

Part 1

Step-1

- 1) Programming language and types
- 2) History of Java
- 3) Edition of Java
- 4) Feature of Java
- 5) Java is Platform ~~isolate~~ independent
- 6) JDK, JRE, JVM
- 7) Step of Java Program

1. Java Program2. Structure of Java Program3. Printing statement4. token and types5. Data type and variable6. Primitive typecasting7. Control statementStep-2 Part-21) Method2) Static and non-static member3) Object4) class5) Constructors6) OOPS Principle7) Encapsulation8) Inheritance9) Polymorphism10) Abstraction

Part 3

- 1) Packages
- 2) Access modifiers
- 3) Array
- 4) String
- 5) Wrapper class
- 6) Collection Framework
- 7) Exception Handling

Programming language

- Medium used to give instruction
- Medium used to give instructions to machine to perform specific tasks

Types of Programming Language

⇒ Based on level of Programming Language, there are three types:-

1. Low level Programming Language
2. Mid level Programming Language
3. High level Programming Language

1. Low level Programming Language

→ The Language which is easily readable, understandable and executable by machine directly is known as low level language

- > It is also known as machine level language or machine language
- > Ex:- Binary language (0,1)

2. Mid level language

- > It consists of some Pre-defined word for specific tasks, such as :- Add for addition
Sub for subtraction
Mul for Multiplication
Div for Division etc.
- > These Pre-defined words are known as mnemonics.
- > Predefined words can't be directly understandable by machine.
- > To make machine understand, we are using one translator called assembler.
- > Ex:- Mainly used in Microcontroller and Microprocessors
~~Microprocessor~~ Microcontroller such as 8085 and 8086

3. High level Programming language

- > The language which is easily readable, understandable and executable by programmers directly is known as high level Programming language.
Eg:- C, JAVA etc.
- > High level language are not directly understood by machine.
- > To make machine understand, we are using translators such

as compiler or interpreter

- Every high level language having their own compiler or interpreter.

History of Java

- It is developed by James Gosling in the year of 1995 at Sun Microsystems.
- Currently, it is taken care by Oracle, since 2010.
- First name of Java was OAK.
- OAK changed into Java due to copyright issue, where Java was famous coffee brand which leads the Java logo's symbol coffee cup.
- The Father of Java is James Gosling.
- The current version of Java is Java 24, with respect to March 2025.
- I am having a knowledge of Java 1.8 or 8 with respect to industry requirements.



Versions of Java

Java 1.0	1.1	→ 11	1.21	→ 21
1.1	1.12	→ 12	1.22	→ 22
1.2	1.13	→ 13	1.23	→ 23
1.3	1.14	→ 14	1.24	→ 24
1.4	1.15	→ 15		
1.5	1.16	→ 16		
1.6	1.17	→ 17		
1.7 → 7	1.18	→ 18		
1.8 → 8	1.19	→ 19		
1.9 → 9	1.20	→ 20		
1.10 → 10				

Feature of Java

-> Java having Unique Feature! -

1> Platform Independant

2> Portable

3> Robust

4> Interpreted

5> Architectural Neutral

6> Dynamic

7> Distributed

8> High Performance

9> oops

10> Simple

11> Secure

12> Multithreaded etc



Platform Independent

Platform

- It is combination of OS and Processor.
- Ex: Linux, Mac, windows etc.

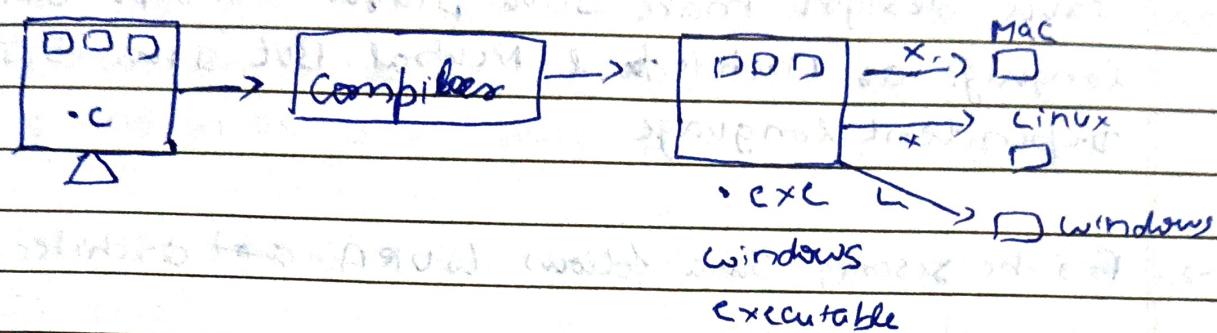
Platform dependant

- If one application develop in one platform, after that it can run only some kind of platform, it is known as Platform dependant.
- Ex: C is platform dependant language.

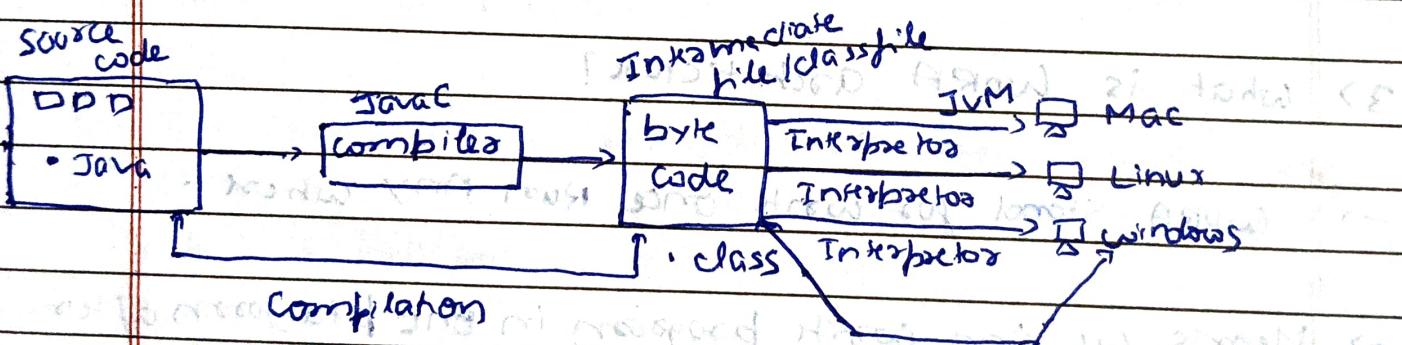
Platform ~~Independent~~ Independent

- If one application develop in one platform, after that it can run on any kind of platform is known as Platform Independent.
- Java is a Platform Independent language.

C is Platform Dependent



Java is Platform independent



Q) Why Java is Platform Independent?

- Java compiler does not directly convert the high level language to machine understandable language. It converts the high level language to one intermediate language which contains "Byte code". This file is known as class file with .class extension.
- Once class file is ready, it can be executable on any machine.

Which has JVM (Java Virtual Machine) in it.

- Java design make Java platform independent language or Architectural Neutral. But Java is JVM dependent language.
- For the system, Java follows WORA ~~as~~ Architecture

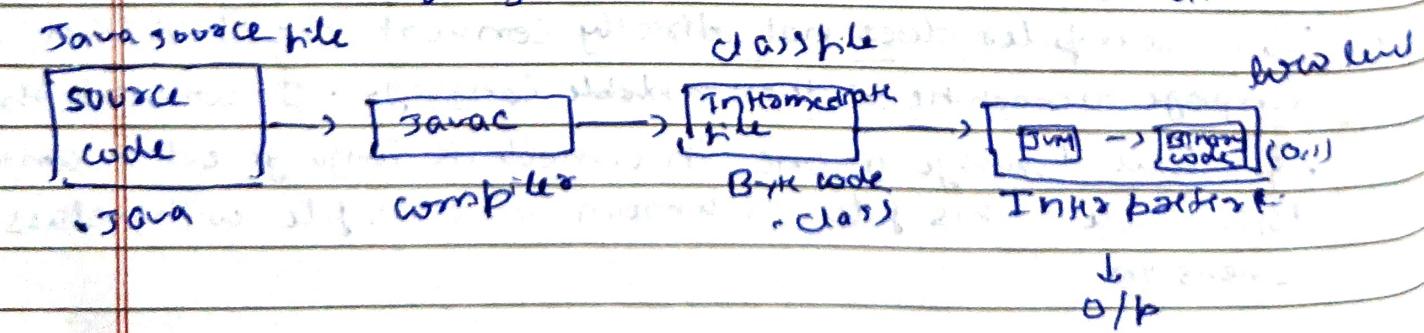
2) What does make Java Platform Independent?

- Intermediate Code which is nothing but byte code and JVM (Java Virtual Machine) make Java Platform Independent.

3) What is WORA Architecture?

- WORA Stand for write once Run Anywhere.
- Means we can write program in one platform after compilation of the Program, we can execute it on any of the Platform which has JVM in it.

* Conversion of High level to low level in Java.



What is Java source file?

- The file generated by the programmer that consists instruction is written in Java programming language is known as Java source file.
- The Java source file should always have the extension as .java

What is class file?

- The file generated by the Java compiler is known as class file which consists of instructions in byte code or intermediate code.
- The extension of class file is always .class.

What is byte code?

- The intermediate code generate by the compiler after the successful compilation which is neither understandable by the programmer nor by the machine is known as intermediate code or byte code.

1) What is the name of Java compiler?

→ JavaC

2) What is the Java interpreter?

→ JVM

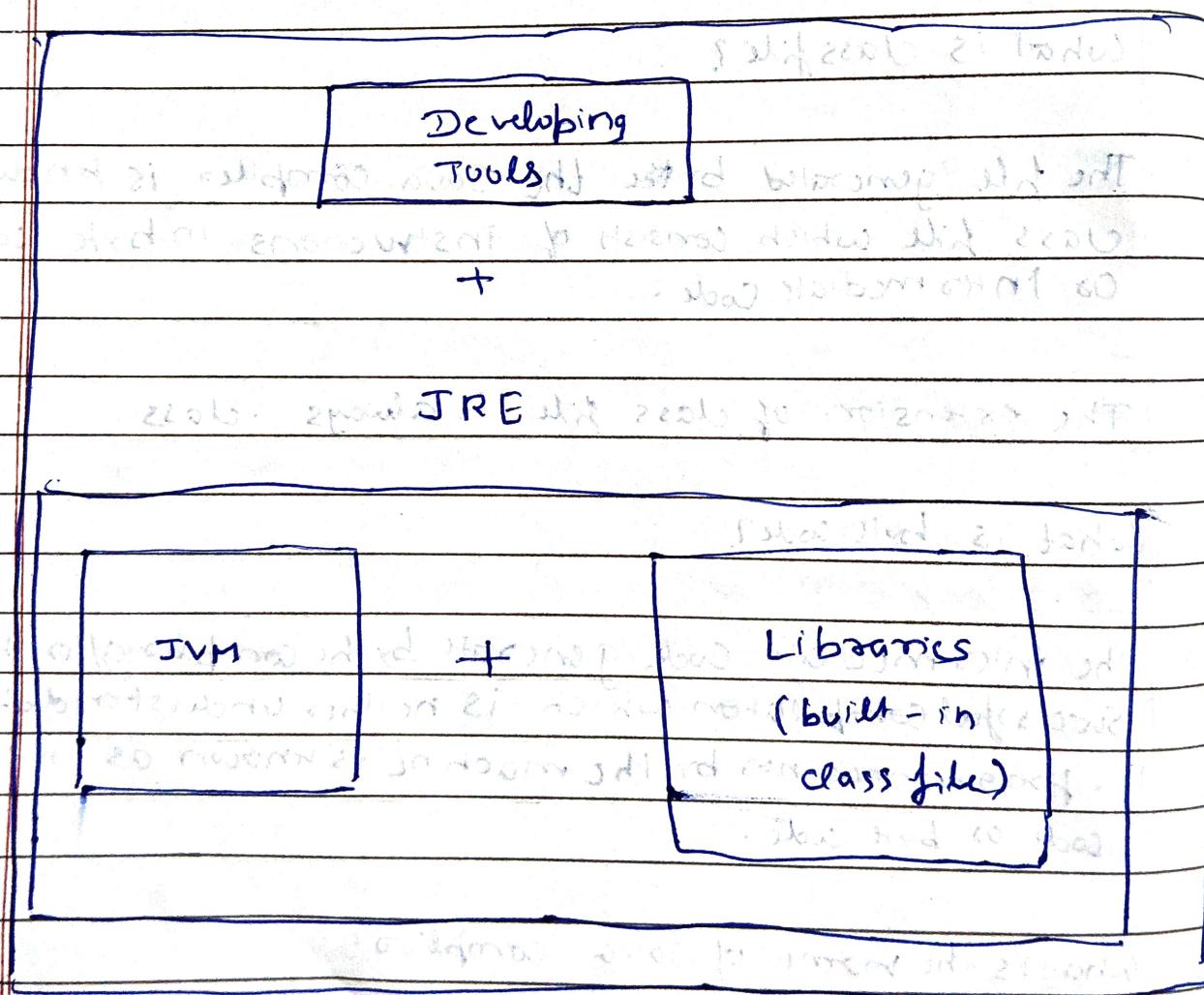
3 What is the extension file of Java source file?

→ .Java

4. What is the extension of class file or intermediate code?

→ .class (dot class)

Overall Diagram of JDK



Development tools

Javac
compiler

JRE

in built
class file

JVM

JDK

- JDK stand for Java development ~~tools~~ kit which consist of development tools such as Java Compiler and JRE for execution.

JRE

- JRE stand for Java Runtime Environment. It is used to provide runtime environment memory for the execution of the Program and its consist of JVM and Java library.

JVM

- JVM stand for Java Virtual Machine.
- It use to convert Byte code into low level language or

machine language by help of interpreter and JIT compiler.

Interpreter

- The nature of interpreter is converting the code line by line

JIT Compiler

- It stands for just in time compiler.
- It used to enhance the code.

Java Library

- It is a package or folder which consists of many more sub folder or sub packages.
- These Predefine class files mainly consist of variable, methods, classes etc.

⇒ Step to go for Java program

1) Create a Java Program

2) Compile the Program

3) Execute the Program

⇒ Type of Editors

1) BASIC

2) IDE (Integrated Development Environment)

1) Basic

- Notepad
- Editplus
- Sublime Text

2) IDE

- VS Code
- IntelliJ
- Eclipse

Compile

→ Java C

→ JAVAC FileName.java

Execute

→ Java

→ Java className

- ⇒ Java instructions are always written inside the class.

Syntax

class
block

```
class class name -> tell the system what  
{  
    // Stmt,  
}  
} written in lower case
```

- ⇒ Every class having own name is known class name.

- ⇒ Every class having block is known as class block.

Note

- ⇒ We should save the file with the name of class name.

- ⇒ Syntax to use to compile the program is
javac filename.java.

- ⇒ Syntax to use to execute the program is
java className.

Note :-

- A class in Java can be executed only if the main method is created as follows:
- Syntax to create the main method:-

Public static void main (String [] args)

{
 // Statement;
}

- We can create a class without main method. It is compiled successfully and the class file is generated, but we can't execute that class.

Q1) Can we execute a Java program without main method? Justify

- No

Justification

- We can create a class without main method. It is compiled successfully and the class file is generated, but we can't execute that class.

- Because in Java, the program execution starts from a main method. As main method is not there we can't execute the program.

In Java mainly two types of printing statement
are there :-

i) Print statement

ii) println statement

i) Print statement

→ Print Statement is used to print the data.

→ We can't use the Print Statement without passing any data, if we use then we will get a compile time error.

→ Ex- System.out.print ("Hi");
System.out.print (s);

ii) println statement

→ println Statement is used to print data as well as create a new line.

→ We can use the println Statement without passing any data, it is just for printing new line.

→ Ex- System.out.println ("Hi")