

Day2

July-20 CoreJava

Q1. Explore JIT -

<https://www.geeksforgeeks.org/jvm-works-jvm-architecture/>

ANS:-

The Just-In-Time (JIT) compiler is a component of the Java™ Runtime Environment that improves the performance of Java applications at run time.

Java programs consists of classes, which contain platform-neutral bytecodes that can be interpreted by a JVM on many different computer architectures. At run time, the JVM loads the class files, determines the semantics of each individual bytecode, and performs the appropriate computation. The additional processor and memory usage during interpretation means that a Java application performs more slowly than a native application. The JIT compiler helps improve the performance of Java programs by compiling bytecodes into native machine code at run time.

RESOURCE :-

<https://www.ibm.com/docs/en/sdk-java-technology/7.1?topic=compiler-jit-overview>

Q2. Explore the class file of any existing code and check it.

ANS:-

A Java class file is a file (with the `.class` extension) containing java bytecode that can be executed on the java virtual machine . A Java class file is usually produced by a java compiler from java programming language source file (`.java` files) containing Java classes. If a source file has more than one class, each class is compiled into a separate class file.

This is how the `.java` file looks when opened in notepad:-

```

class Hello
{
    public static void main(String args[])
    { System.out.print("hello world");
    }
}

```

And this is how the .class file looks when open it with javap command :-

```

Microsoft Windows [Version 10.0.19042.1706]
(c) Microsoft Corporation. All rights reserved.

C:\Users\coditas\Desktop\java>javap Hello
Compiled from "HelloWorld_Batch2.java"
class Hello {
    Hello();
    public static void main(java.lang.String[]);
}
C:\Users\coditas\Desktop\java>

```

Q3. difference between access specifier and access modifier?

ANS:-

Java provides entities called "Access Modifiers or access specifiers" that help us to restrict the scope or visibility of a package, class, constructor, methods, variables, or other data members.

These access modifiers are also called "Visibility Specifiers".

Access modifiers:

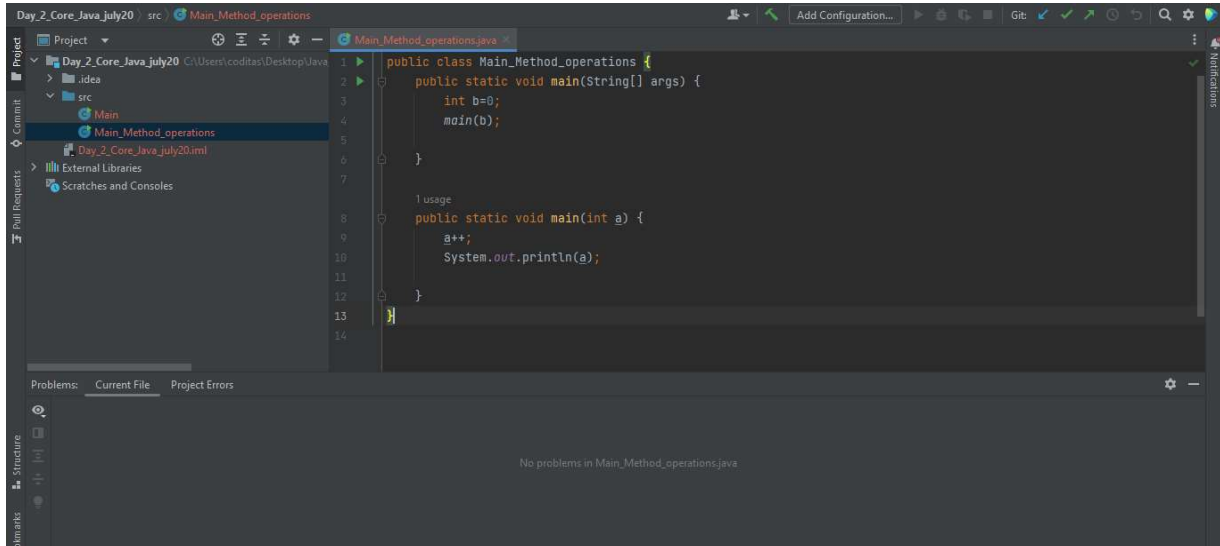
- 1.Default : When no access modifier is specified, it is treated as default modifier. Its scope is limited within the package.
- 2.Public: The word itself indicates that it has scope everywhere, i.e; it is visible everywhere even outside the package.
- 3.Private: It has scope only within the class
- 4.Protected : Its scope limits within the package and all sub classes.

Non-access modifiers are those keywords which do not have anything related to the level of access but they provide a special functionality when specified.

Eg:- Final, Strictfp, Static, Abstract.

Q4. can we have multiple main methods in class?

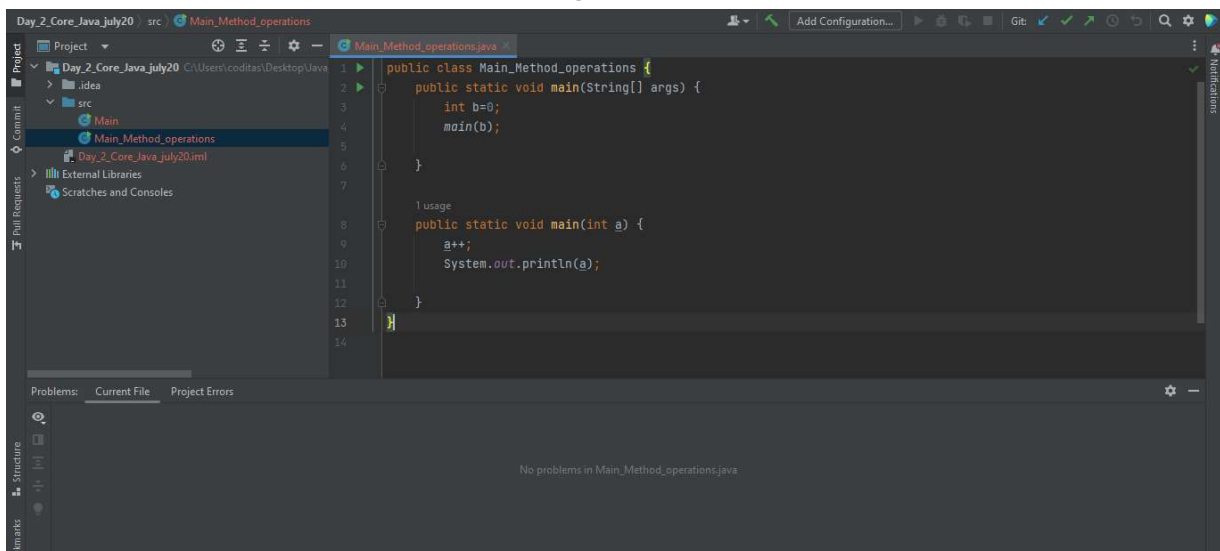
ANS:- Java can have multiple main methods but with the concept of overloading. I.e. same name but different parameter.



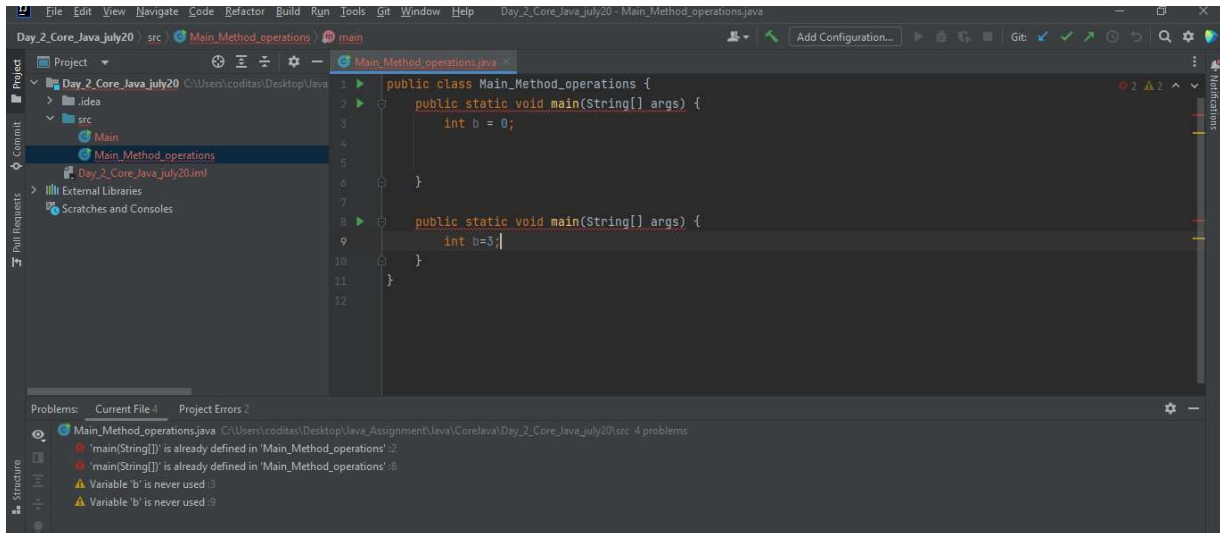
Q5. Can we overload and override the main method?

ANS:- It's possible to overload main in Java but it's not possible to override it, simply because it's a static method.

We have overloaded the main method in given snippet.



We have tried to override the main method but it gives the error



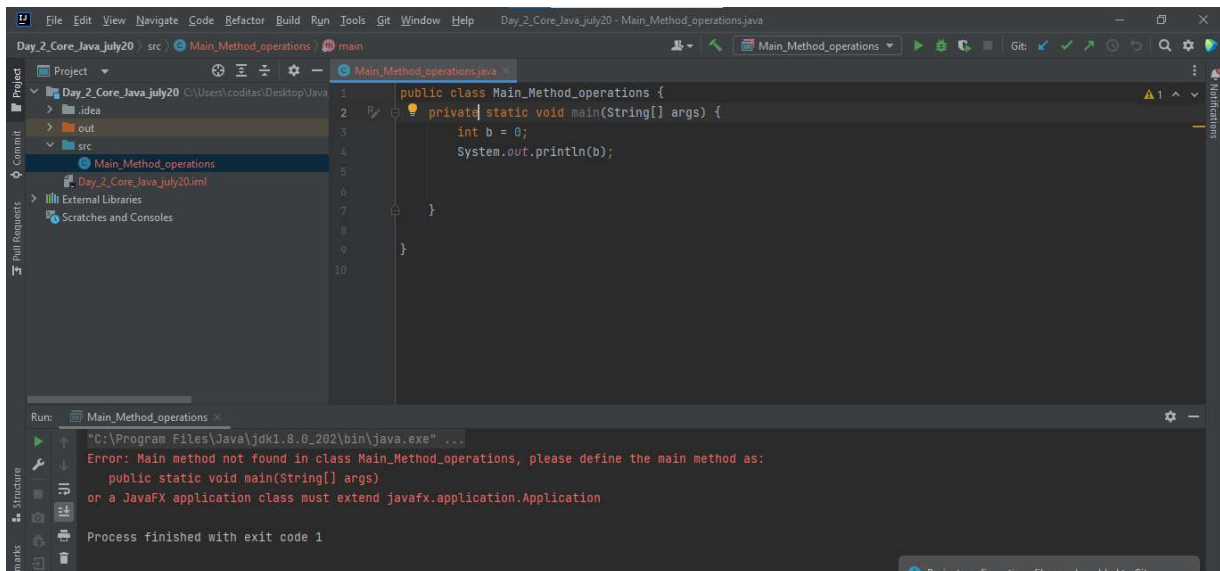
Q6. Can I write the main method as private, protected and default?

ANS:-

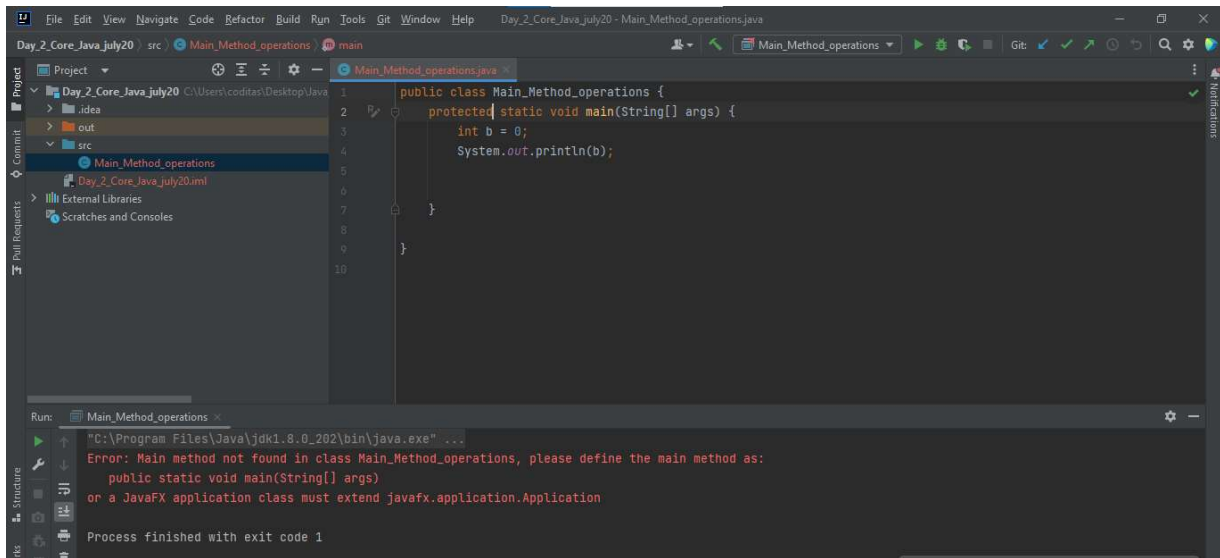
We can write it as a private protected and default and it will also get compiled but at runtime it will say that main method is not public.

So the jvm does not consider this as the entry point of the program.

For private:-



For protected:-



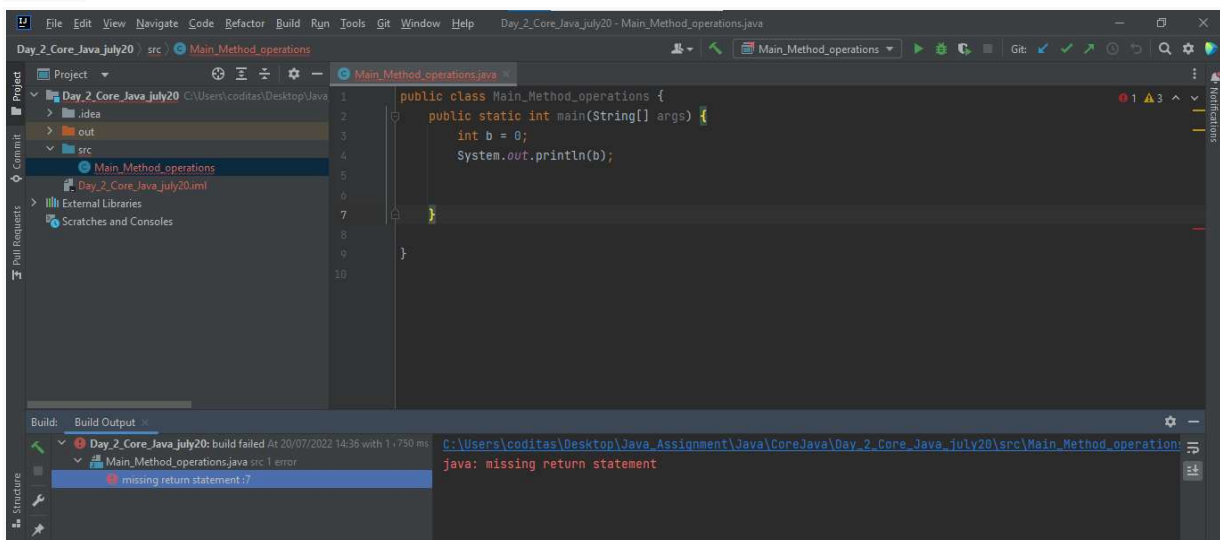
Q7. without a main method can we execute our code? How?

ANS:- Yes, we can execute a java program without a main method by using a static block before the version 1.7. Static block in Java is a group of statements that gets executed only once when the class is loaded into the memory by Java ClassLoader, It is also known as a static initialization block.

Q8. Can we change the return type of the main() method from void to any other data type

like int return?

ANS:-



No we can not change the return type of main method.

Java main method does not return anything that is why its return type is void. If we try to return anything from main method, it will give an unexpected value error because it is predefined signature in JVM.

Q9. explore keyword strictfp?

ANS:- Released in version 1.2

strictfp is used to ensure that floating points operations give the same result on any platform. As floating points precision may vary from one platform to another. strictfp keyword ensures the consistency across the platforms. strictfp can be applied to class, method or on interfaces but cannot be applied to abstract methods, variable or on constructors.