**Day 2**

**Inheritance:** Inheritance is use to inherits the properties and behaviour of old class to new class.

To achieve inheritance in java we need to use extends.

class OldClass { super class or base class or parent class

property

behaviour

}

class newClass extends OldClass{ sub class or child class or derived class.

property

behaviour

}

Type of inheritance

1. Single inheritance : one super class and one sub class

Class A {}

Class B extends A{}

1. Multilevel inheritance : one super class and n number of sub classes connected one by one

Class A {}

Class B extends A {}

Class C extends B{}

1. Hierarchical inheritance : one super class and n number of sub classes connected directly to super class

Class A {}

Class B extends A{}

Class C extends A{}

1. Multiple inheritance : more than one super class and one sub class

Class A {}

Class B {}

Class C extends A,B{} java doesn’t support this type of inheritance. Which we can achieve using interface.

Oops relationship

Manager/ Developer is a Employee

Employee has a Address

class Employee {

id,name,salary

Scanner obj = new Scanner();

Address add = new Address();

readEmp()

disEmp();

}

class Manager extends Employee{

numberOfEmp;

readMgr()

disMgr()

}

class Developer extends Employee{

projectName;

readDev()

disDev()

}

class ProjectManager extends Manager{

clientId

readPMgr();

disPmgr();

}

class Address {

city,state

Scanner obj = new Scanner();

readAdd()

disAdd();

}

**Polymorphism :**

One name and many forms or many implementation

2 types

1. Compile time or static binding or early binding

Method overloading:

Method have same name but different parameter list ie type of parameter list or number of parameter list must be different.

1. Run time or late binding or dynamic binding

Method overriding: method have same name and same method signature (number of parameter list, type of parameter list and return type must be same) . we can achieve method overriding using inheritance.