Akash Patil

Page :
Date : / /

Java -

What is Java - Java is a High level object oriented language used for developing java applications in laptops smartphones.
It is a general purpose class and object based.

History of Java - Java initially called as Oak developed by sun microsystem and after that by oracle James Goslin.

Java features - ① Easy to use programming language to learn
② Write once and run it on almost every platform
③ Multithreaded and automatic memory mangement

Components of Java - ① JDK ② JVM ③ JRE

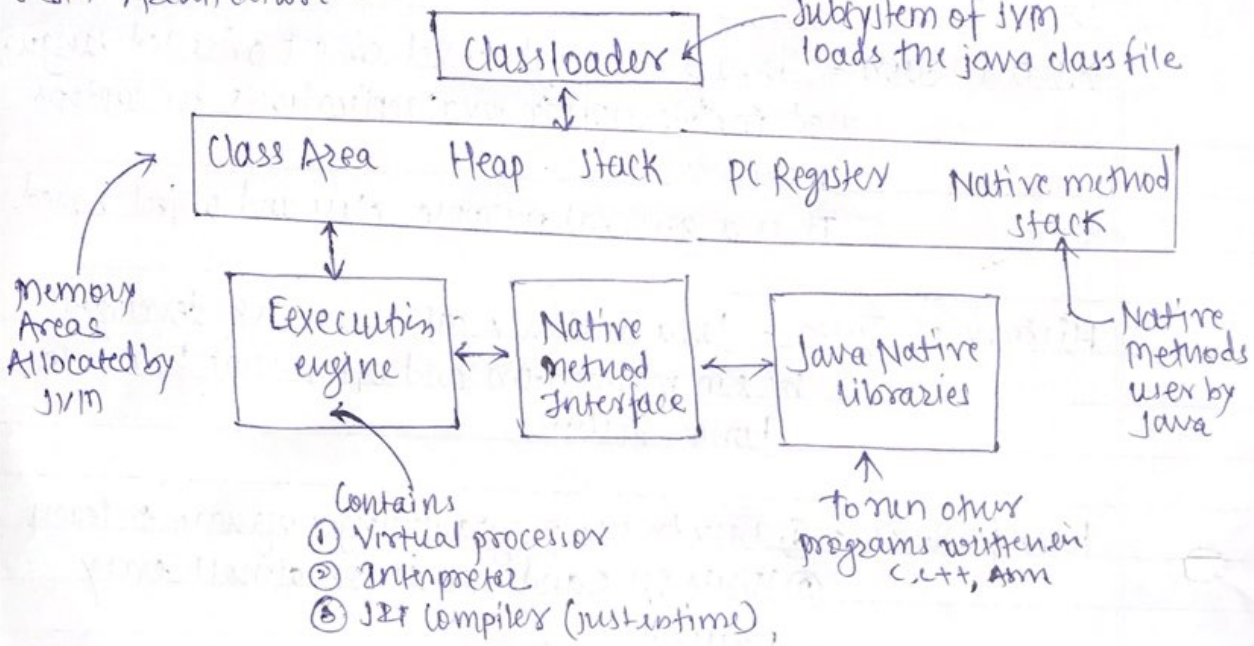
JDK - Java Development Kit
JDK contains all tools required to write and execute the program. it contains compiler

JVM - Java Virtual Machine
- It is a engine that provides environment to drive the java code
- It converts the java bytecode into machine code

JRE - Java runtime environment
- It is a piece of software to run other software
It contains all the class, libraries, JVM and other supporting files.
- It uses important packages like, swing, util, lang, awt and runtime libraries

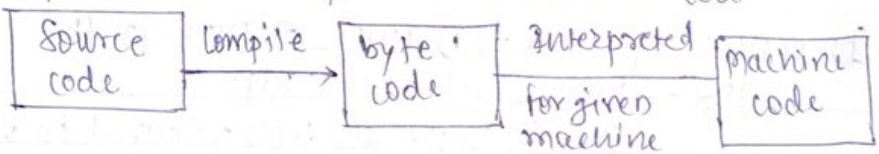
Variables and Datatypes

JVM Architecture -



→ How Java Works -

First Java source code compiled into bytecode and then bytecode is interpreted into machine code



→ Basic Structure of Java program

```
Package <Packagename>
import <Packagename>
public class main {

    public static void main (String... Arg) {
        // block of statements
    }
}
```

Entry point

Variable argument

Naming Conventions →

Class - PascalCase → Employee
method - CamelCase - myMethod()

Akash Patil
Page :
Date : / /

Variables and Datatypes

Variable - It is a container that stores a value and this value can be changed using execution

Datatypes - 8 primitive → ① byte, short, int, long, float, double, char, boolean.

Non primitive - class, array, interface (not defined in java, created by us)

Keywords - Reserved words used by Compiler

Reading data from keyboard -

```
import java.util.Scanner
    Scanner input = new Scanner(System.in)
                                  ↳ Read from keyboard)

int a = input.nextInt();
```

Operators and Expressions -

performs operations on variables

Types -

① Arithmetic operator → +, -, *, /, %, ++, --
② Assignment operator → =, +=, -=, *=, /=
③ Comparision operator → <, >, >=, <=, ==, !=
④ Logical operator → &&, ||, !
⑤ Bitwise operator → &, | (operates on bit)

Increment & Decrement

post and pre ++, --

post a++ performs operation and then increment
++a first increments and then performs operation

Akash Patil

→ Strings - String is a sequence of Characters.
In java, String is a Class / Reference and treated as Datatype

→ Instance of String -
String str = "Anystring";
String str2;
str = new String("Anystring").

→ print on Console
System.out.{print(), println() printf(), format()}
↑ without new line — with new line — ↑ System.out.printf("%c", ch);

Different String methods - String str = new String("DKTE")

① str.length() → gives 5
② str.toLowerCase() → given string in lowercase
③ str.toUpperCase() → gives string in Uppercase
④ str.trim() → Return new string after removing all the spaces at start and end.
⑤ str.substring(start, end) → returns the string from start index to end-1 index
end is excluded
⑥ str.replace('r', 'p'); Return new string by replacing r with p
⑦ str.startsWith("D") ← true
⑧ str.endsWith('s') ← true
⑨ name.charAt(2) ← Returns T
⑩ name.indexof('D') ← 0
⑪ name.lastIndx() ← last indx

Escape Sequence \n, \t, \', \\ backslash
→ Single Quote
→ tab
→ new line

Akash Patil

Conditional in Java -

if, if else, if else if ← Decision making

Switch case -

If we have choices

```
Switch () {
    case< > :
        // Statement
        break;
    defaut :
}
```

Loops - Sometime we want to execute a few set of instruction again and again

Types of loops

while loop	while(condition) { }
do while	do { } while(condition);
for loop	for(iteration, condition) { }
for each loop	for(int X : valArr) { }

break statement -

If we want to stop the loop then we can use break keyword

It complitely exits the loop

Continue -

escape the steps and move to the next step

Akash patil

→ Arrays -

Array is a collection of datatype

use - storing marks of 5 subjects

Declaration - int[] marks = new int[5]
int mark[] = new int[5]
int marks[] = {50, 60, 70, 80, 90}

find length of array
marks.length

Display - using loops

foreach loop → for(int x : marks) { print x }

Multidimential array -
It is Array of Arrays
Declaration int[][] marks = new int[][]
Rows Columns

→ Methods - It is a function in java which is a subprogram executed when we call it in main or another method

→ Method overloading - Method overloading means creating a 2 or more methods with same name but different parameters list

→ Static Keyword - Static variable is associated with a class and not with object or ~~method~~ instance of class. It is initialized only once at start of execution in heap

static blocks to initialize static variable | Static method | Static variable | nested class | class X { static class Y }

Akash Patil

Oops - Object oriented programming tries to map instruction with real world to make code easier and understandable

Object oriented programming -
Trying to solve the problem by creating object

Class - Class is a blueprint of object

Object/Instance - Object is a real world entity or instantiation of a class. A memory is allocated to object in memory only.

Oops - Terminologies

① Abstraction - Hiding the internal details and showing only essential details.

② Encapsulation - Encapsulating the properties and method in single unit
creating a class is a encapsulation

③ Inheritance - Simply means creating something new things from existing thing
In java creating a new class extending existing class
In inheritance child can access all the properties by parent class

④ Polymorphism - polymorphism means many forms
Achieving one task in different way

Runtime
Interface
method
overriding

Compiletime
method
overloading

Akash Patil

Creating a class class X {
properties
methods
constructor
Destructor
}

Access modifier
Access modifiers specifies where the property and method can be accessible (4) types
private ← cannot access outside class
public ← accessible anywhere
Default ← Only accessible within package/ by default default
protected ← Accessible in class and in child class but cannot outside the package

	class	package	outside package by subclass	outside pckg
private	✓	X	X	X
Default	✓	✓	X	X
protected	✓	✓	✓	X
public	✓	✓	✓	✓

Abstraction in java - by Abstract classes and interfaces

Getters and setters -
to set and retrive the data (methods)
Constructor - It is a method with the same name of class without any returntype to initialize the object
Types - Default
parameterized
Overloaded parameterized

Akash Patil

Destructor - Destructor is used for deallocating memory and clear the used resources it is explicitly called by java
to perform some operation through destructor finelize() method is used

Inheritance - Inheritance is used to use the properties of existing class - borrow

Declaration Inheritance
Parent ⇒ class A { }
Child ⇒ class B extends A { }

Constructor and destructor in java Inheritence
Parent constructor
child constructor
child destructor
parent destructor

Constructor Overloading in Inheritance
In Inheritance Defaut constructor is called by child class object. to call parameterized constructor we can use super keyword
super (a,b)

this keyword -
this keyword points to object
this is a reference to current object

Method overriding -
Methods from parent are rewritten/redefined in child is known as Method overriding

Abstract class And Interface

Abstract in english → idea without concrete existence

Abstract method - A method is declared without implementation with abstract keyword

Abstract class - If class contains abstract methods then class must be declared with abstract keyword
A class containing abstract methods called as abstract class
the instance of object cant be instantiated
abstract class only extended
and child class should override all methods else it also must declared as Abstract

Interface - In english Interface means a point where two system will interact
In Java interface is a group of related method with empty bodies.

Inheritance in Interfaces

```
interface X {
    void method1(int x);
}

interface Y extends interface X {
    void method2(int y);
}
```

Akash patil

Packages in Java -

Interpreter and compiler -

Interpreter scans the code and translate into machine code line by line

Compiler scans entire source file and then translate to machine code

Packages -

Packages are used for group similer classes (related)

packeges helps in avoiding name conflicts

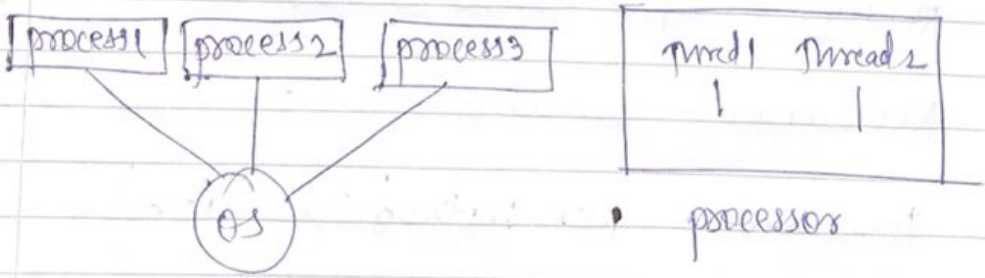
Two types of packages -

1. Built in package
2. User defined package

```
Package packagename      ← Create package
Import packagename       ← use package
```

Multithreading -

multithreading and multithreading and multiprocessing are used to achieve multitasking



(b)

Akash Patil

Thread - Thread is a lightweight context of java program

Avery java program atleast one thread created by JVM at start of program when main method is invoked

Multithreading -

Multithreading is a process running multiple thread simultenously

thread is light weight process

Creating thread

① By extending thread class

② By implementing Runnable interface

life cycle of thread

New → Runnable → Running → Terminated

Nonrunnable

New - ① Instance of thread is created but not started

Start()

Runnable - After invoking of start() before it is selected to be run by scheduler

Running - After thread scheduler has selected it

Nonrunnable - Thread alive but not eligible to run

Terminated - run method has existed

Akash Patil

Methods are in Thread Class -

run();
start();
join();
getpriority();
setpriority();

Errors and Exception -

Three types of error

① Syntax error
② Logical error
③ Runtime error ← Exception Handling will be next part ↓

Java sometimes encounter an error when program is Running, this error is known as Runtime error or exception

types → ① checked → handled by compiler
② Unchecked → Runtime exception

Handle exceptins

Using try catch blocks we can handle exceptin

Userdefined exceptin -

we can create user defined exception by creating class and extending with Exceptin class pass the passage to Exceptin class by super keyword

Throw & throws -

↳ Throw keyword used to throw exception explicitely by programmer

↳ To declare an exception this gives an information that their might be exception.

Akash Pathi

→ Advance java - Collection framework
Java provides classes and interfaces for us to write the code quickly and efficiently.

Arraylist, sets, Stack, Hashmap
← For variable size collection

sets - for distinct collection

Stacks - A lifo data structure

Hashmap - to store key, value pair
Classes are available in java.util package

→ Generics in java -
Generics in java is like templets in C++ it allows to give a type to method while instantiating

```
Class Test<T, F, L> {
        T obj;
        F obj2;
        L obj3;
}
```

Create object → `Test <String, int, float> X = new test <String, t>`

GUI Program :-

Graphical user interface -

It is a type of interface (user interface) through which user can interact with electronic device via visuals.

Java Awt - ~~Application~~ window toolkit

Abstract window toolkit

awt is an API to develope gui or window based application. ↳ Api used for exchange and functionality

Java awt package provides for AWT API such as textfield, Label, Textarea, Radiobutton Checkbox choices etc

awt is platform dependent because it calls native platform subroutine for creating API component.

It displays the component according to os

File handling - Updating and deleting

Creating, reading, writing the data into the file is known as filehandling.

Stream - Bytestream

Characterstream

Bytestream < Inputstream (reads)

outputstream (write)

Characterstream < Reader classes (Reads)

Writer classes (Write