Video. Important Components of Java.

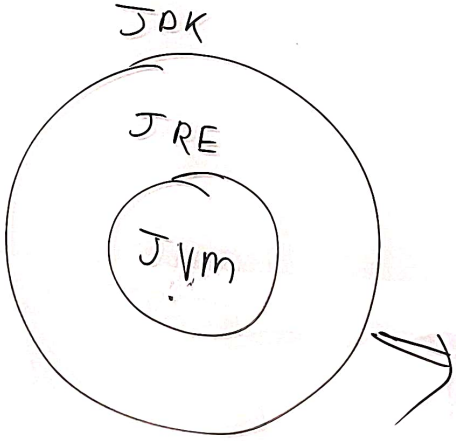
3 main things are in Java.

1). JDK → Java Development Kit → Biggest
2) JRE → Java Runtime Environment → Medium
3). JVM → Java Virtual Machine → Smallest.

Size are hierarchy of this is in Alphabetical order.

[JDK, JRE, JVM]

[D, R, V]



Now whole thing can be explained.

1) Lets say You wrote a code [hello.java]
2) You go to the terminal and say javac hello.java
3). Then You execute .hello

Lets Explain this.

To run a java program you need JDK (100%)

Why?

JDK has Javac (Responsible for compiling and converting java code to byte code).

(1)

hello.java —Javac→ hello (Byte code)

(2) This byte code is taken by JVM to convert into machine code.

ByteCode —JVM→ machine code.

As JDK contains Javac

Similarly JVM contains JIT Just In Time Compiler

So

.java file $\xrightarrow[\text{JDK}]{\text{Javac}}$ .bytecode $\xrightarrow[\text{JIT}]{\text{Jvm}}$ .machine code → to be run by Operating System.

What is the role of JRE then?

well JRE → Java Runtime Env

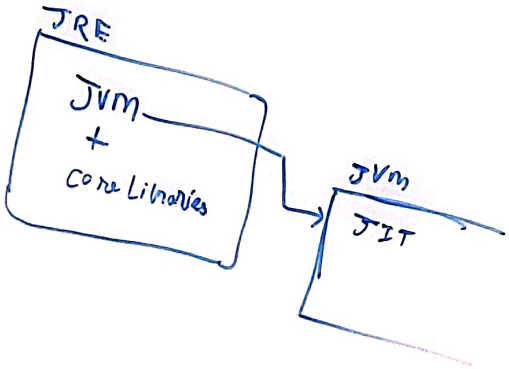
= JDK + core libraries.

What is this?

So in Java, there are lot of libraries like Math.max( ), math.pow( ) etc.

You can think of it as tools required to run the program.

JDK = javac +



If you compare Java program to making a Pizza.

JDK → whole Kitchen.

javac → (Knives & tools to cut the Pizza)

JRE → JVM + Libraries (Ingridients + Oven).

Libraries → Ingridients like dough, spices etc.

JVM → OVEN

This is the reason. JAVA is called platform Independent.

Compile Java code → .byteCode [Give it to any platform (Windows, Linux, Mac) it will run.

WORA → Write Once Run Anywhere.

*** JVM is not platform independent. It is platform Dependent.

Windows → windows JVM.
MAC → macJVM
Linux → LinuxJvm

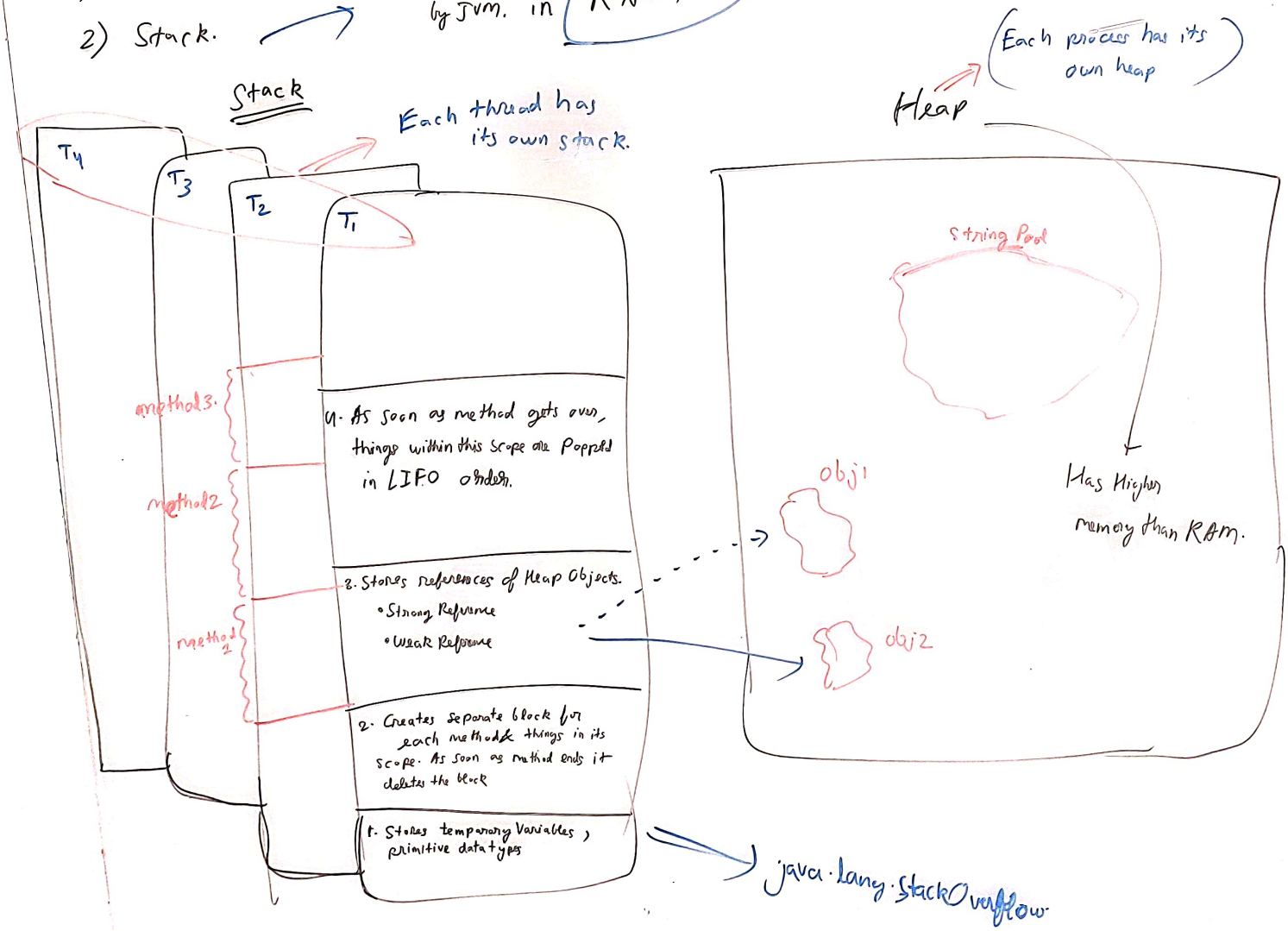
→ Since each Operating system has a different architecture, & set of assembly instructions. Hence JVM needs to convert byte code accordingly to machine code.

Video 9. Java Memory Management & Garbage Collection.

2 types of Memory

1) Heap.
2) Stack.

→ Both are created by JVM. in RAM.



Understand with Code

```
public class MemoryManagement{

    public static void main(){

        int primitiveVariable = 10;

        Person personObj = new Person()

        String stringLiteral1 = "24";

        MemoryManagement memObj = new MemoryManagement();

        memObj.memoryManagementTest(personObj);

    }

    private void memoryManagementTest(Person personObj){

        Person personObj1 = personObj;

        String stringLiteral2 = "24";

        String stringLiteral3 = new String("24");

    }

}
```

Since new keyword