NATIONAL INSTITUTE OF TECHNOLOGY UTTARAKHAND

OBJECT ORIENTED PROGRAMMING

CSL 253

Submitted By	Submitted To
Vidya Chauhan	Ms. Sneha Chauhan
BT23CSE015	

WRAPPER CLASSES

In Java, **wrapper classes** are special classes that allow us to treat primitive data types (like int, char, boolean, etc.) as objects. Normally, primitive types are simple values, not objects. But sometimes, we need to work with objects instead of basic values—especially when dealing with things like **collections** (e.g., ArrayList or HashMap), which can only store objects.

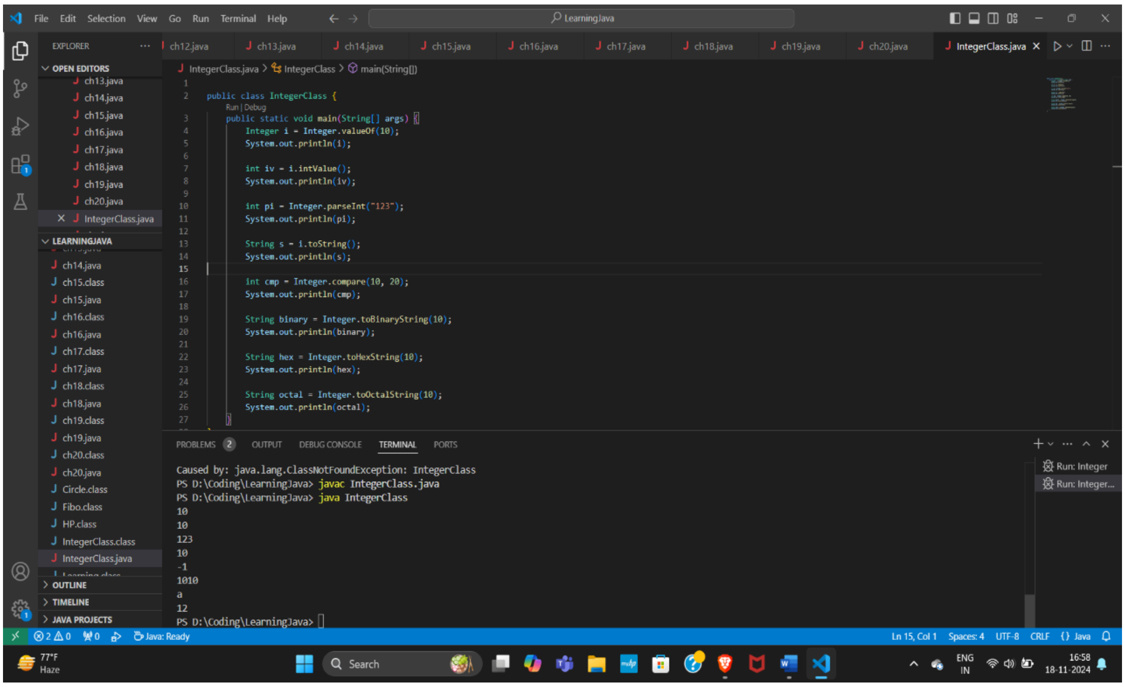
For example, if you need to store a list of numbers in an ArrayList, you can't directly use primitive types like int or double. Instead, you use their corresponding **wrapper classes**: Integer for int, Double for double, and so on.

Wrapper Classes in Java

Here's a list of the primitive types and their matching wrapper classes:

Primitive Type	**WrapperClass**
byte	Byte
short	Short
int	Integer
long	Long
float	Float
double	Double
char	Character
boolean	Boolean

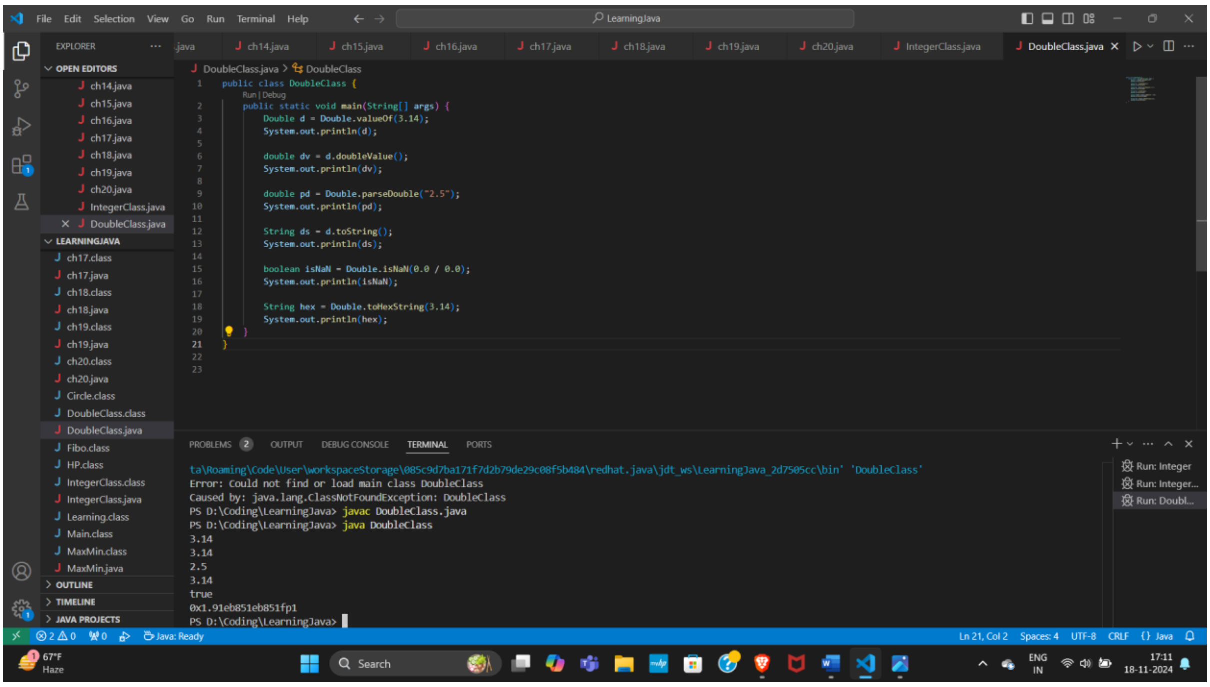
Integer Wrapper Class

- The Integer class in Java is a wrapper class for the primitive data type int. It provides methods to convert int to Integer and vice versa, along with many utility methods for performing operations on integers.
- Used when you need to treat primitive int values as objects, especially in collections like ArrayList that cannot hold primitive types.
- 

```java
public class IntegerClass {
    public static void main(String[] args) {
        Integer i = Integer.valueOf(10);
        System.out.println(i);

        int iv = i.intValue();
        System.out.println(iv);

        int pi = Integer.parseInt("123");
        System.out.println(pi);

        String s = i.toString();
        System.out.println(s);

        int cmp = Integer.compare(10, 20);
        System.out.println(cmp);

        String binary = Integer.toBinaryString(10);
        System.out.println(binary);

        String hex = Integer.toHexString(10);
        System.out.println(hex);

        String octal = Integer.toOctalString(10);
        System.out.println(octal);
    }
```

```
Caused by: java.lang.ClassNotFoundException: IntegerClass
PS D:\Coding\LearningJava> javac IntegerClass.java
PS D:\Coding\LearningJava> java IntegerClass
10
10
123
10
-1
1010
a
12
PS D:\Coding\LearningJava>
```

Double Wrapper Class

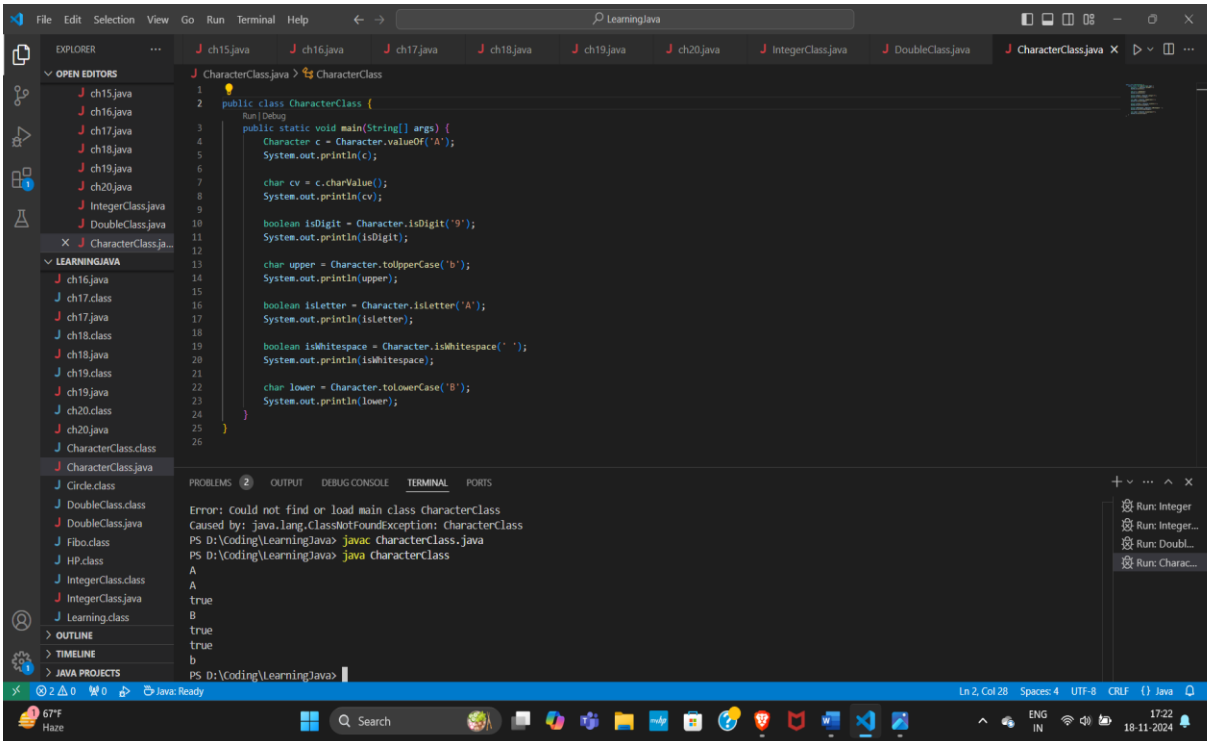
- The Double class is a wrapper for the double primitive type. It allows you to treat double values as objects, and provides methods for parsing and converting double values.
- Useful for performing complex mathematical operations and handling floating-point numbers with more precision.
- 

```java
public class DoubleClass {
    public static void main(String[] args) {
        Double d = Double.valueOf(3.14);
        System.out.println(d);

        double dv = d.doubleValue();
        System.out.println(dv);

        double pd = Double.parseDouble("2.5");
        System.out.println(pd);

        String ds = d.toString();
        System.out.println(ds);

        boolean isNaN = Double.isNaN(0.0 / 0.0);
        System.out.println(isNaN);

        String hex = Double.toHexString(3.14);
        System.out.println(hex);
    }
}
```

```
ta\Roaming\Code\User\workspaceStorage\085c9d7ba171f7d2b79de29c08f5b484\redhat.java\jdt_ws\LearningJava_2d7505cc\bin' 'DoubleClass'
Error: Could not find or load main class DoubleClass
Caused by: java.lang.ClassNotFoundException: DoubleClass
PS D:\Coding\LearningJava> javac DoubleClass.java
PS D:\Coding\LearningJava> java DoubleClass
3.14
3.14
2.5
3.14
true
0x1.91eb851eb851fp1
PS D:\Coding\LearningJava>
```

Character Wrapper Class

- Useful for performing complex mathematical operations and handling floating-point numbers with more precision.
- Useful for operations on characters, such as checking if a character is a letter, digit, or whitespace, or converting it to uppercase/lowercase.

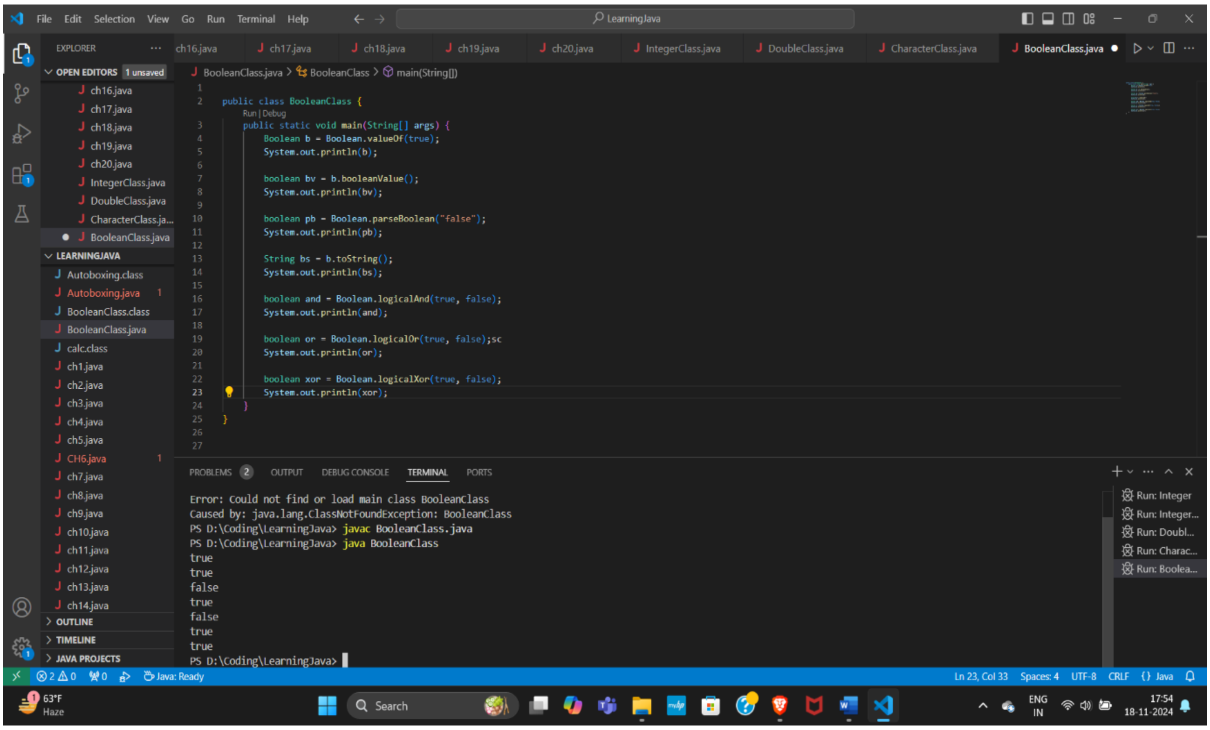


```java

public class CharacterClass {
    public static void main(String[] args) {
        Character c = Character.valueOf('A');
        System.out.println(c);

        char cv = c.charValue();
        System.out.println(cv);

        boolean isDigit = Character.isDigit('9');
        System.out.println(isDigit);

        char upper = Character.toUpperCase('b');
        System.out.println(upper);

        boolean isLetter = Character.isLetter('A');
        System.out.println(isLetter);

        boolean isWhitespace = Character.isWhitespace(' ');
        System.out.println(isWhitespace);

        char lower = Character.toLowerCase('B');
        System.out.println(lower);
    }
}

```

```
Error: Could not find or load main class CharacterClass
Caused by: java.lang.ClassNotFoundException: CharacterClass
PS D:\Coding\LearningJava> javac CharacterClass.java
PS D:\Coding\LearningJava> java CharacterClass
A
A
true
B
true
true
b
PS D:\Coding\LearningJava>
```

Boolean Wrapper Class

- The Boolean class is a wrapper for the boolean primitive type. It is used for handling boolean values (true or false) as objects.
- Used in scenarios where boolean values need to be stored in collections, or when performing logical operations on boolean values.

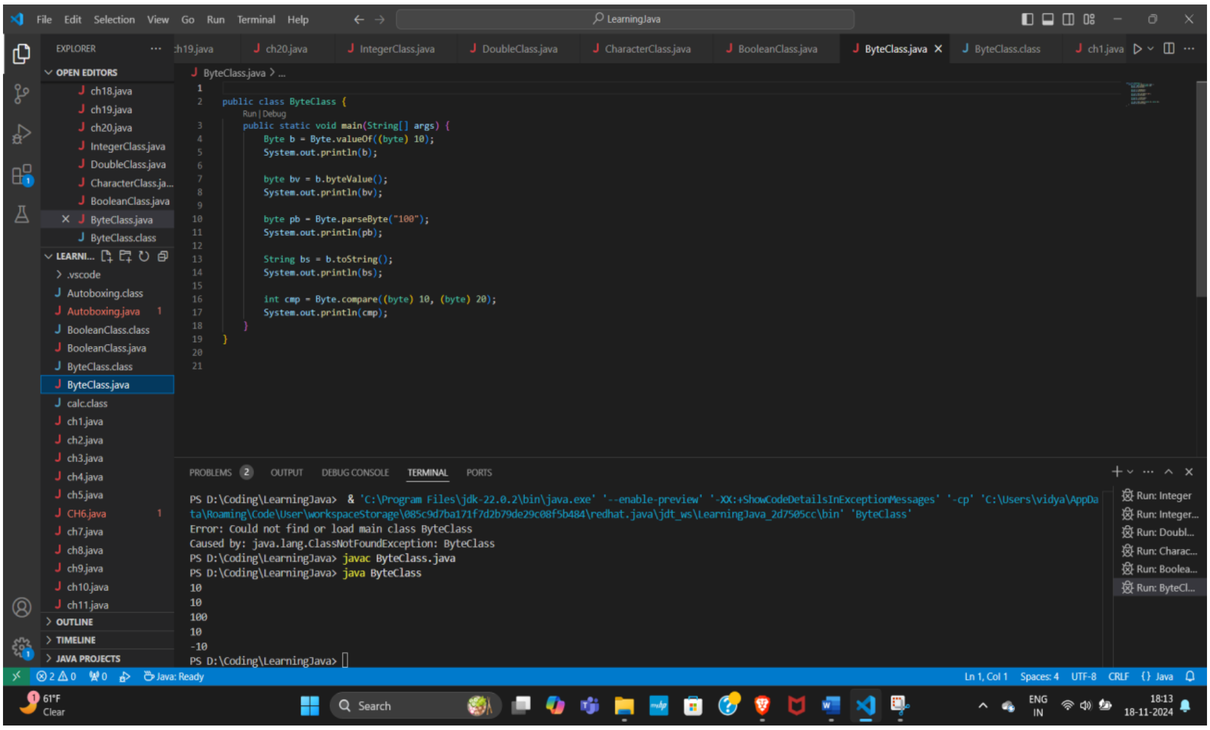


```java

public class BooleanClass {
    public static void main(String[] args) {
        Boolean b = Boolean.valueOf(true);
        System.out.println(b);

        boolean bv = b.booleanValue();
        System.out.println(bv);

        boolean pb = Boolean.parseBoolean("false");
        System.out.println(pb);

        String bs = b.toString();
        System.out.println(bs);

        boolean and = Boolean.logicalAnd(true, false);
        System.out.println(and);

        boolean or = Boolean.logicalOr(true, false);sc
        System.out.println(or);

        boolean xor = Boolean.logicalXor(true, false);
        System.out.println(xor);
    }
}
```

```
Error: Could not find or load main class BooleanClass
Caused by: java.lang.ClassNotFoundException: BooleanClass
PS D:\Coding\LearningJava> javac BooleanClass.java
PS D:\Coding\LearningJava> java BooleanClass
true
true
false
true
false
true
true
PS D:\Coding\LearningJava>
```

Byte Wrapper Class

- The Byte class is a wrapper for the byte primitive type. It allows for the manipulation and conversion of byte values as objects
- Often used when working with byte-level operations, such as file manipulation or network communication where byte values are important

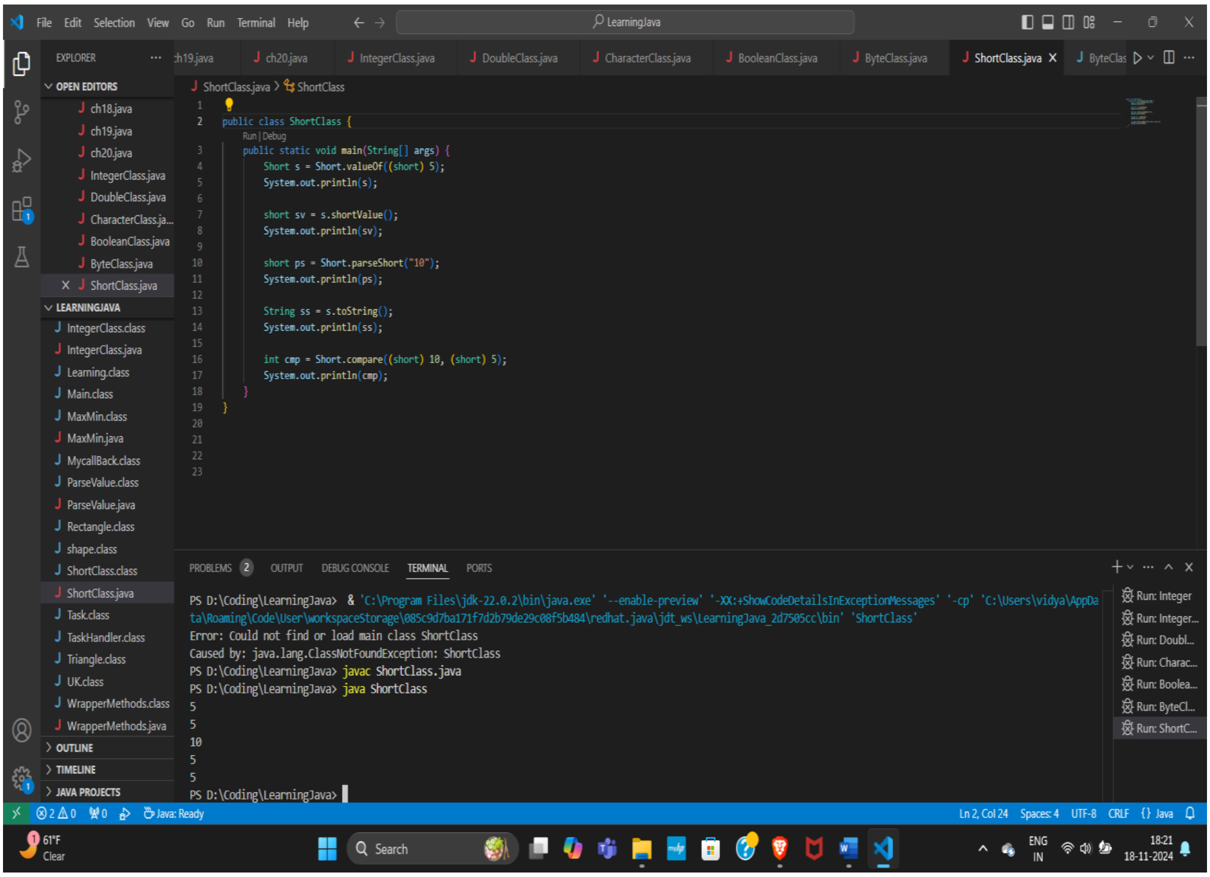


```java

public class ByteClass {
    public static void main(String[] args) {
        Byte b = Byte.valueOf((byte) 10);
        System.out.println(b);

        byte bv = b.byteValue();
        System.out.println(bv);

        byte pb = Byte.parseByte("100");
        System.out.println(pb);

        String bs = b.toString();
        System.out.println(bs);

        int cmp = Byte.compare((byte) 10, (byte) 20);
        System.out.println(cmp);
    }
}
```

```
PS D:\Coding\LearningJava>  & 'C:\Program Files\jdk-22.0.2\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\vidya\AppData\Roaming\Code\User\workspaceStorage\085c9d7ba171f7d2b79de29c08f5b484\redhat.java\jdt_ws\LearningJava_2d7505cc\bin' 'ByteClass'
Error: Could not find or load main class ByteClass
Caused by: java.lang.ClassNotFoundException: ByteClass
PS D:\Coding\LearningJava> javac ByteClass.java
PS D:\Coding\LearningJava> java ByteClass
10
10
100
10
-10
PS D:\Coding\LearningJava> 
```

Short Wrapper Class

- The Short class is a wrapper for the short primitive type. It is useful for treating short values as objects and performing utility operations.
- Typically used in situations where memory optimization is important, and `short` values need to be treated as objects, such as in large datasets.

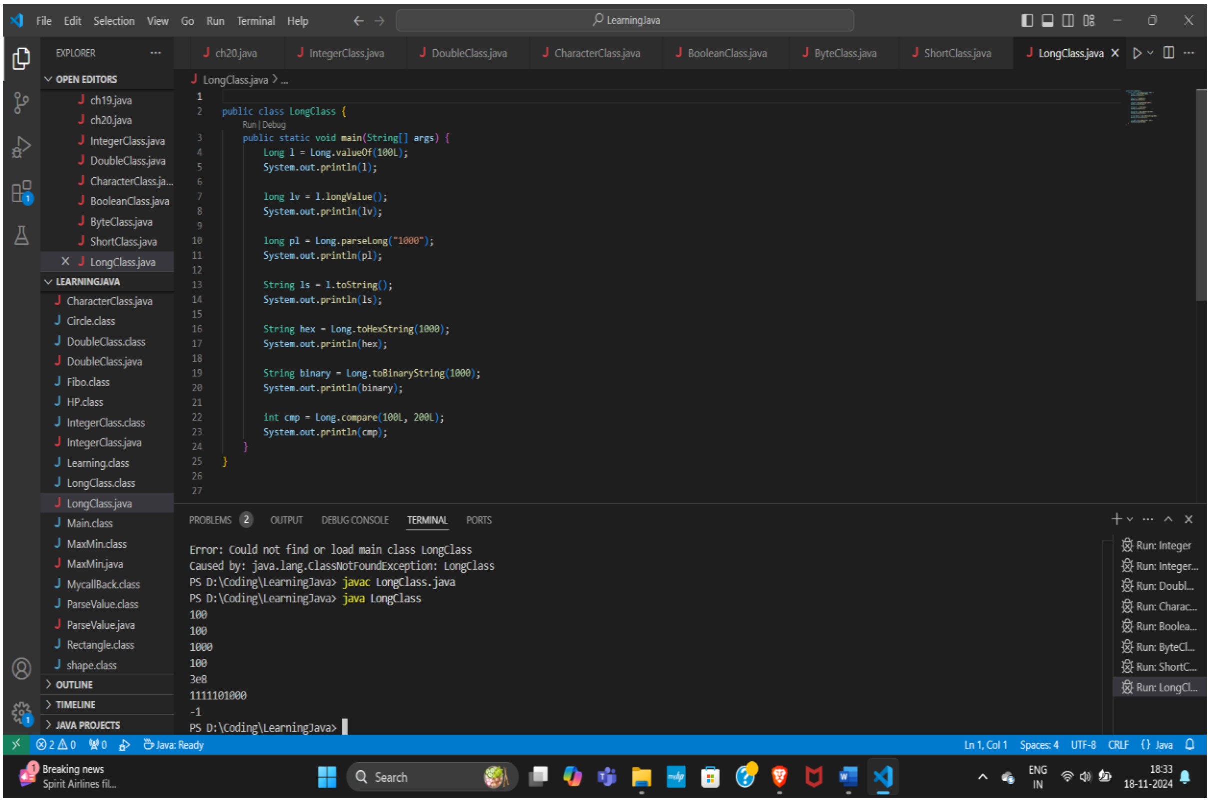


```java
public class ShortClass {
    public static void main(String[] args) {
        Short s = Short.valueOf((short) 5);
        System.out.println(s);

        short sv = s.shortValue();
        System.out.println(sv);

        short ps = Short.parseShort("10");
        System.out.println(ps);

        String ss = s.toString();
        System.out.println(ss);

        int cmp = Short.compare((short) 10, (short) 5);
        System.out.println(cmp);
    }
}
```

```
PS D:\Coding\LearningJava>  & 'C:\Program Files\jdk-22.0.2\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\vidya\AppData\Roaming\Code\User\workspaceStorage\085c9d7ba171f7d2b79de29c08f5b484\redhat.java\jdt_ws\LearningJava_2d7505cc\bin' 'ShortClass'
Error: Could not find or load main class ShortClass
Caused by: java.lang.ClassNotFoundException: ShortClass
PS D:\Coding\LearningJava> javac ShortClass.java
PS D:\Coding\LearningJava> java ShortClass
5
5
10
5
5
PS D:\Coding\LearningJava>
```

Long Wrapper Class

- The Long class is a wrapper for the long primitive type. It provides utility methods to work with long values as objects.
- Used when dealing with large numeric values, especially for monetary calculations or high-precision timestamps

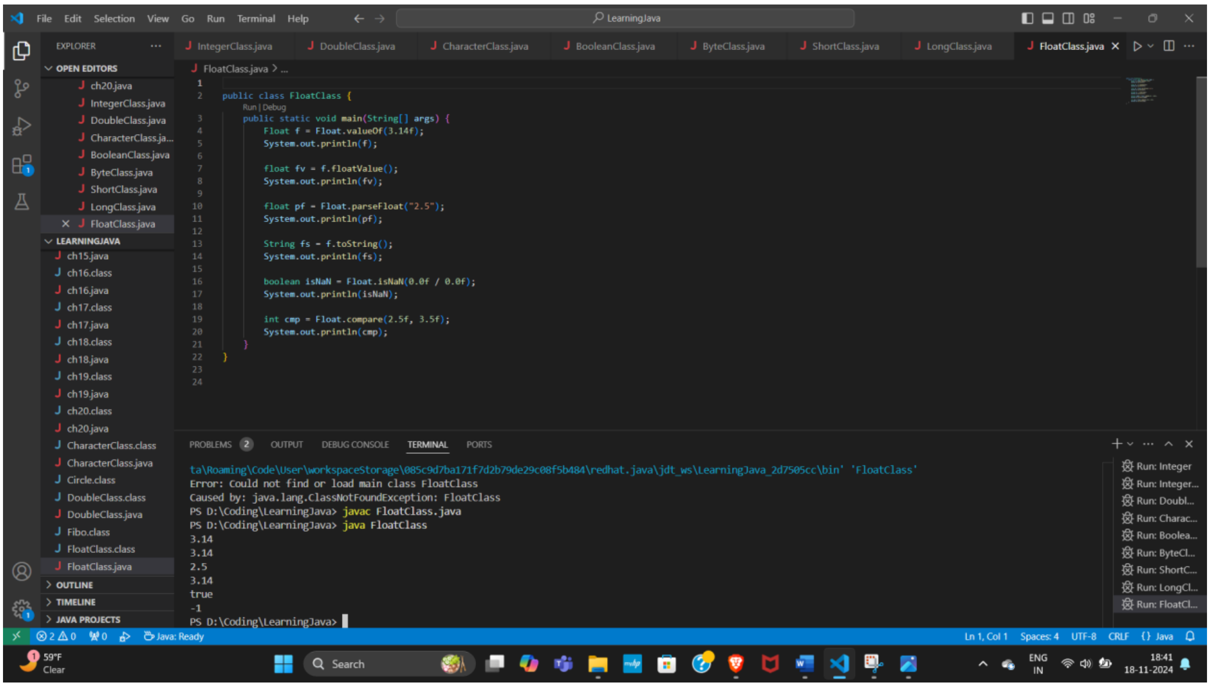


```java

public class LongClass {
    public static void main(String[] args) {
        Long l = Long.valueOf(100L);
        System.out.println(l);

        long lv = l.longValue();
        System.out.println(lv);

        long pl = Long.parseLong("1000");
        System.out.println(pl);

        String ls = l.toString();
        System.out.println(ls);

        String hex = Long.toHexString(1000);
        System.out.println(hex);

        String binary = Long.toBinaryString(1000);
        System.out.println(binary);

        int cmp = Long.compare(100L, 200L);
        System.out.println(cmp);
    }
}


```

```
Error: Could not find or load main class LongClass
Caused by: java.lang.ClassNotFoundException: LongClass
PS D:\Coding\LearningJava> javac LongClass.java
PS D:\Coding\LearningJava> java LongClass
100
100
1000
100
3e8
1111101000
-1
PS D:\Coding\LearningJava>
```

Float Wrapper Class

- The Float class is a wrapper for the float primitive type. It provides methods to convert and manipulate float values.
- Used in scenarios where floating-point precision is important but memory usage needs to be optimized compared to using `double`.



```java
public class FloatClass {
    public static void main(String[] args) {
        Float f = Float.valueOf(3.14f);
        System.out.println(f);

        float fv = f.floatValue();
        System.out.println(fv);

        float pf = Float.parseFloat("2.5");
        System.out.println(pf);

        String fs = f.toString();
        System.out.println(fs);

        boolean isNaN = Float.isNaN(0.0f / 0.0f);
        System.out.println(isNaN);

        int cmp = Float.compare(2.5f, 3.5f);
        System.out.println(cmp);
    }
}
```

```
ta\Roaming\Code\User\workspaceStorage\085c9d7ba171f7d2b79de29c08f5b484\redhat.java\jdt_ws\LearningJava_2d7505cc\bin' 'FloatClass'
Error: Could not find or load main class FloatClass
Caused by: java.lang.ClassNotFoundException: FloatClass
PS D:\Coding\LearningJava> javac FloatClass.java
PS D:\Coding\LearningJava> java FloatClass
3.14
3.14
2.5
3.14
true
-1
PS D:\Coding\LearningJava>
```

CONCLUSION

Wrapper classes are useful in Java when you need to treat primitive types (int, double, boolean, etc.) as objects. This is particularly helpful in situations where primitive types cannot be used directly, such as in collections like ArrayList, which only work with objects. The wrapper classes also provide utility methods to convert between types, perform conversions, and handle operations more conveniently.