**Day-03: 22-01-2025**

**Introduction to Java**

**====================**

-> High level programming language

because like other languages java also can support the five types of statements and five types of tokens to develop any applications

-> Platform independent language

Ex: X-app ==> Windows ==> ARM Processor /Intel Processor

MAC-OS

Why only java to learn?

===============

-> With java we have more opportunities.

Java Applications:

==================

Java Platform has three types of editions:

1) JAVA SE (Java Standard Edition)

=======================

--> Using Java SE, we can develop "standard applications".

--> The apps which can be installed into a device and controlled or used with in the same device, are called as "standard applications".

Ex: Mobile: Camera, Dialer, Gallery

Desktop: Notepad, MS Excel, Paint etc.

2) JAVA EE (Java Enterprise Edition)

=======================

--> to develop web applications based on the organization requirement, we have a platform in java called "Java EE".

Ex: Flipkart, IRCTC etc.

3) JAVA ME (Java Micro Edition)

====================

--> This can be used to develop gaming applications and distributed applications.

WhatsApp ==> Distributed application

at a time it can interact with local machine for resources and also with cloud for other resources

Phonepe ===> ICICI user

Check Balance

Phonepe ==============> Bank Server

====================================================

Java Version History

=============

JDK 1.0 ===> 1996

JDK 1.1 ===> 1997

J2SE 1.2 ==> 1998

J2SE 1.3 ==> 2000

J2SE 1.4 ==> 2002

J2SE 5.0 ==> 2004

Java SE 6.0 ===> 2006

Java SE 7.0 ===> 2011

Java SE 8.0 ===> 2014 (Standard Version)

Java SE 11.0 ==> 2018 (Standard Version)

Java SE 17.0 ==> 2021 (Standard Version)

Current Version ==> Java SE 24 ==> 2024

--> Java programming language written by "James Gosling" in 1996 when he was working at Sun micro System.

--> Now, Sun Micro System was adopted by Oracle Corporation from 2010, so now Java is under the control and maintenance of Oracle.

Java Software Installation:

=================

1) use this link to get the software of java to download

https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html

2) Based on the your device configuration:

we should start download

Ex: For windows OS:

Windows x64 MSI Installer ==> click on the corresponding link

then file get start to download

3) After the downloading:

Go to download location/folder

double click on that downloaded file

by following the next button the Java Software can be get installed.

Q: How we can conform whether the Java Software has installed or not?

========================================

Search bar in computer ==> CMD ==> open the command prompt ==> type: "java --version" and press enter

========================================

**Day-04:**

**23-01-2025**

=======================

**Java Program Structure:**

=======================

Java program can be classified into two types:

1) Static program

2) Dynamic program

-> Static program can provide the same output by processing the same input by doing the again and again of execution.

-> Dynamic program can process the different input and provide the different output in every execution time.

-> The java program structure consisting of multiple sections according to the java standards:

Documentation Section

===============

-> Not optional

-> not mandatory

-> but it is suggestable

-> Because of the documentation, anybody can understand our development easily.

-> With comments

can use in two ways:

1) Single line comments ==> //

2) Multi line comments ==> /\* multi line comment \*/

-> documentation is allowed in any where of the program.

-> documentation is not executable part of the program.

-> the documentation can increase the readability of the program.

package section;

===========

-> Optional

-> When we want to integrate multiple modules into single unit we can use "packages".

Syntax:

package packagename;

import section;

==========

-> Optional

-> When we want to use the functionality of one module into another module, instead to re-writing of the functionality we can import that module into required module. Here we can use "importing section".

Syntax:

import moduleName;

class definition section

==============

-> Java programming language is an "Object Oriented Programming Language".

-> High level programming languages:

classified into three types:

1) Procedural programming languages/Functional Programming languages

=====================================

-> when we can build/develop any application with built-in functions those are called as "procedural programming languages".

Ex: C, Python etc.

2) Object based programming languages

=====================================

-> The languages which can allow to use only class and object from OOPs called as "Object based programming languages".

Ex: VB Script

3) Object Oriented Programming languages

=====================================

-> The languages which can implement all OOPs principles like: Encapsulation, inheritance, Abstraction and polymorphism are called "Object Oriented Programming languages"

Ex: Python, C++, Java etc.

OOPs ==> Object Oriented Programming System

concepts:

class, object, method, constructor, destructor etc...

principles:

Encapsulation, Inheritance, Polymorphism and Abstraction

-> Java programming language is one of the strongly object oriented programming language so, we can write any amount of code with the class only.

So, class definition in java program is mandatory.

Syntax:

class ClassName{

class definition

}

main method

========

-> At the time of installation of the java software, there is a module has installed automatically into our computer named as "JVM (Java Virtual Machine)".

-> JVM can search about main() in every program before going to start the execution.

If the main() is not available, JVM can throw an error.

Syntax:

public static void main(String[] args)

{

// method definition

}

Example for static program:

==================

/\*

This is my first program.

And I want to display Welcome message

along with wish/greetings \*/

class MyFirstProgram{

public static void main(String[] args){

// I want to say greetings

System.out.println("Hello");

System.out.println("Welcome To Ashok IT for Java Fullstack Program.");

}

}

**Day-04**

**24-01-2025**

**Where we should write Java Programs?**

=========================

-> To write any java program:

we have two options:

1) Text Editors: Notepad, Notepad++

2) IDE (Integrated Development Environment) : Eclipse, IntelliJ, VS Code etc.

**Text Editors:**

=========

1) We need to create a folder for your practice with Java

2) Create a program in notepad file

3) Save that file into the created folder with ".java" extension.

Syntax:

File-name.java

4) Open the command prompt

5) Navigate to the location that where my java program is.

cd file-path (exclude the .java)

6) We should compile the file:

javac file-name.java

because of the javac command the source code (.java file) can convert into byte code can be store internally as .class file.

7) we can convert that class code (byte code) into machine code (object code) using:

java file-name.java

// Writing a static program in Notepad

/\* Write a program using java to find the sum of three numbers.

and print that result.

\*/

class SumOfThreeNumbers{

public static void main(String[] args){

// data declaration

int a,b,c;

// data assignment

a = 100;

b = 200;

c = 300;

int d = a + b + c;

System.out.println("The Sum of Three numbers = "+d);

}

}

**-> The Java is strongly typed programming language**

=====================================

High level programming languages are classified into:

three types:

1) Statically Typed

=============

-> the languages which can convert the assigned value based on the type of the data (which mentioned at the time of declaration of data),such type of languages are called as "Statically typed languages".

Ex: C

#include<stdio.h>

void main(){

int a; // data declaration

a = 123; // data assignment

printf("%d\n",a);

a = 1.00023f; // float

printf("%f",a);

}

2) Dynamically Typed

==============

The Languages which can understand the type of the data based on the assignment and it never be declare the data with its type such type of languages are called "Dynamically typed languages".

Ex: Python

a = 123 # data assignment

print(a)

print(type(a))

a = 1.00023

print(a)

print(type(a))

3) Strongly Typed

============

class Example{

public static void main(String[] args){

int a; // data declaration

a = 123; // assignment

System.out.println(a);

a = 1.00023f; // re-assignment

System.out.println(a);

}

}

Output:

=======

Example.java:8: error: incompatible types: possible lossy conversion from float to int

a = 1.00023f; // re-assignment

^

1 error

error: compilation failed

========================================

Q: What happened when we cannot define the main() method in java program?

========================================

At the time of compilation, there is no error,

but the JVM can throw the error as "can't find main() method in the class : className"

-> In java, there are two types of errors:

1) Compile time error ==> can throw by the compiler

2) Run time error ==> can throw by the JVM