DAY - 5

Inheritance :

Accessing One Class Property into another class with the help of "extends" keyword.

Purpose of Inheritance :

- to reduce the memory wastage
- and to reduce the object creation

In Java, there are two classes:

1. Parent class (Super or Base class)

2. Child class (Subclass or Derived class) ▪

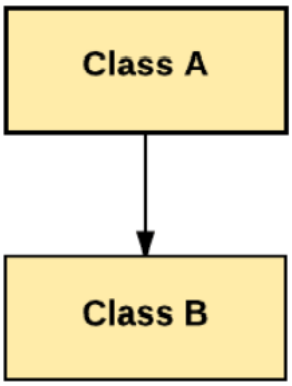
A class which inherits the properties is known as Child Class whereas a class whose properties are inherited is known as Parent class

Types of Inheritance

- Single
- Multilevel
- Hierarchical
- Multiple
- Hybrid

Single Inheritance

In single inheritance, one class inherits the properties of another.

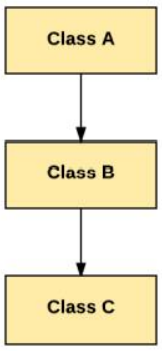



```
Class A
{
---
}
Class B extends A {
---
}
```

Here, Class A is your parent class and Class B is your child class which inherits the properties and behavior of the parent class.

Multilevel Inheritance

When a class is derived from a class which is also derived from another class, i.e. a class having more than one parent class but at different levels, such type of inheritance is called Multilevel Inheritance.



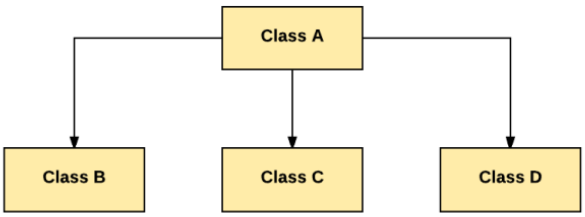

```
Class A{
---
}
Class B extends A{
---
}
Class C extends B{
---
}
```

If we talk about the flowchart, class B inherits the properties and behavior of class A and class C inherits the properties of class B. Here A is the parent class

for B and class B is the parent class for C. So in this case class C implicitly inherits the properties and methods of class A along with Class B. That's what is multilevel inheritance

Hierarchical Inheritance

When a class has more than one child classes (sub classes) or in other words, more than one child classes have the same parent class, then such kind of inheritance is known as hierarchical

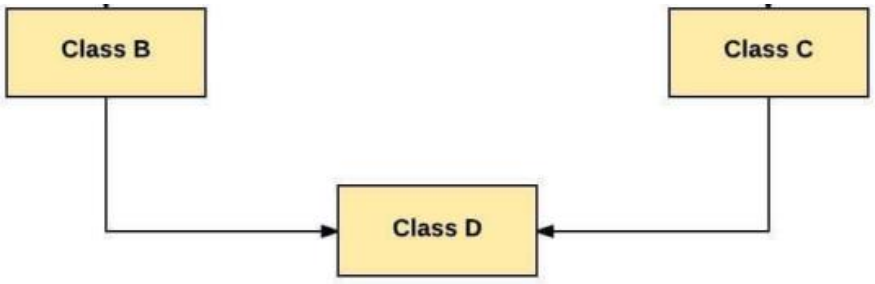



```
Class A{
---
}
Class B extends A{
---
}
Class C extends A{
---
}
```

If we talk about the flowchart, Class B and C are the child classes which are inheriting from the parent class i.e Class A

Multiple Inheritance

In this inheritance, a derived class is created from more than one base class. This inheritance is not supported Java Language. It can be achieved through Interfaces.

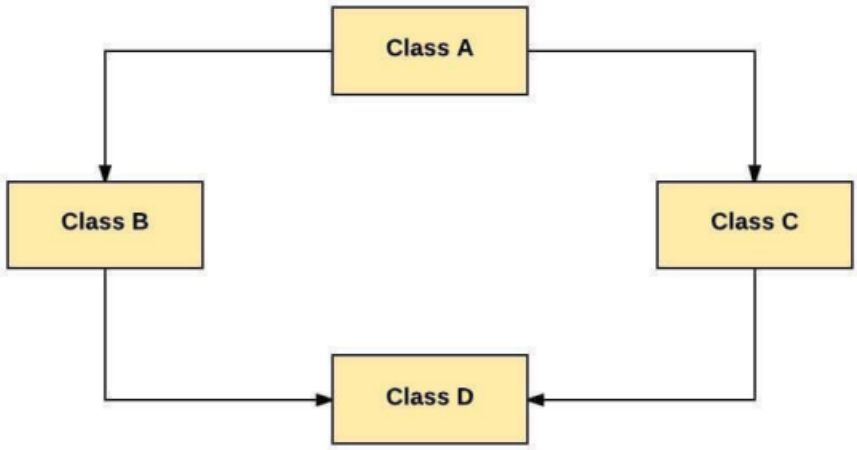


In the given example, class D inherits the properties and behavior of class B and class C at the same level. So, here B and Class C both are the parent classes for Class D.

Hybrid Inheritance

This is a combination of more than one inheritance. Hence, it may be a combination of Multilevel and Multiple inheritance or Hierarchical and Multilevel inheritance or Hierarchical, Multilevel and Multiple inheritances.

In Hybrid inheritance a combination of multiple inheritance and multilevel inheritance is not possible. Since multiple inheritance is not supported in Java as it leads to ambiguity, so this type of inheritance can only be achieved through the use of the interfaces.



If we talk about the flowchart, class A is a parent class for class B and C, whereas Class B and C are the parent class of D which is the only child class of B and C.