

24/1/25. Java \Rightarrow 90% Companies uses Java from Fortune 500 Companies.

①

- \rightarrow Fastest language.
- \rightarrow Mostly used in Banking Sectors.
- \rightarrow highlevel, robust, Secured and object oriented Programming language.
- \rightarrow Till 1995, called as "Oak". "Oak already exists"
- \rightarrow Team members suggested new names like "dynamic", "revolutionary", "Silk", etc..
- \rightarrow In a coffee shop they name was born as "Java". (Java Island first coffee produced island)
- \rightarrow James Gosling & team members originally developed at "Sun microsystem".
- \rightarrow Jan 23, 1996 released JDK 1.0.
- \rightarrow write once, Run Anywhere (WORA).
- \rightarrow Nov 26, 2006 released JVM.

Java used in:-

- \rightarrow Desktop, web, mobile Applications.
- \rightarrow Robotics.
- \rightarrow Games.
- \rightarrow Banking.

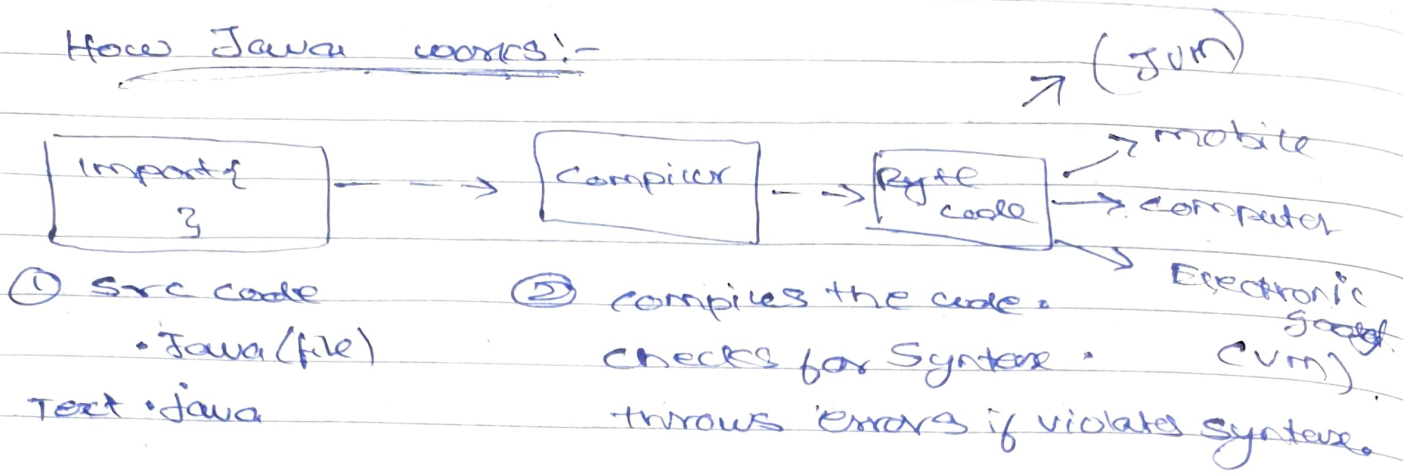
Types of Java applications:-

- \rightarrow standalone application, \rightarrow mobile Application.
- \rightarrow web Application.
- \rightarrow Enterprise Application.

Features of Java:-

- | | |
|------------------------------------|---------------------------------|
| \rightarrow simple | \rightarrow dynamic |
| \rightarrow object oriented | \rightarrow Robust |
| \rightarrow portable | \rightarrow high performance. |
| \rightarrow platform independent | \rightarrow multithreaded. |
| \rightarrow secured. | \rightarrow Distributed |

How Java works:-

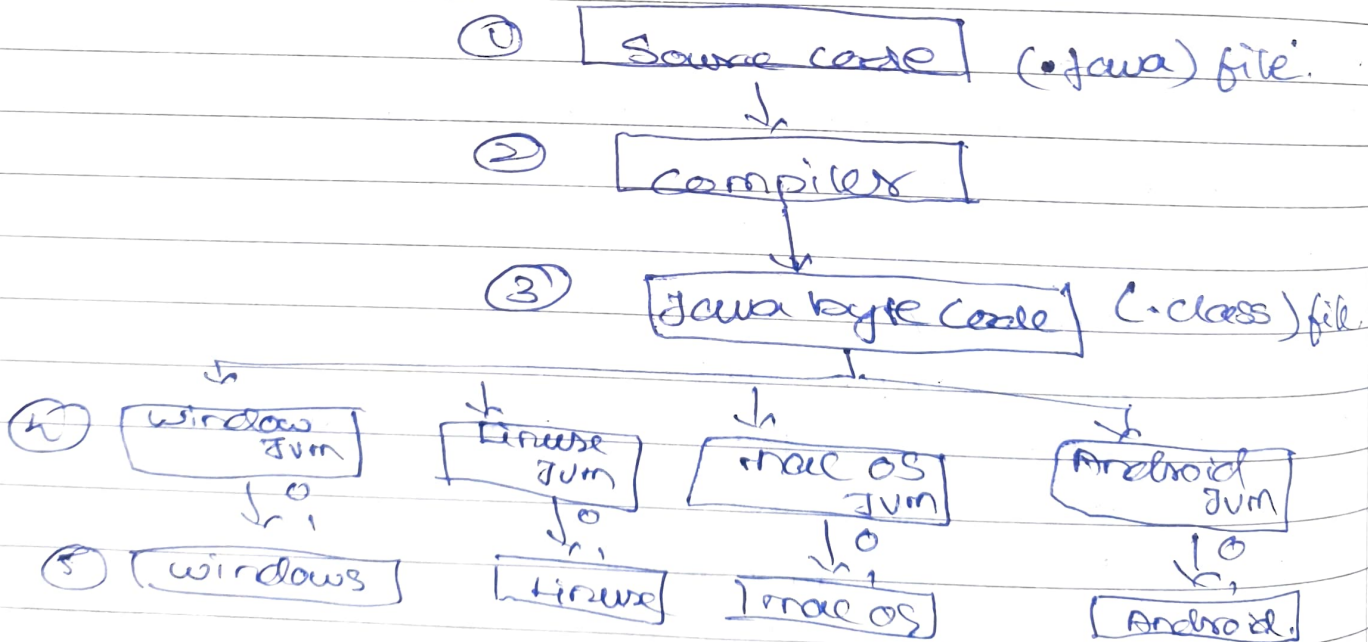


③ Java Byte code is platform Independent

Any device capable of running java, will be able to understand & translate java byte code.

④ Java virtual machine runs the Java byte code.

Java platform independent:-



JVM: (Java Virtual Machine) :-

- ~~Rea~~ Runs Java byte code.
- For different systems (linux, windows, macs).
- class loader + run time + Execution Engine.

JRE: (Java Runtime Environment) :-

- Provides libraries, JVM & other components.
- 2 key tech:-

1) Java plugin:

2) Java web start:

- JRE doesn't contain tools like compiler & debuggers.

JDK: (Java Development Kit) :-

- Support of JRE.
- contains JRE + compilers + debugger.

Java oops concepts:-

- 1) Object :- real entity like human, dog, etc..
 - State & behavior.
- 2) Class :- "Collection of objects"
 - blueprints.
- 3) Inheritance - object inherits properties from other objects.
- 4) polymorphism - having multiple forms.
- 5) Encapsulation - Binding together.
- 6) Abstraction - hiding internal details.

1) hello.java

① oops concept.

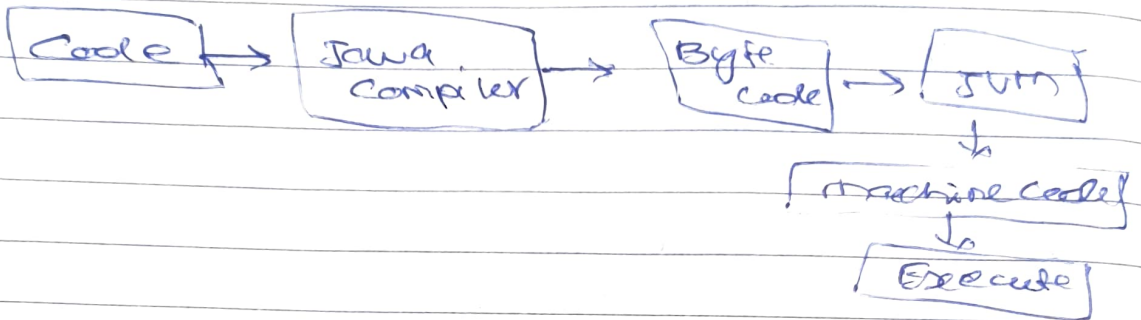
② main function.

class hellof

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

3.

- javac hello.java.
- java hello.java.



Class & Object.

Class: blueprint that describes the state & behavior of an object.

Object: It has state & behavior

Ex: human has states like height, weight, skin tone, ...

walking, eating, sleeping, working.

object:-

Human rakosh = new Human();

Human Sadhish = new Human();


```

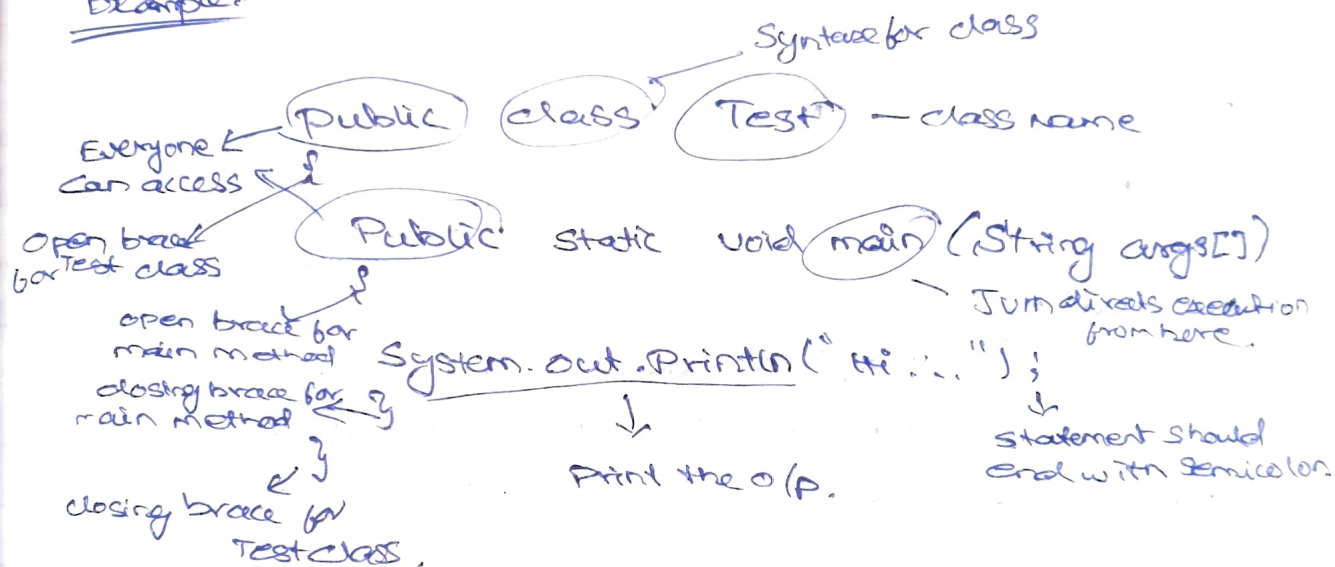
public class Human
{
    float height;
    float weight;
    void getWeight() {
    }
    void get height() {
    }
}

```

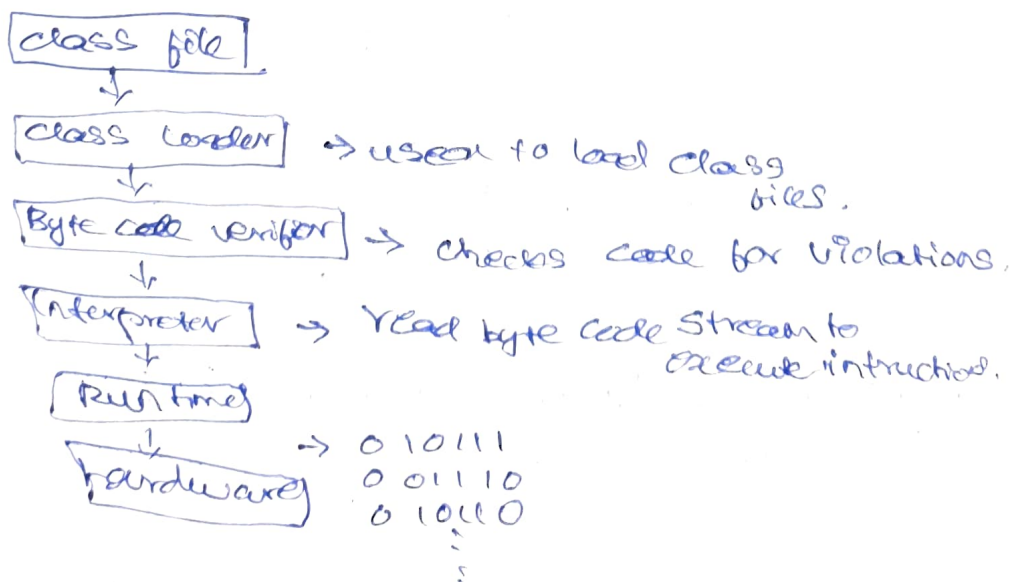
Anatomy of a JAVA class:-

- when JVM runs it looks for main() method.
- It executes everything inside the main method.

Example:-



Runtime operations:-



Java Naming Conventions:-

- Rules to name variables, class & etc.
- Not mandatory rule but it's recommended.

Conventions:-

- ① Class name: Start with uppercase letter &
Ex: Human be a noun (Camelcase)
- ② Interface: Start with uppercase letter &
Ex: Runnable be an adjective (Camelcase).
- ③ Method: Start with ^{lowercase} ~~uppercase~~ letter &
Ex: print(). be a verb (Camel case).
- ④ variable & package: Start with lowercase variable
- ⑤ Constants: All characters in uppercase.
Ex: STATUS

PSVM (Tab) shortcut

Variables:-

- Helps to identify the things.
- variable is a name which is associated with value that can be changed.

How to declare a variable :-

`datatype VARname = value;`

Ex:-

```
m1 : char letter = 'a';  
m2 : int deposit;  
      deposit = 1000;
```

→ Variable are containers that will help you to store the data.

Variable Naming Conventions:-

① cannot contain white space. ("X")

Ex: `int num ber = 100;`

② variable can't begin with special character.

Ex: `int $deposit = 1000;`

("X")

③ variable are case sensitive.

Ex: `rank` \neq `Rank`

("X")

④ Recommended value.

→ variable name should begin with lowercase.

Ex: `int dep = 1000;`

Memory Creations:-

① Byte → 8 bits.


② short → 16 bits.

③ int → 32 bits.

④ long → 64 bits.

primitive data types:-

`byte a = 7;`

 00000111

a binary representation of 7 = 00000111

Reference variable:-

`Account rankAcc = new Account();`

Step 1: Declaring a reference variable.

+ `rankAcc` is reference variable.

+ JVM Allocation space for ref variable.

Step 2: create an object

+ JVM allocates space for new account object in heap.

Step 3: Link object reference & object.



① class hello

```
public static void main (String args[]) {  
    int Container = 67;           variable 1  
    int Container2 = (John);      variable 2  
    String John = "J";           variable 3  
}
```

```
    System.out.print (Container);  
    System.out.print (Container2);  
}
```

② Addition of two numbers:-

class hello

```
public static void main (String args[]) {  
    int a = 10;  
    int b = 20;  
    System.out.print (a+b);  
    System.out.print ("a+b");  
}
```