**Day 1: 29 – July 2024 – Core Java**

Java is pure object and platform independent programming language.

Java 1.0 java 22

Till java 11 java is known open source programming language.

From 11 onward if we want to use for commercial purpose we need to purchase or we need to use openjdk.

From java 8 onward java also known as functional programming language because of lambda expression.

OOPs

object: any real world entity.

Property or state-🡪 have -🡪 fields or variable

Person

Behaviour -🡪do/does -🡪 function / methods

Car

Employee

Account

Product

class : blue print of object or template of object or user defined data type which help to create the object

class Test {

variable;

methods;

}

In java variable are divided into 3 types.

1. Instance variable: the variable which declare outside a method but inside a class is known as instance variable. It hold default value.
2. Local variable : variable declared inside a method is known a local. Doesn’t hold default value.
3. Static variable. The variable declared outside a method with static keyword is known as static variable. It hold default value.

Constructor : it is a type of special method which help to create the memory.

Pts

1. Constructor have same name as class itself.
2. No return type.
3. No need to call it will call automatically when we create the object.

Encapsulation : binding or wrapping data and code in a single unit is known as Encapsulation.

Normal class

Java Bean is known as pure encapsulation class.

Inheritance : it is use to inherits the property and behaviour of old class to new class.

class OldClass { super class, base class or parent class.

property

behaviour

}

class NewClass extends OldClass{ sub class, child class or derived class.

property

behaviour

}

Types of inheritance

1. Single inheritance : one super class and one sub class
2. Multilevel inheritance : one super class and n number of sub classes connected one by one
3. Hierarchical inheritance : one super class and n number of sub classes connected one by one.
4. Multiple inheritance : more than one super class and one sub class. Java doesn’t support this type of inheritance using class level. We can achieve this type of inheritance using interface.

OOPs relationship

1. Is a relationship : inheritance Manager is a Employee
2. Has a relationship : inside one class we need to create the object another class.

class Employee {

id,name,salary . Scanner

readEmp

calSalary:

disEmp

}

class Manager extends Employee{

numberOfEmp

Address add = new Address(); Manager has Address

readMgr()

add.readAdd();

disMgr()

add.disAdd();

}

class Developer extends Employee{

projectName

readDev

disDev

}

class ProjectManager extends Manager{

clientId

readPgr()

disPmgr()

}

class Address {

city and state , Scanner

readAdd

disAdd

}

Has a relationship

class A {

B obj = new B();

}

class B {

A obj = new A();

}

Class Employee {

Address add = new Address();

}

Class Address {

City, state

}

Class Student {

StudentHistroy sh = new StudentHistory();

}

Class StudentHistory {

}

Association : if we want to achieve association. Iside one class we need to create another class object ie one or many

Aggregation : type of association. Employee has Address class reference it aggregation. But is it is a weak association.

Composition type of association. Student class has StudentHistory class reference it is strong association.