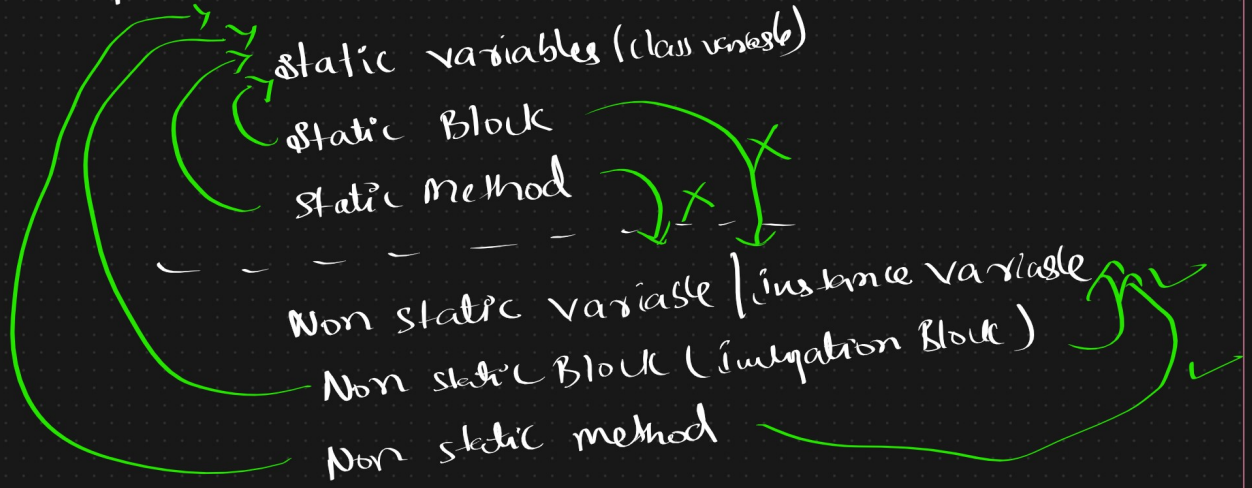
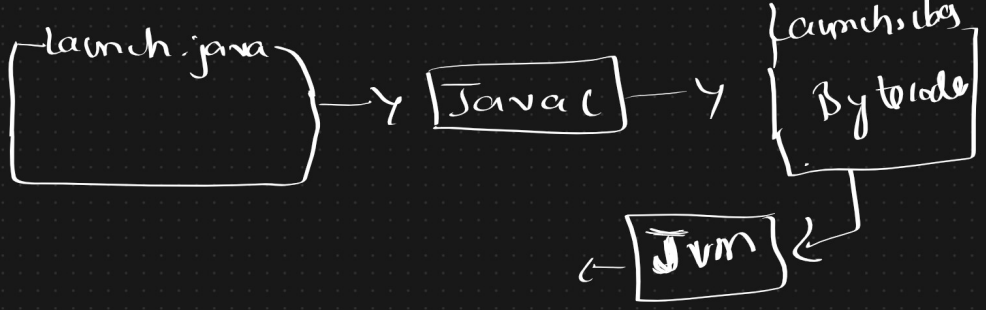
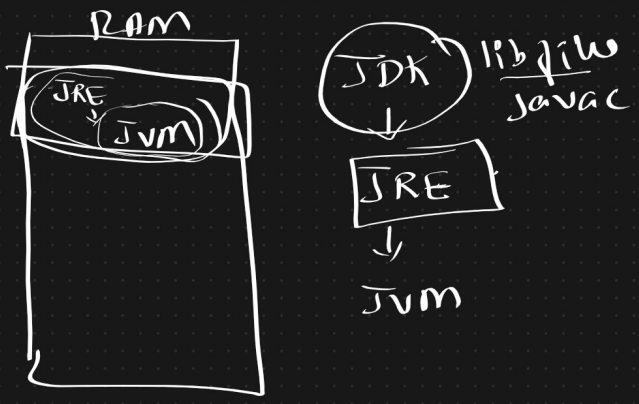
=> Static keyword

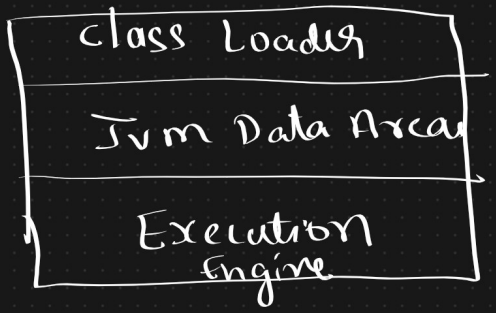



=> (3) main method => { (1) Static var
(2) Static Block




Jvm =>



## 1) By class loader subsystem

- (a) Loading
  - Bootstrap loader
  - Extension loader
  - Application
- (b) Linking
  - verify
  - prepare
  - Resolve
- (c) Initialization
  - { static block }

Heap

## JVM Runtime Data areas

interview

| Method area | stack area | Heap | PC Register | Native stack |
|---|---|---|---|---|
| Many class method code | method call stack frame<br>ref var<br>local | Object<br>instance var<br>static var | address of each currently running instruction | C, C++ |

## Execution Engine

for ( ) { }

| Interpreter | JIT compiler | GC |
|---|---|---|

JNI

low

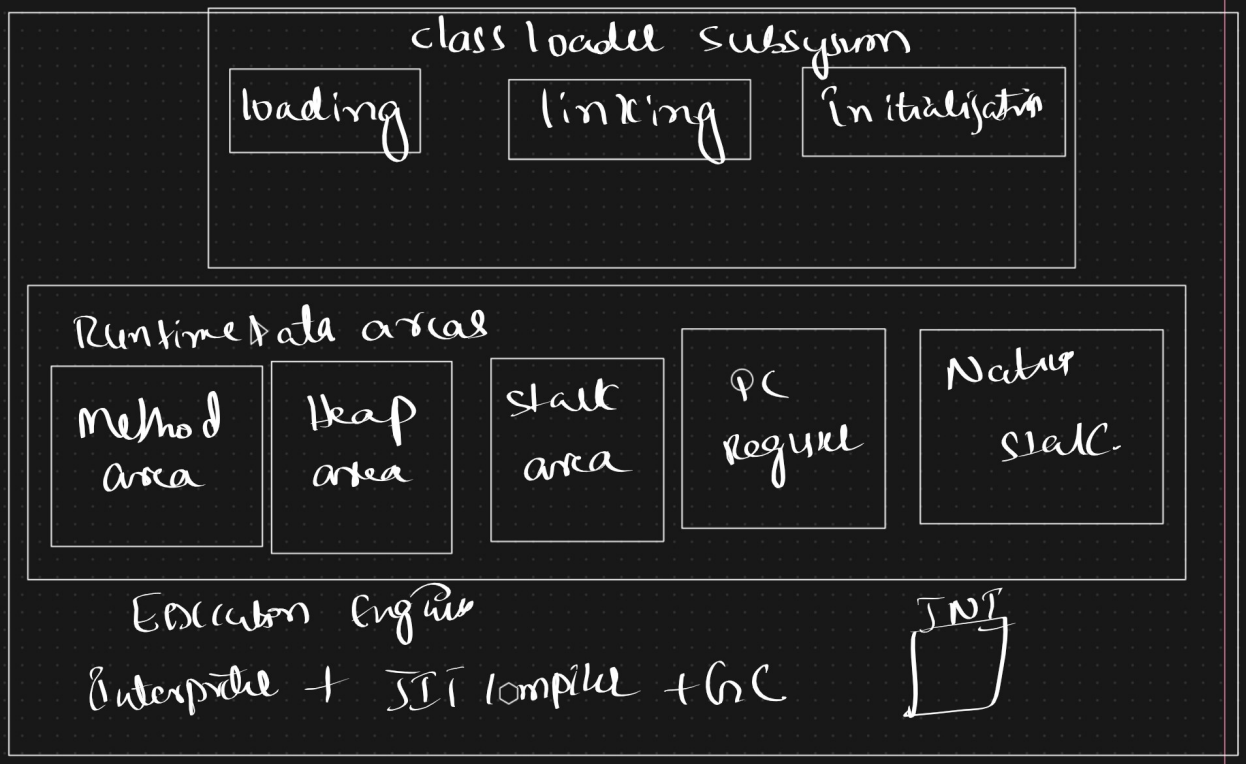
01001

Java → Javac → Byte code → Interpreter

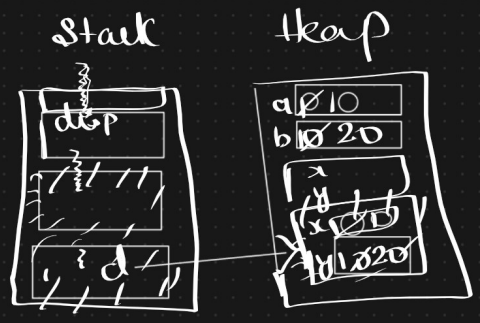
Java - Hybrid
Both compiler & Interpreter

| Interpreted | compiled | Java |
|---|---|---|
| Python | C | compiled + Interpreted |
| JS | C++ | |

JVM




```java
class Demo1
{
    static int a,b;
    int x,y;
    static
    {
        a=10;
        b=20;
        System.out.println("Static block");
    }
    {
        x=10;
        y=20;
        System.out.println("Non Static block/Java Block");
    }
    public Demo1()
    {
//      {
//          x=10;
//          y=20;
//          System.out.println("Non Static block/Java Block");
//      }
        System.out.println("Constructor");
    }
    public static void disp()
    {
        System.out.println("Value of a : " + a);
        System.out.println("Value of b : " + b);
    }

    public void disp2()
    {
        System.out.println("Value of x : " + x);
        System.out.println("Value of y : " + y);
    }
}
public class LaunchStNst {

    public static void main(String[] args)
    {
        Demo1.disp();

        //Demo1.disp2();

        Demo1 d=new Demo1();
        d.disp2();
        //d.disp();


    }

}
```







```java
import java.util.*;
class Farmer
{
    float pa;
    float td;
    float ri;
    float si;

    public void input()
    {
        Scanner scan=new Scanner(System.in);
        System.out.println("Please enter the principal amount required");
        pa=scan.nextFloat();
        System.out.println("Please enter time duration");
        td=scan.nextFloat();

        ri=2.5f;
    }

    public void compute()
    {
        si=pa*td*ri/100;
    }

    public void disp()
    {
        System.out.println("Si is : "+ si);
    }

}
public class LaunchLoan
{
    public static void main(String[] args)
    {
        System.out.println("Farmer load application: ");
        Farmer f1=new Farmer();
        Farmer f2=new Farmer();
        Farmer f3=new Farmer();

        f1.input();
        f1.compute();
        f1.disp();

        f2.input();
        f2.compute();
        f2.disp();

        f3.input();
        f3.compute();
        f3.disp();
```