Name of the Module : JSE (Java Standard Edition)

Duration /Session : 101 Session (4 Months) [80% Core Java -> ADV Java]

Sunday classes : Online Sunday classes [9 -10 sessions] [Helping Topics OR Common topic]

Java Version : Upto Java 17V

Lab Assignment : I will provide all the Assignments [MCQs + Sample coding + Scenario based]

Validity Of Task : 1 day [11AM]

Lab is compulsory : Without Lab, Permission is not for Core Java classes

After course : 3+ years developer knowledge

Notes : Google Class Room

What is a Language ?

* A language is a **communication media** through which we can communicate with each other.

What is a Programming Language ?

* A Programming language is an intermediate between the user and computer system. By using programming languages we can submit the instructions to get the desired output.

Characteristics of the programming language :

* We have 2 characteristics :

a) Syntax [Rules given by that particular language]
b) Semantics [Meaning OR Structure given by that particular language]

Example :

English Translation :

Syntax [Subject + Verb + Object]

He is a boy. //Valid
He is a box. //Invalid

* In our Java programming Language we have 2 levels of security :

1) At compiler level :

Here our Java compiler will only **verify** whether the programmer is following the syntax or not ? Compiler will never take care regarding the execution of the program. The compiler will generate syntax error if something is wrong syntax wise.

2) At execution level :

Here Our runtime environment will verify whether the code is meaningful OR meaning less, If the code is meaning less then it will generate a Runtime error.

So, the final conclusion is : Java compiler will verify the syntax and our runtime environment will verify the semantics [meaning OR structure of the code]

What is Java ?

* Java is a high level, Platform independent, secure, robust (strong), multithreaded and object oriented programming language.

* Java was originally developed by sun microsystem in 1995 but the first version of java i.e JDK (Java Development KIT) 1.0 was released on 23rd January 1996.

* In the year 2010, Java was acquired by Oracle Corporation.

* By using java language, We can develop various applications like

a) Stand-alone application [Desktop OR Software]
b) Web application [websites]
c) Enterprise application
d) Gaming application [Multithreading]
e) Mobile application
f) Distributed application

Flavors of Java language :

The following are the flavors of java language.

1) SSE (Java Standard Edition) -> Core Java [Stand-alone applications]

2) EEE (Java Enterprise Edition) -> Advanced Java [Web application OR Websites]

3) JME (Java Micro/Mobile Edition) -> Android Java [Mobile Applications]

4) JAVA FX (Out dated, By Using Java FX we can develop GUI applications like button, radio button and so on).

History of Java :

Name of the Language : Java
First Name of Java : OAK (Tree Name) 1991
Release Date : 23rd JAN 1996
Developed By : James Gosling and his friends
Project Name : Green Project
Place : Island, Indonesia
Official Symbol : Coffee CUP.

They asked a question to James Gosling ?

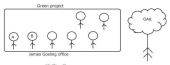
Can you create a common remote control ?

James Gosling started the project.

C and C++

Failed [Platform Dependent]

Green project



James Gosling office

*? (Star ?)

They sent this language -> Lab [Registration [OAK]]

Rename of OAK

Indonesia, Java island

Java

Coffee CUP

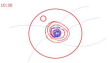
Virtual Machine

What is a function ?

() -> Symbol of the function

RPM

10:38



Approach 1

```
void main()
{
  a + b;
  a - b;
  a * b;
  a /b;
}
```

Approach 2 :

1) Modularity

2) Easy understanding

3) Reusability

4) Debugging is easy

Approach 2

```
void main()
{
}
void sum(int x, int y)
{
}
void sub(int x, int y)
{
}
void mul(int x, int y)
{
}
void div(int x, int y)
{
}
```

Function defination :

A function is a self defined block which is used to perform some calculation, printing the data and so on.
A function may OR may not return a value.

Example:
public void sum(int x, int x)
{
}

A function is divided into two types :

1) Predefined OR Built-in function
2) User-defined OR Custom function

Predefined OR Built-in function

A function which is developed by language creator itself is called predefined function.

User-defined OR Custom Function

A function which is written by user/developer for its own requirement and specification is called User-defined function.

Advantages of Function :

1) Modularity :

Dividing the bigger task into number of smaller task..

2) Easy to understand :

Once the task is divided into number of independent modules then it is easy to understand the entire module.

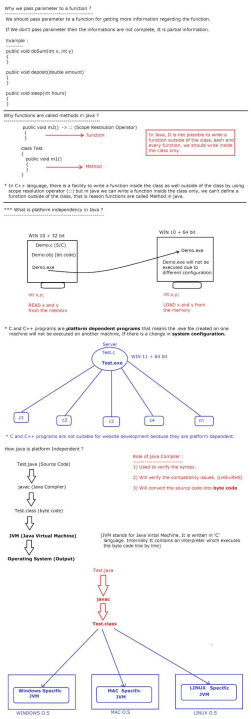
3) Reusability :

We can reuse a particular module for 'n' number of times.

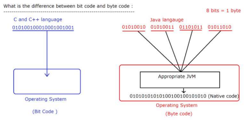
Note: In java, we always reuse our java classes.

4) Easy debugging :

Each module is isolated from another module so the debugging is easy because we can debug only one module where we have syntax or semantics error.

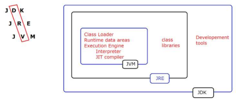

Why we pass parameter to a function ?
We should pass parameter to a function for getting more information regarding the function.
If We don't pass parameter then the informations are not complete, it is partial information.
Example :
public void doSum(int x, int y)
{
}
public void deposit(double amount)
{
}
public void sleep(int hours)
{
}
Why functions are called methods in Java ?
public void m2() -> :: (Scope Resolution Operator)
{
}
function
class Test
{
public void m1()
{
}
Method
}
In Java, It is not possible to write a function outside of the class, each and every function, we should write inside the class only.
* In C++ language, there is a facility to write a function inside the class as well outside of the class by using scope resolution operator (::) but in Java we can write a function inside the class only, we can't define a function outside of the class, that is reason functions are called Method in Java.
*** What is platform independency in Java ?
WIN 10 + 32 bit
Demo.c (S/C)
Demo.obj (M/c code)
Demo.exe
WIN 10 + 64 bit
Demo.exe
Demo.exe will not be executed due to different configuration
int x,y;
READ x and y from the memory
int x,y;
LOAD x and y from the memory
* C and C++ programs are platform dependent programs that means the .exe file created on one machine will not be executed on another machine, if there is a change in system configuration.
Server
Test.c
Test.exe
WIN 11 + 64 bit
c1
c2
c3
c4
c5
* C and C++ programs are not suitable for website development because they are platform dependent.
How java is platform Independent ?
Test.java (Source Code)
javac (Java Compiler)
Test.class (byte code)
JVM (Java Virtual Machine)
Operating System (Output)
Role of Java Compiler :
1) Used to verify the syntax.
2) Will verify the compatibility issues. (x86-x64)
3) Will convert the source code into byte code
[JVM stands for Java Virtal Machine, It is written in 'C' language. Internally it contains an interpreter which executes the byte code line by line]
Test.java
javac
Test.class
Windows Specific JVM
WINDOWS O.S
MAC Specific JVM
MAC O.S
LINUX Specific JVM
LINUX O.S
Java is a platform independent language. Whenever we write a java program, the extension of java program must be .java.
Now this .java file we submit to java compiler (javac) for compilation process. After successful compilation the compiler will generate a very special byte code file i.e .class file (also known as bytecode). Now this .class file we submit to JVM for execution purpose.
The role of JVM is to load and execute the .class file. Here JVM plays a major role because it converts the .class file into appropriate machine code instruction (Operating System format) so java becomes platform independent language and it is highly suitable for website development.
Note :- Each OS (Windows, Linux, Mac) has its own JVM implementation that means JVM is platform dependent technology where as Java is platform independent technology.
JVM internally contains an interpreter so it executes the code line by line. It is written in 'C' language hence platform dependent.
Note :- All the browsers internally contain JVM are known as JDE (Java Enabled Browsers) browser.

What is the difference between bit code and byte code :



* Bit code is directly understood by Operating System but on the other hand byte code is understood by JVM, JVM is going to convert this byte code into appropriate machine understandable format.

*** What is the difference between JDK, JRE, JVM and JIT compiler?



JDK = JRE + Development tools

JRE = JVM + Class Libraries

JVM = Class Loader + Runtime Data areas + Execution Engine
Interpreter
JIT compiler

JDK :

* It stands for Java Development kit.
* It is a **developer version** that means by using JDK we can develop as well as execute the java programs.
* In order to develop and execute the java programs, we have various development tools :

a) javac : java compiler, responsible for compilation.
b) java : Java launcher,responsible for executing java program.
c) jdb : java debugger, for debugging purpose
d) jconsole : java Console, to display the output in the console.
e) javap : java Profiler, To get the details of a class
f) javadoc : Java documention, for Generating java documentation.

All the predefined .class file is available in a jar file called jrt-fs.jar (lib folder)

JRE :

It stands for Java Runtime Environment.It contains JVM and class libraries.

It is a client version that means by using JRE we can only execute our java programs (We can't develop).

From java 11 version JRE folder is removed so now, from java 11v we can execute a java program without compilation by using following commnd :

java FileName.java [This is the command to directly execute our java code]

```
//Program :

public class Demo
{
    public static void main(String[] args)
    {
        System.out.println("Hello Welcome to Java!");
    }
}
```

java Demo.java [We can directly execute the program]

JVM :

* It stands for Java Virtual Machine.
* The main purpose of JVM to load and execute the .class file.
* JVM takes .class file and convert the .class file into Operating System understandable format.

* JVM is platform dependent, It provides various features like Class loading, class verification, allocating the memory for all the data members, Garbage collection, Security Manager and so on.

What is the difference between Compiler (javac) and Interpreter (JVM contains Interpreter) :

Compiler	Interpreter
1) Scans the entire program once.	1) Scans line by line.
2) Will convert the S/C into byte code.	2) Will convert the byte code into O.S format.
3) byte code is generated so, It needs a separate memory.	3) Read the byte code line by line and generate the output concurrently.
4) It displays all the errors and warnings at a time so, debugging is slow	4) It will display only one runtime error at a time so, debugging is fast
5) After successful compilation, execution is fast.	5) Execution is slow because If we make a mistake at line number 5 then after resolving the issue It will again start from line number 1
6) After successful compilation, We can delete the source code.	6) We cannot delete the byte code due to line by line execution

Note : Java uses compiler and Interpreter.

** JIT Compiler : [improve the performnace of java program at runtime]

JIT (Just-In-Time) Compiler is a part of the JVM execution engine.

It is used to increase the performance of Java programs.

Internally JVM uses an interpreter, which executes bytecode line by line hence execution slow.

To overcome this limitation, JVM uses JIT Compiler.

How JIT Works:

When the JVM finds some methods or instructions that are repeated frequently (called hotspots), Instead of interpreting them line by line again and again, JIT compiler compiles those bytecode instructions into native machine code.

At the time of execution, JIT compiler directly provides these native machine code to JVM for fast execution.

What is a Variable ?

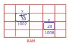
* A variable is a **name given for the memory location**.

Example :

```
int x = 10;

int y = 20;

x = x + y;
```



* It can hold some meaningful value within the range.

Variable

Vary + able

Change + able

* A variable can change it's value during the execution of the program.

Drawback of an Ordinary Variable ?

```
int x = 12, 90; //Invalid
```

* An ordinary variable can hold **only one value at a time** and It stored in the memory in random memory location so to avoid this we introduced next level of concept i.e Array.

* Another drawback is , It is volatile that means once the execution of the program is completed the variable value will be deleted from the memory, In order to hold it permenently we should use File Handling.

What is data type in java ?

It defines the type of data that a variable can store.

It also describes :

a) Size : How much memory it occupies in RAM.

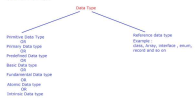
b) Range : What values it can hold (e.g., byte → -128 to 127).

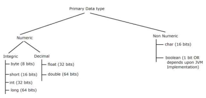
c) Operations : What kind of operations can be performed (e.g., addition for numbers, Concatenation of String).

How many data types in java ?

* Java data types are mainly divided into two categories:

1) Primitive Data type
2) Reference Data type





Note : WE HAVE TOTAL 8 PRIMITIVE DATA TYPES IN JAVA , AMONG THESE 8 DATA TYPES 6 DATA TYPES ARE REPRESENTING NUMBER i.e byte, short, int, long, float, double.

What is the difference between Statically typed langauge and Dynamically typed language ?

Statically typed language :

The languages where data type is compulsory before initialization of a variable are called Statically typed language.

In these languages, once we define the type of the variable then it will hold same kind of value till the end of the program.

Example of statically typed languages :
C,C++, JSE, C# and so on

Dynamically typed language :

The languages where Data type is not compulsory, It is optional to initialize the variable are called Dynamically Typed Language.

In these languages we can provide all different kinds of values to the variable during the execution of the program.

Example of dynamically typed languages :
Visual Basic, Javascript, Python and so on

What is comment in java ?

Comments are used to enhance the readability of the code. It is ignored by the Compiler.

In java, We have 3 types of comments

1) Single line Comment
```
//
```

2) Multiline Commment
```
/*


*/
```

3) Documentation comment
```
/**
Name of the Project : Online Shopping
  Number of Modules : 70 Modules
      Project Date : 12th March 2024
  Last Modification : 9th SEP 2025
          Author : James Goling and his friends
*/
```

==