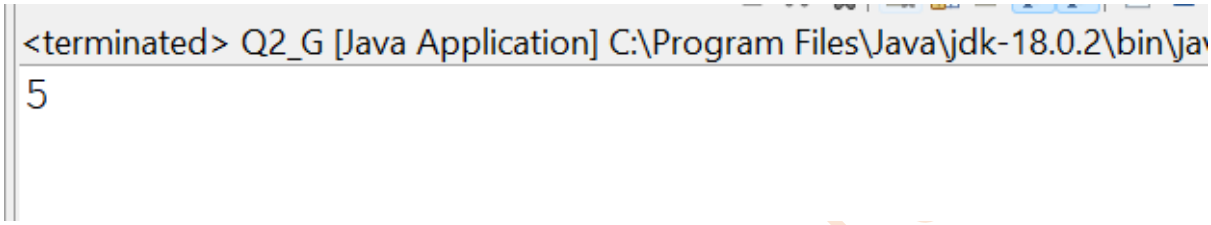
f. **Declare a method-local variable `strNumber` of type `String` with the value `"Ab12Cd3"` and attempt to convert it to a `byte` value. (Hint: `parseByte` method will throw a `NumberFormatException`).**

```
package in.Cdac.Byte;

public class Q2_F {
	public static void main(String[] args) {
		String strNumber = "Ab12Cd3";
		byte b = Byte.parseByte(strNumber);
		System.out.println(b);
	}
}
```

```
<terminated> Q2_F [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe  (09-Sep-2024, 11:20:47 pm – 11:20:47 pm) [pid: 8632]
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 in.Cdac.Byte.Q2_F.main(Q2_F.java:6)
```

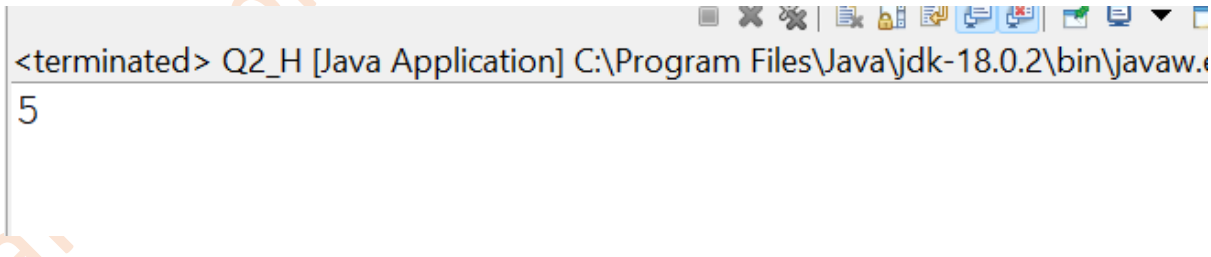
g. **Declare a method-local variable `number` of type `byte` with some value and convert it to the corresponding wrapper class using `Byte.valueOf()`. (Hint: Use `Byte.valueOf(byte)`).**

```
package in.Cdac.Byte;

public class Q2_G {
	public static void main(String[] args) {
		byte b1 = 5;
		byte b = Byte.valueOf(b1);
		System.out.println(b);
	}
}
```



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)`).**

```
package in.Cdac.Byte;

public class Q2_H {
	public static void main(String[] args) {
		String strNumber = "5";
		byte b = Byte.parseByte(strNumber);
		System.out.println(b);
	}
}
```



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

3. Working with `java.lang.Short`

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

Explanation: Short is a class in java.lang(which is a root of all classes and also “cosmic superclass”) which is most important classes and pre-imported in the java. The short

datatype is not a class in it rather defined in wrapper class(java.lang.Short) and has a final keyword which signifies that it cannot have a child class.

short datatype----→java.lang.Short (wrapper class)----→java.lang(superclass)

short ranges from----------→-32768 to 32767

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

```
package in.Cdac.Short;

public class Q3_B {
	public static void main(String[] args)
	{
		System.out.println(Short.BYTES);
	}
}
```

```
<terminated> Q3_B [Java Application] C:\Program Files\Java\jdk-18.0.2
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`).**

```
package in.Cdac.Short;

public class Q3_C {
	public static void main(String[] args)
	{
		System.out.println(Short.BYTES);
		System.out.println(Short.MIN_VALUE);
		System.out.println(Short.MAX_VALUE);
	}

}
```

```
<terminated> Q3_C [Java Application] C:\Program Files\Java\jdk-18.0.
2
-32768
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)`).**

```
package in.Cdac.Short;

public class Q3_D {
	public static void main(String[] args) {
		short num = 3352;
		String n1 = Short.toString(num);
		System.out.println(n1);
	}
}
```

```
<terminated> Q3_D [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\java
3352
```

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)`).**

```
package in.Cdac.Short;

public class Q3_E {
	public static void main(String[] args) {
		String strNumber = "4586";
		short sh = Short.parseShort(strNumber);
		System.out.println(sh);
	}
}
```

```
<terminated> Q3_E [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.e
4586
```

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`).**

```
package in.Cdac.Short;

public class Q3_F {
	public static void main(String[] args) {
		String strNumber = "Ab12Cd3";
		short sh = Short.parseShort(strNumber);
		System.out.println(sh);
	}
}
```

```
<terminated> Q3_F [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe  (09-Sep-2024, 11:23:22 pm – 11:23:22 pm) [pid: 22820]
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 in.Cdac.Short.Q3_F.main(Q3_F.java:6)
```

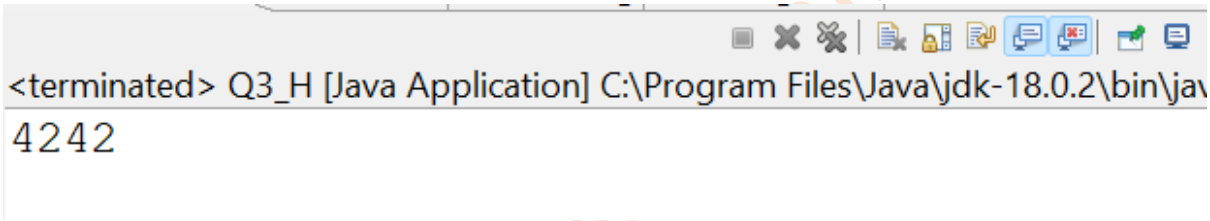
g. **Declare a method-local variable `number` of type `short` with some value and convert it to the corresponding wrapper class using `Short.valueOf()`. (Hint: Use `Short.valueOf(short)`).**

```
package in.Cdac.Short;

public class Q3_G {
	public static void short1() {
		short number = 4242;
		short sh = Short.valueOf(number);
		System.out.println(sh);
	}
	public static void main(String[] args) {
		short1();
	}
}
```

```
<terminated> Q3_G [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\
4242
```

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)`).**

```
package in.Cdac.Short;

public class Q3_H {
	public static void short1() {
		String strNumber = "4242";
		short sh = Short.valueOf(strNumber);
		System.out.println(sh);
	}
	public static void main(String[] args) {
		short1();
	}
}
```



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

4. Working with `java.lang.Integer`

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

Explanation: Integer is a class in java.lang(which is a root of all classes and also "cosmic superclass") which is most important classes and pre-imported in the java. The int datatype is not a class in it rather defined in wrapper class(java.lang.Short) and has a final keyword which signifies that it cannot have a child class.

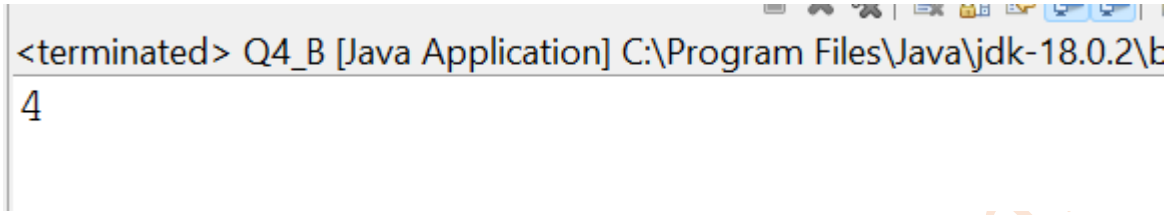
int datatype----→java.lang.Integer (wrapper class)----→java.lang(superclass)

int ranges from----------→-2147483648 to 2147483647

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

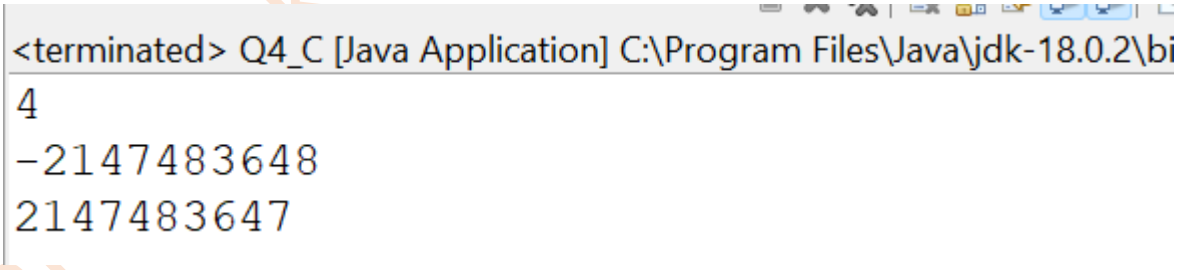
```
package in.Cdac.Integer;

public class Q4_B {
```

```
	public static void main(String[] args) {
		int b = Integer.BYTES;
		System.out.println(b);
	}
}
```



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`).**

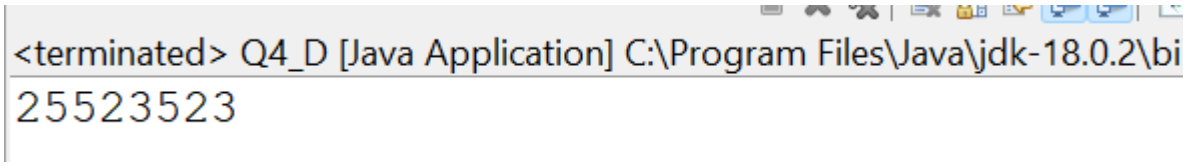
```
package in.Cdac.Integer;

public class Q4_C {
	public static void main(String[] args) {
		int b = Integer.BYTES;
		System.out.println(b);
		System.out.println(Integer.MIN_VALUE);
		System.out.println(Integer.MAX_VALUE);
	}
}
```



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)`).**

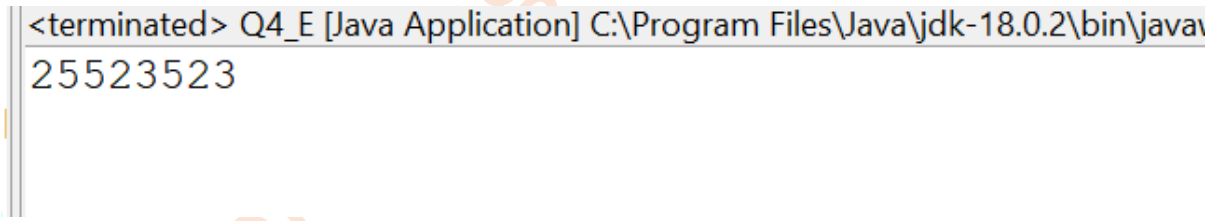
```
package in.Cdac.Integer;

public class Q4_D {
	public static void main(String[] args) {
		int num = 25523523;
```

```
		String str = Integer.toString(num);
		System.out.println(str);
	}
}
```



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)`).**

```
package in.Cdac.Integer;

public class Q4_E {
	public static void main(String[] args) {
		String strNumber = "25523523";
		int b = Integer.parseInt(strNumber);
		System.out.println(b);
	}
}
```



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`).**

```
package in.Cdac.Integer;

public class Q4_F {
	public static void main(String[] args) {
		String num = "Ab12Cd3";
		int str = Integer.parseInt(num);
		System.out.println(str);
	}
}
```

```
<terminated> Q4_F [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe  (09-Sep-2024, 11:25:57 pm – 11:25:57 pm) [pid: 12356]
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.Integer.parseInt(Integer.java:784)
        at in.Cdac.Integer.Q4_F.main(Q4_F.java:6)
```

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)`).**

```
package in.Cdac.Integer;

public class Q4_G {
	public static void main(String[] args) {
		int num = 25523523;
		int b = Integer.valueOf(num);
		System.out.println(b);
	}
}
```

```
<terminated> Q4_G [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw
25523523
```

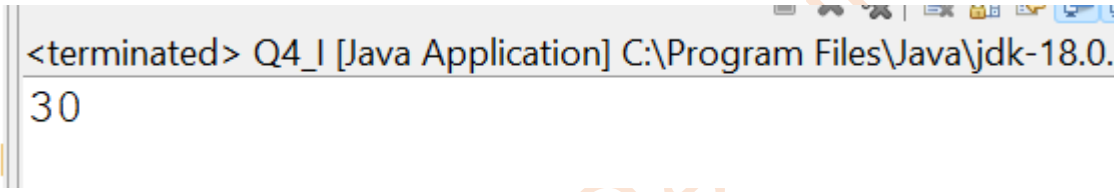
h. **Declare a method-local variable `strNumber` of type `String` with some integer value and convert it to the corresponding wrapper class using `Integer.valueOf()`. (Hint: Use `Integer.valueOf(String)`).**

```
package in.Cdac.Integer;

public class Q4_H {
	public static void main(String[] args) {
		String strNumber = "25523523";
		int str = Integer.parseInt(strNumber);
		System.out.println(str);
	}
}
```

```
<terminated> Q4_H [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe
25523523
```

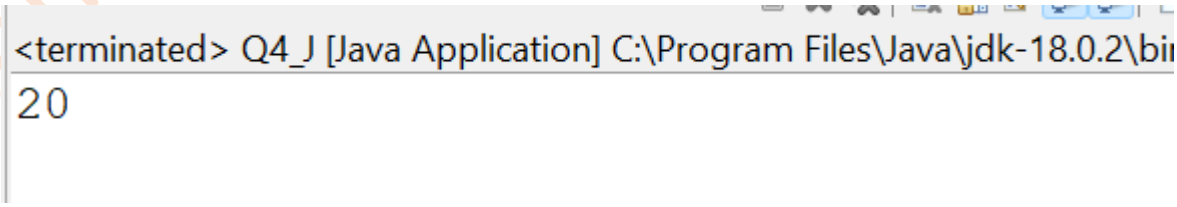
i. **Declare two integer variables with values `10` and `20`, and add them using a method from the `Integer` class. (Hint: Use `Integer.sum(int, int)`).**

```
package in.Cdac.Integer;

public class Q4_I {
	public static void main(String[] args) {
		int a=10;
		int b=20;
		System.out.println(Integer.sum(a,b));
	}
}
```



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)`).**

```
package in.Cdac.Integer;

public class Q4_J {
	public static void main(String[] args) {
		int a=10;
		int b=20;
		System.out.println(Integer.max(a,b));
	}
}
```



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)`).**

```
package in.Cdac.Integer;

public class Q4_K {
	public static void main(String[] args) {
		int a=7;
		System.out.println(Integer.toBinaryString(a));
		System.out.println(Integer.toOctalString(a));
		System.out.println(Integer.toHexString(a));
	}
}
```

```
<terminated> Q4_K [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\j
111
7
7
```

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

5. Working with `java.lang.Long`

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

Explanation: Long is a class in java.lang(which is a root of all classes and also "cosmic superclass") which is most important classes and pre-imported in the java. The long datatype is not a class in it rather defined in wrapper class(java.lang.Short) and has a final keyword which signifies that it cannot have a child class.

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

```
package in.Cdac.Long;

public class Q5_B {
	public static void main(String[] args) {
		long b = Long.BYTES;
		System.out.println(b);
	}
}
```

```
<terminated> Q5_B [Java Application] C:\Program Files\Java\jdk-18.0.2\b
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`).**

```
package in.Cdac.Long;

public class Q5_C {
	public static void main(String[] args) {
		System.out.println(Long.MIN_VALUE);
		System.out.println(Long.MIN_VALUE);
	}
}
```

```
<terminated> Q5_C [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw
-9223372036854775808
-9223372036854775808
```

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)`).**

```
package in.Cdac.Long;
public class Q5_D {
	public static void main(String[] args) {
		long num = 12412;
		String str = Long.toString(num);
		System.out.println(str);
	}
}
```

```
<terminated> Q5_D [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.e
12412
```

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)`).**

```
package in.Cdac.Long;

public class Q5_E {
	public static void main(String[] args) {
		String strNumber = "23532";
		long b = Long.parseLong(strNumber);
		System.out.println(b);
	}
}
```

```
<terminated> Q5_E [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw
23532
```

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`).**

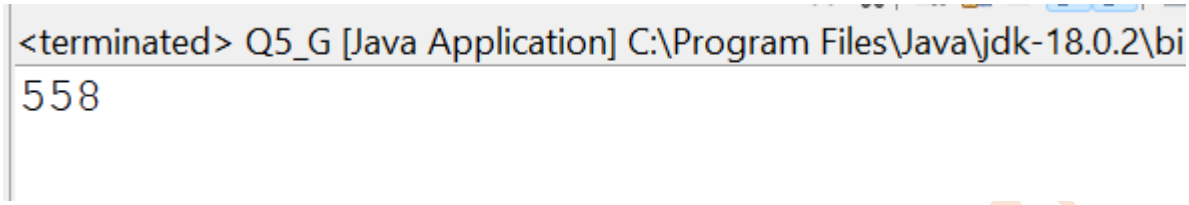
```
package in.Cdac.Long;

public class Q5_F {
	public static void main(String[] args) {
		String strNumber = "Ab12Cd3";
		long b = Long.parseLong(strNumber);
		System.out.println(b);
	}
}
```

```
<terminated> Q5_F [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe  (09-Sep-2024, 11:29:43 pm – 11:29:43 pm) [pid: 2312]
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:834)
	at in.Cdac.Long.Q5_F.main(Q5_F.java:6)
```

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)`).**

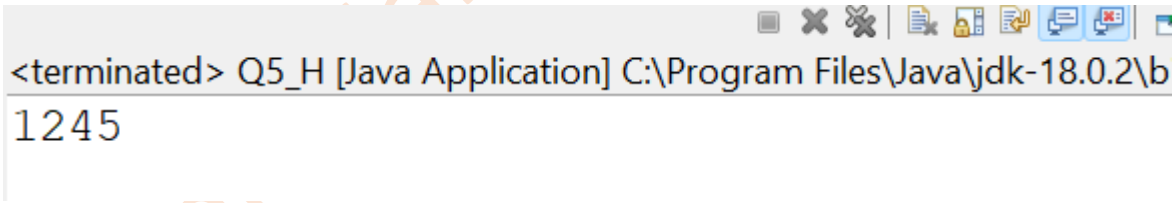
```
package in.Cdac.Long;

public class Q5_G {
	public static void main(String[] args) {
```

```
		long b = 558;
		long by = Long.valueOf(b);
		System.out.println(by);
	}
}
```



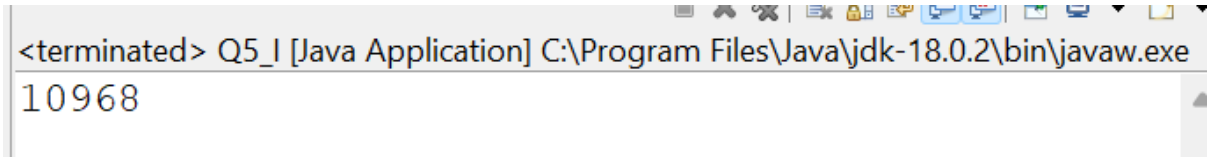
h. **Declare a method-local variable `strNumber` of type `String` with some `long` value and convert it to the corresponding wrapper class using `Long.valueOf()`. (Hint: Use `Long.valueOf(String)`).**

```
package in.Cdac.Long;

public class Q5_H {
	public static void main(String[] args) {
		String strNumber = "1245";
		long b = Long.valueOf(strNumber);
		System.out.println(b);
	}
}
```



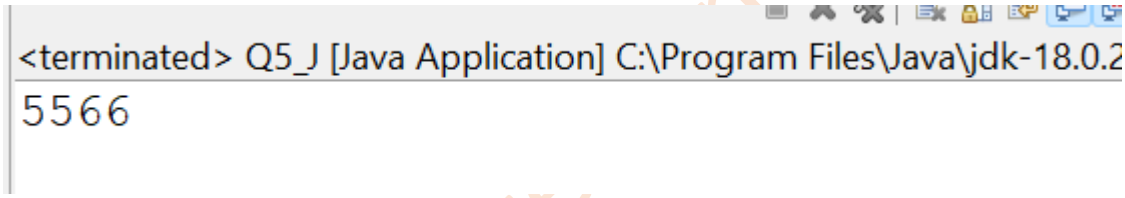
i. **Declare two long variables with values `1123` and `9845`, and add them using a method from the `Long` class. (Hint: Use `Long.sum(long, long)`).**

```
package in.Cdac.Long;

public class Q5_I {
	public static void main(String[] args) {
		long a = 1123;
		long b = 9845;
		System.out.println(Long.sum(a, b));
	}
}
```



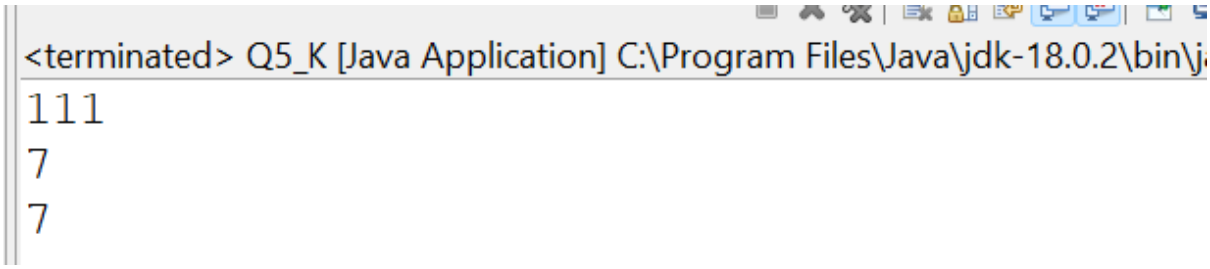
j. **Declare two long variables with values `1122` and `5566`, and find the minimum and maximum values using the `Long` class. (Hint: Use `Long.min(long, long)` and `Long.max(long, long)`).**

```
package in.Cdac.Long;

public class Q5_J {
	public static void main(String[] args) {
		int a=1122;
		int b=5566;
		System.out.println(Long.max(a,b));
	}
}
```



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)`).**

```
package in.Cdac.Long;

public class Q5_K {
	public static void main(String[] args) {
		long a=7;
		System.out.println(Long.toBinaryString(a));
		System.out.println(Long.toOctalString(a));
		System.out.println(Long.toHexString(a));
	}
}
```



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

6. Working with `java.lang.Float`

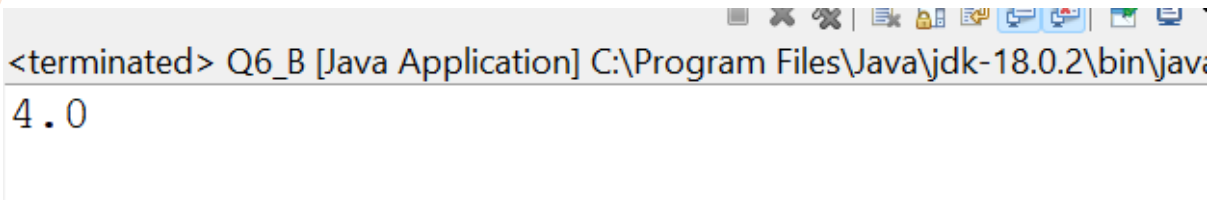
a. **Explore the Java API documentation for `java.lang.Float` and observe its modifiers and super types.**

Explanation: Float is a class in java.lang(which is a root of all classes and also "cosmic superclass") which is most important classes and pre-imported in the java. The float datatype is not a class in it rather defined in wrapper class(java.lang.Float) and has a final keyword which signifies that it cannot have a child class.

float datatype----→java.lang.Float (wrapper class)----→java.lang(superclass)

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

```
package in.Cdac.Float;

public class Q6_B {
	public static void main(String[] args) {
		float b = Float.BYTES;
		System.out.println(b);
	}
}
```



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

```
package in.Cdac.Float;

public class Q6_C {
	public static void main(String[] args) {
		float b = Float.BYTES;
		System.out.println(b);
		System.out.println(Float.MIN_VALUE);
		System.out.println(Float.MAX_VALUE);
	}
}
```

```
<terminated> Q6_C [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw
4.0
1.4E-45
3.4028235E38
```

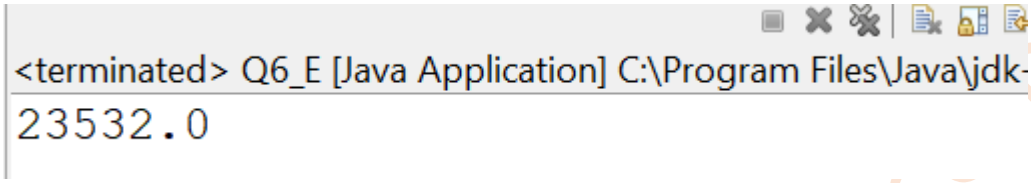
d. **Declare a method-local variable `number` of type `float` with some value and convert it to a `String` using the `toString` method. (Hint: Use `Float.toString(float)`).**

```
package in.Cdac.Float;
public class Q6_D {
	public static void main(String[] args) {
		float num = 12412;
		String str = Float.toString(num);
		System.out.println(str);
	}
}
```

```
<terminated> Q6_D [Java Application] C:\Program Files\Java\jdk-1
12412.0
```

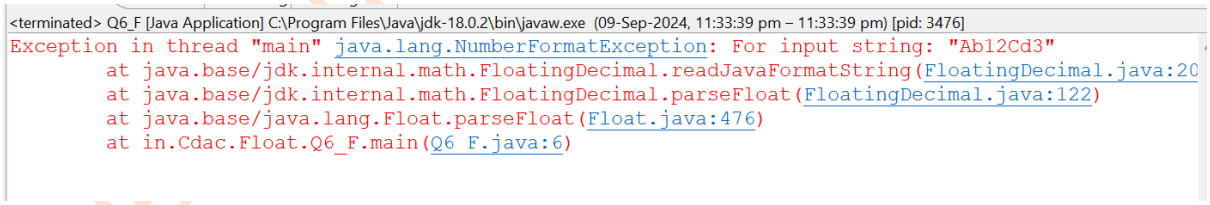
e. **Declare a method-local variable `strNumber` of type `String` with some value and convert it to a `float` value using the `parseFloat` method. (Hint: Use `Float.parseFloat(String)`).**

```
package in.Cdac.Float;

public class Q6_E {
	public static void main(String[] args) {
		String strNumber = "23532";
		float b = Float.parseFloat(strNumber);
		System.out.println(b);
	}
}
```



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

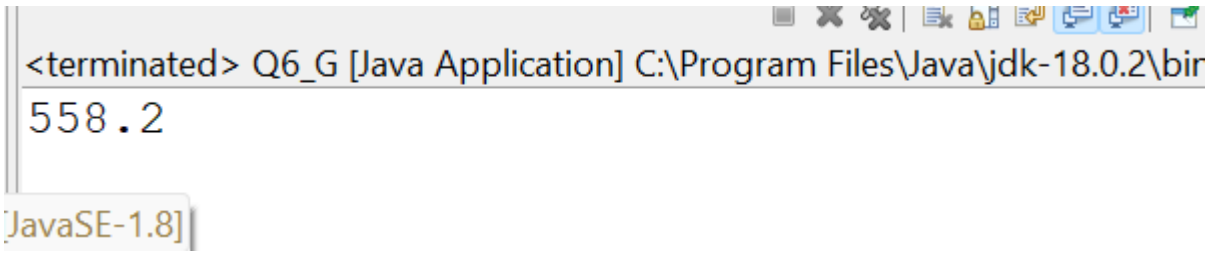
```
package in.Cdac.Float;

public class Q6_F {
	public static void main(String[] args) {
		String strNumber = "Ab12Cd3";
		float b = Float.parseFloat(strNumber);
		System.out.println(b);
	}
}
```



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)`).**

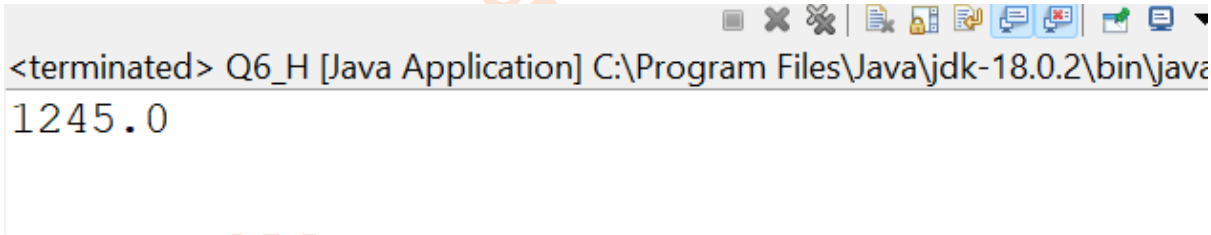
```
package in.Cdac.Float;

public class Q6_G {
	public static void main(String[] args) {
		float b = 558.2f;
		float by = Float.valueOf(b);
		System.out.println(by);
	}
```

```
}
```



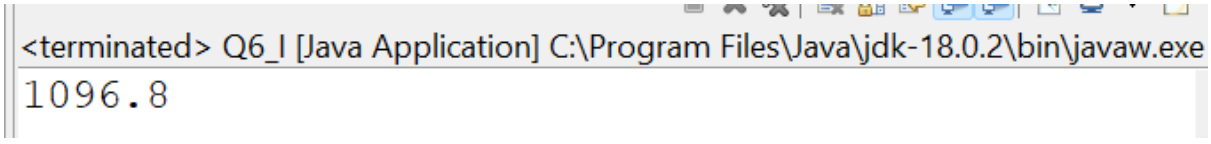
h. **Declare a method-local variable `strNumber` of type `String` with some `float` value and convert it to the corresponding wrapper class using `Float.valueOf()`. (Hint: Use `Float.valueOf(String)`).**

```
package in.Cdac.Float;

public class Q6_H {
	public static void main(String[] args) {
		String strNumber = "1245";
		float b = Float.valueOf(strNumber);
		System.out.println(b);
	}
}
```



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)`).**

```
package in.Cdac.Float;

public class Q6_I {
	public static void main(String[] args) {
		float a = 112.3f;
		float b = 984.5f;
		System.out.println(Float.sum(a, b));
	}
}
```



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)`).**

```
package in.Cdac.Float;

public class Q6_J {
	public static void main(String[] args) {
		float a=112.2f;
		float b=556.6f;
		System.out.println(Float.min(a,b));
		System.out.println(Float.max(a,b));
	}
}
```

```
<terminated> Q6_J [Java Application] C:\Program Files\Java\jdk-1
112.2
556.6
```

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

```
package in.Cdac.Float;

public class Q6_K {
	public static void main(String[] args) {
		float a= -25.0f;
		System.out.println(Math.sqrt(a));
	}
}
```

```
<terminated> Q6_K [Java Application] C:\Program Files\Java\jdk-18.0.2\b
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).**

```
package in.Cdac.Float;

public class Q6_L {
	public static void main(String[] args) {
```

```
		float a = 0.0f;
		float b =0.0f;
		System.out.println(a/b);
	}
}
```

```
<terminated> Q6_L [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\jav
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.**

Explanation: double is a class in java.lang(which is a root of all classes and also "cosmic superclass") which is most important classes and pre-imported in the java. The double datatype is not a class in it rather defined in wrapper class(java.lang.Double) and has a final keyword which signifies that it cannot have a child class.

double datatype----→java.lang.Double (wrapper class)----→java.lang(superclass)

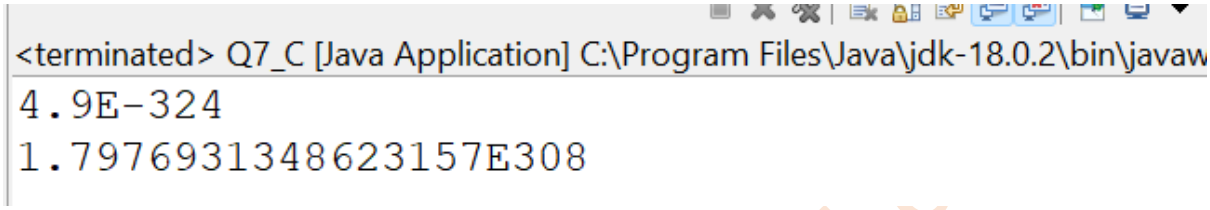
b. **Write a program to test how many bytes are used to represent a `double` value using the `BYTES` field. (Hint: Use `Double.BYTES`).**

```
package in.Cdac.Double;

public class Q7_B {
	public static void main(String[] args) {
		double b = Double.BYTES;
		System.out.println(b);
	}
}
```

```
<terminated> Q7_B [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\j
8.0
```

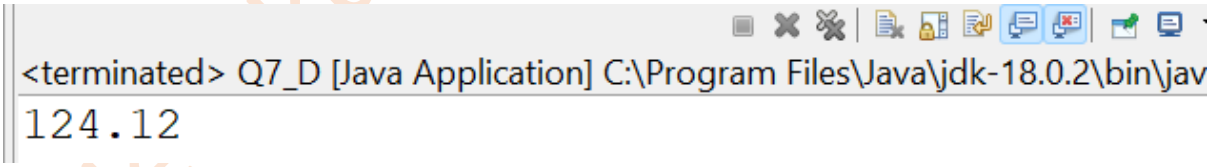
c. **Write a program to find the minimum and maximum values of `double` using the `MIN_VALUE` and `MAX_VALUE` fields. (Hint: Use `Double.MIN_VALUE` and `Double.MAX_VALUE`).**

```
package in.Cdac.Double;

public class Q7_C {
	public static void main(String[] args) {
		double b = Double.BYTES;
		System.out.println(Double.MIN_VALUE);
		System.out.println(Double.MAX_VALUE);
	}
}
```



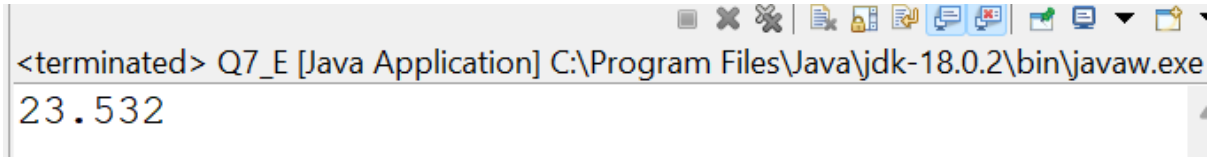
d. **Declare a method-local variable `number` of type `double` with some value and convert it to a `String` using the `toString` method. (Hint: Use `Double.toString(double)`).**

```
package in.Cdac.Double;
public class Q7_D {
	public static void main(String[] args) {
		double num = 124.12;
		String str = Double.toString(num);
		System.out.println(str);
	}
}
```



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)`).**

```
package in.Cdac.Double;

public class Q7_E {
	public static void main(String[] args) {
		String strNumber = "23.532";
		double b = Double.parseDouble(strNumber);
		System.out.println(b);
	}
}
```



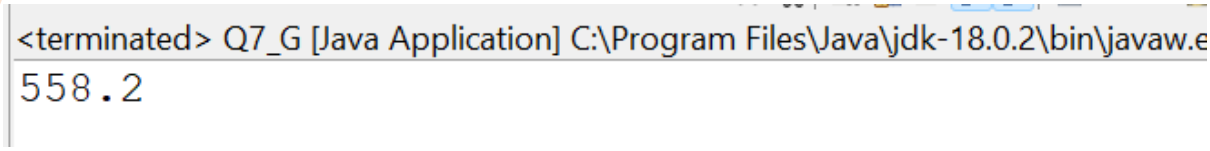
f. **Declare a method-local variable `strNumber` of type `String` with the value `"Ab12Cd3"` and attempt to convert it to a `double` value. (Hint: `parseDouble` method will throw a `NumberFormatException`).**

```
package in.Cdac.Double;

public class Q7_F {
	public static void main(String[] args) {
		String strNumber = "Ab12Cd3";
		double b = Double.parseDouble(strNumber);
		System.out.println(b);
	}
}
```

```
<terminated> Q7_F [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe (09-Sep-2024, 11:37:42 pm – 11:37:42 pm) [pid: 8748]
Exception in thread "main" java.lang.NumberFormatException: For input string: "Ab12Cd3"
	at java.base/jdk.internal.math.FloatingDecimal.readJavaFormatString(FloatingDecimal.java:20
	at java.base/jdk.internal.math.FloatingDecimal.parseDouble(FloatingDecimal.java:110)
	at java.base/java.lang.Double.parseDouble(Double.java:651)
	at in.Cdac.Double.Q7_F.main(Q7_F.java:6)
```

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)`).**

```
package in.Cdac.Double;

public class Q7_G {
	public static void main(String[] args) {
		double b = 558.2;
		double by = Double.valueOf(b);
		System.out.println(by);
	}
}
```



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)`).

```
package in.Cdac.Double;

public class Q7_H {
	public static void main(String[] args) {
		String strNumber = "12.45";
		double b = Double.valueOf(strNumber);
		System.out.println(b);
	}
}
```

```
<terminated> Q7_H [Java Application] C:\Program Files\Java\jdk-18.0.2\bir
12.45
```

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)`).**

```
package in.Cdac.Double;

public class Q7_I {
	public static void main(String[] args) {
		double a = 112.3;
		double b = 984.5;
		System.out.println(Double.sum(a, b));
	}
}
```

```
<terminated> Q7_I [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.ex
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)`).**

```
package in.Cdac.Double;

public class Q7_J {
	public static void main(String[] args) {
```

```
            double a=112.2;
            double b=556.6;
            System.out.println(Double.min(a,b));
            System.out.println(Double.max(a,b));
      }
}
```

```
<terminated> Q7_J [Java Application] C:\Program Files\Java\jdk-18.0.2\bir
112.2
556.6
```

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

```
package in.Cdac.Double;

public class Q7_K {
      public static void main(String[] args) {
            double a= -25.0;
            System.out.println(Math.sqrt(a));
      }
}
```

```
<terminated> Q7_K [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.ex
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).**

```
package in.Cdac.Double;

public class Q7_L {
      public static void main(String[] args) {
            double a = 0.0;
            double b =0.0;
            System.out.println(a/b);
      }
}
```

```
<terminated> Q7_L [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\java
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()`).**

```
class PTToStrings{
	public static void main(String args[])
	{
		byte d = 127;
		String e = Byte.toString(d);
		System.out.println(e);
		String f= String.valueOf(e);
		System.out.println(f);

		short a = 32767;
		String b = Short.toString(a);
		System.out.println(b);
		String c = String.valueOf(b);
		System.out.println(c);

		int g = 1421015127;
		String h = Integer.toString(g);
		System.out.println(h);
		String i = String.valueOf(h);
		System.out.println(i);

		long j = 14210151279882388L;
		String k = Long.toString(j);
		System.out.println(k);
		String l = String.valueOf(k);
		System.out.println(l);

		float m = 1421015127f;
		String n = Float.toString(m);
		System.out.println(n);
		String o = String.valueOf(n);
		System.out.println(o);

		double p = 14210151279828388d;
		String q = Double.toString(p);
		System.out.println(q);
		String r = String.valueOf(q);
		System.out.println(r);
```

```
	}
}
```

```
D:\CDAC\OOPJ\Day 2\Day 2>javac PTToStrings.java

D:\CDAC\OOPJ\Day 2\Day 2>java PTToStrings
127
127
32767
32767
1421015127
1421015127
14210151279882388
14210151279882388
1.42101517E9
1.42101517E9
1.4210151279828388E16
1.4210151279828388E16

D:\CDAC\OOPJ\Day 2\Day 2>
```

9. Default Values of Primitive Types

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

```
class Default{
		private boolean b;
		private char c;
		private byte by;
		private short s;
		private int i;
		private float f;
		private long l;
		private double d;

		public void types(){
			System.out.println("Default value of boolean :" + b);
			System.out.println("Default value of char :" + c);
			System.out.println("Default value of byte :" + by);
			System.out.println("Default value of short :" + s);
			System.out.println("Default value of int :" + i);
			System.out.println("Default value of float :" + f);
			System.out.println("Default value of long :" + l);
			System.out.println("Default value of double :" + d);
		}
	}
public class Values{
	public static void main(String args[]){
		Default d = new Default();
```

```
            d.types();
      }
}
```

```
D:\CDAC\OOPJ\Day 2\Day 2>javac Values.java

D:\CDAC\OOPJ\Day 2\Day 2>java Values
Default value of boolean :false
Default value of char :
Default value of byte :0
Default value of short :0
Default value of int :0
Default value of float :0.0
Default value of long :0
Default value of 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).

```
public class Calculator_New {
      public static void main(String args[]){
            int num1 = Integer.parseInt(args[0]);
            System.out.println(num1);
            int num2 = Integer.parseInt(args[2]);
            System.out.println(num2);
            String operator = args[1];
            switch(operator){
                  case "+":
                              System.out.println("Sum of " + num1 + " and
" + num2 + " is " + (num1+num2));
                              break;
                  case "-":
                              System.out.println("Difference of " + num1
+ " and " + num2 + " is " +(num1-num2));
                              break;
```

```
                    case "*":
                                System.out.println("Product of " + num1 + "
and " + num2 + " is " +(num1*num2));
                                break;
                    case "/":
                                System.out.println("Quotient of " + num1 +
" and " + num2 + " is " +(num1/num2));
                                break;
                    default:
                                System.out.println("Invalid input");
                                break;
                }
        }

}
```

```
C:\Users\Priyanka\Desktop>javac Calculator_New.java

C:\Users\Priyanka\Desktop>java Calculator_New 12 + 15
12
15
Sum of 12 and 15 is 27

C:\Users\Priyanka\Desktop>java Calculator_New 12 - 15
12
15
Difference of 12 and 15 is -3
```