

Large Scale Data Processing

CSE3025

Dr. Ramesh Ragala

School of Computer Science and Engineering
VIT Chennai

February 19, 2021



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

- 1 Inheritance in Java
- 2 Polymorphism in Java

Inheritance in Java

- It is a mechanism in which one object acquires all the properties and behaviors of its parent object.
- The idea behind inheritance in Java is that you can create new classes that are built upon existing classes.
- The class which inherits the properties of other is known as subclass (derived class, child class).
- Superclass is the class from where a subclass inherits the features. It is also called a base class or a parent class.
- It is possible to reuse methods and fields of the parent class after inheriting from the parent class.
- The child class can add its own methods and fields also
- Inheritance represents the IS-A relationship which is also known as a Parent-Child relationship.

syntax

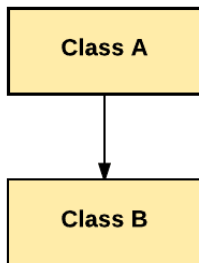
```
class Subclass-name extends Superclass-name
```

```
{
```

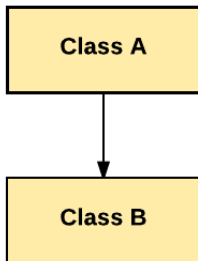
Methods and fields

```
}
```

- **extends** is a keyword, which is used to inherit the properties of a class
- There are Various types of inheritance in Java
 - ▶ Single Inheritance
 - ▶ Multiple Inheritance
 - ▶ Multilevel Inheritance
 - ▶ Hierarchical Inheritance
 - ▶ Hybrid Inheritance

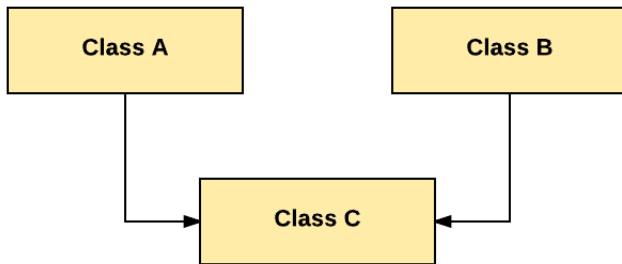


Single Inheritance:



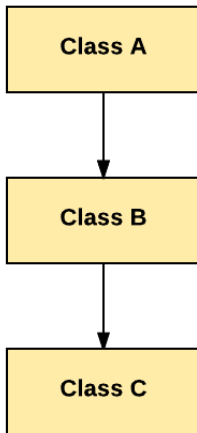
- **Single Inheritance:**

- In this, one class extends another class (one class only)
- In the above diagram, class-B is sub-class and class-A is super-class



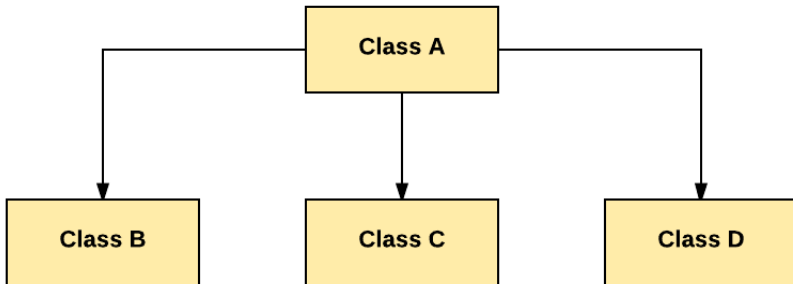
- **Multiple Inheritance:**

- In this, one class extending more than one class.
- Multiple inheritance often leads to problems in the hierarchy.
- Most of the new OO languages like Small Talk, Java, C# do not support Multiple inheritance.

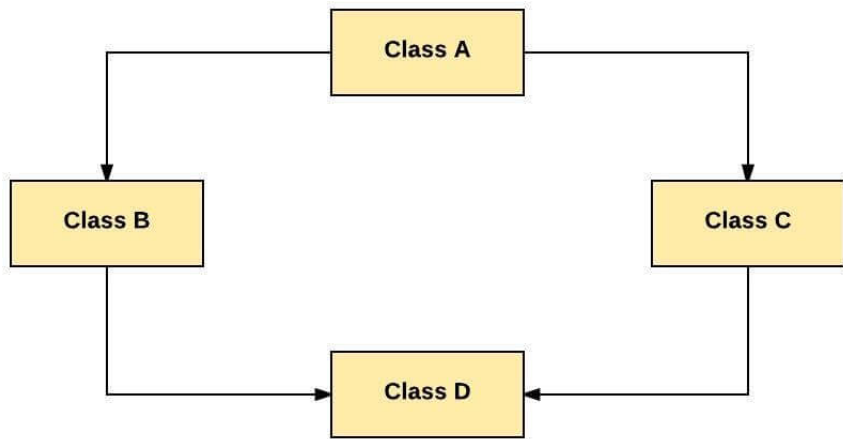


- **Multilevel Inheritance:**

- In this, one class can inherit from a derived class
- Hence, the derived class becomes the base class for the new class.



- **Hierarchical Inheritance:**
- In this, one class is inherited by many sub classes.



- **Hybrid Inheritance:**

- In is a combination of Single and Multiple inheritance.
- Java doesn't support hybrid/Multiple inheritance

1 Inheritance in Java

2 Polymorphism in Java

Polymorphism in Java

- Polymorphism is one of the OOPs feature that allows us to perform a single action in different ways.
- Polymorphism means "many forms", and it occurs when we have many classes that are related to each other by inheritance.
- Two types of polymorphisms in java: Compile-time polymorphism and Run-time polymorphism
- We can perform polymorphism in java by method overloading and method overriding.
- If overload happens on a static method in Java, then it is the example of compile time polymorphism.
- Runtime polymorphism is a process in which a call to an overridden method is resolved at runtime rather than compile-time.

Method Overloading in Java

- If a class has multiple methods having same name but different in parameters, it is known as Method Overloading.
- it increases the readability of the program.
- **Different ways to overload a method:**
 - ▶ By changing number of arguments
 - ▶ By changing the data type
- In Java, Method Overloading is not possible by changing the return type of the method only.

Method Overriding in Java

- If subclass (child class) has the same method as declared in the parent class, Then it is called as method overriding in Java.
- Method overriding is used to provide the specific implementation of a method which is already provided by its superclass.
- Method overriding is used for runtime polymorphism
- **Rules for method overriding in java:**
 - The method must have the same name as in the parent class
 - The method must have the same parameter as in the parent class.
 - There must be an IS-A relationship (inheritance).

Super keyword in Java

- The super keyword in Java is a reference variable which is used to refer immediate parent class object.
- Whenever you create the instance of subclass, an instance of parent class is created implicitly which is referred by super reference variable.
- super can be used to refer immediate parent class instance variable.
- super can be used to invoke immediate parent class method.
- super() can be used to invoke immediate parent class constructor.