# CORE JAVA ASSIGNMENT
# DAY - 2

*By Devayush Bajaj*

Q1) Explore J console (youtube shiva reddy) - FRIDAY DEADLINE

**Q2) explore JIT - (gig for gigs)**

ANSWER -

**VIRTUAL MACHINE** - A virtual machine is a virtual representation of a physical computer. We can call the virtual machine the guest machine, and the physical computer it runs on is the host machine.

**COMPILE AND INTERPRETED LANGAUGES**

- In programming languages like C and C++, the code is first compiled into platform-specific machine code. These languages are called **compiled languages.**
- On the other hand, in languages like JavaScript and Python, the computer executes the instructions directly without having to compile them. These languages are called **interpreted languages.**

**What is the Java Virtual Machine?**

- Java uses a combination of both techniques. Java code is first compiled into **byte code** to **generate** a **class file**. This **class file** is then **interpreted** by the **Java Virtual Machine** for the **underlying platform**. The **same class file** can be executed **on any version** of **JVM running** on **any platform** and **operating system.**
- Similar to virtual machines, the **JVM creates** an **isolated space** on a **host machine**. This space can be used to **execute Java programs irrespective** of the **platform** or **operating system** of the machine.

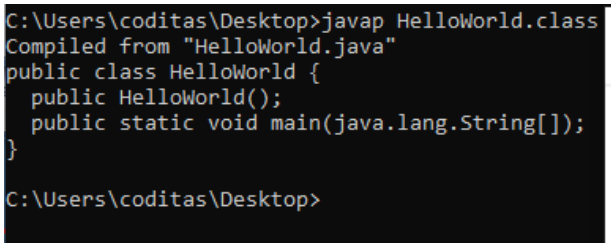
**JIT**

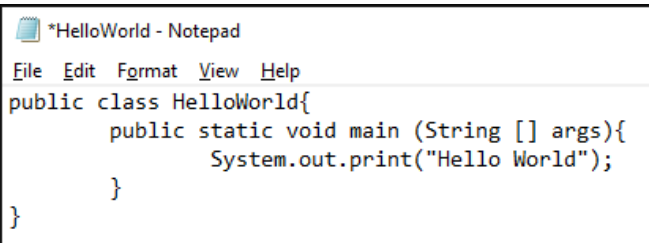
- Just in time (JIT) is a part of JVM whose function is to convert byte code into machine code (JVM executes the byte code).
- Execution of byte code is slower then the execution of machine code because first JVM translates bytecode into machine code

- To solve the above problem an compiler is given to JVM whose name is JIT who will compile the executed bytecode into machine code. WHich also results in improving the performance of JVM

Q3) explore .class file.

ANSWER




JAVA CODE CLASS FILE CODE

Default constructor has been created in class file

In .class file we can see an object of Helloworld has been created after reading class name

In Java file we are having (String [] arhs) and in class file we can see java.util.String[] because String is not a data type it is an object present in UTIL pakage)

Q4) difference between access specifier and access modifier?

- 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:**

**There are 4 types of 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.

Q6) can we have multiple main methods in class?

ANSWER:

1)

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

2)

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

3)

```
public static void main(String[] arguments) { }
```

4)

```
public static void main(String configuration[]) { }
```

Q7) can we overload and override the main method?

We can overload the main method but we cannot override it.

```
devayush-bajaj *
public class demo {
    devayush-bajaj *
    protected static void main(String[] args) {
        System.out.println("Hello Java");

    }
}
```

```
demo
"C:\Program Files\Java\jdk1.8.0_202\bin\java.exe" ...
Hello Java
Hello Coditas

Process finished with exit code 0
```

Q8) can I write main method as protected and default?

**PROTRACTED:-**

No, We cannot make main method as protracted it will give error "Main method not found in class demo"

```
devayush-bajaj *
public class demo {
    devayush-bajaj *
    static void main(String[] args) {
        System.out.println("Hello Java");

    }
}
```

```
"C:\Program Files\Java\jdk1.8.0_202\bin\java.exe" ...
Error: Main method not found in class demo, please define the main method as:
   public static void main(String[] args)
or a JavaFX application class must extend javafx.application.Application

Process finished with exit code 1
```

**DEFAULT:-**

**No, we cannot make main method default** it will give error "Main method not found in class demo"

```
1
    devayush-bajaj *
2   public class demo {
        devayush-bajaj *
3       static void main(String[] args) {
4           System.out.println("Hello Java");
5
6       }
7   }
```

```
"C:\Program Files\Java\jdk1.8.0_202\bin\java.exe" ...
Error: Main method not found in class demo, please define the main method as:
   public static void main(String[] args)
or a JavaFX application class must extend javafx.application.Application

Process finished with exit code 1
```

Q9) without a main method can we execute our code? How?

Yes, it can can execute a java program without a main method by using a static block.

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.

**But this will not execute with Java 7 version or later.**

```
devayush-bajaj *
class demo{
    static {
        System.out.println("class without a main method");
    }
}
```

```
"C:\Program Files\Java\jdk1.8.0_202\bin\java.exe" ...
Error: Main method not found in class demo, please define the main method as:
   public static void main(String[] args)
or a JavaFX application class must extend javafx.application.Application

Process finished with exit code 1
```

Q10) can we change return type of main() method from void?

You can write the main method in your program with return type other than void, the program gets compiled without compilation errors.

But, at the time of execution JVM does not consider this new method (with return type other than void) as the entry point of the program

Q11) explore strict fp

Java strictfp keyword ensures that you will get the same result on every platform if you perform operations in the floating-point variable. The precision may differ from platform to platform. That is why the Java programming language has provided the strictfp keyword, so that you get the same result on every platform. So, now you have better control over the floating-point arithmetic.