

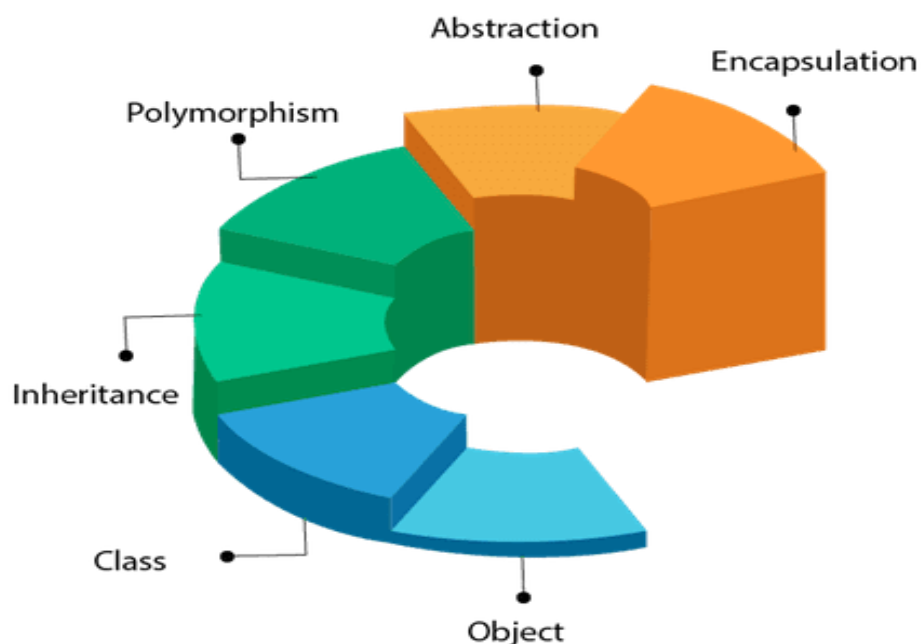
OOPs (Object-Oriented Programming System)

Object means a real-world entity such as a pen, chair, table, computer, watch, etc. **Object-Oriented Programming** is a methodology to design a program using classes and objects.

Pillars/feature of OOPs:-

1. Object
2. Class
3. Inheritance
4. Polymorphism
5. Abstraction
6. Encapsulation

OOPs (Object-Oriented Programming System)



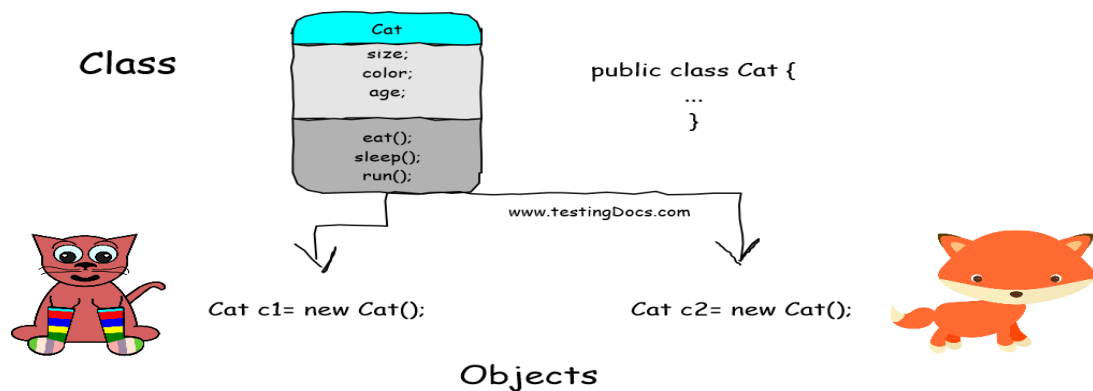
Object: -

Any entity which have state and behaviour is known as object. For example a car, book, clock, laptop, bike, etc.



- ❖ **State:** represents the data (value) of an object.
- ❖ **Behaviour:** represents the behaviour (functionality) of an object such as deposit, withdrawal, etc.
- An Object can be defined as an instance of a class, and there can be multiple instances of a class in a program.
- It can be physical or logical entity.
- An object contains an address and takes up some space in memory.

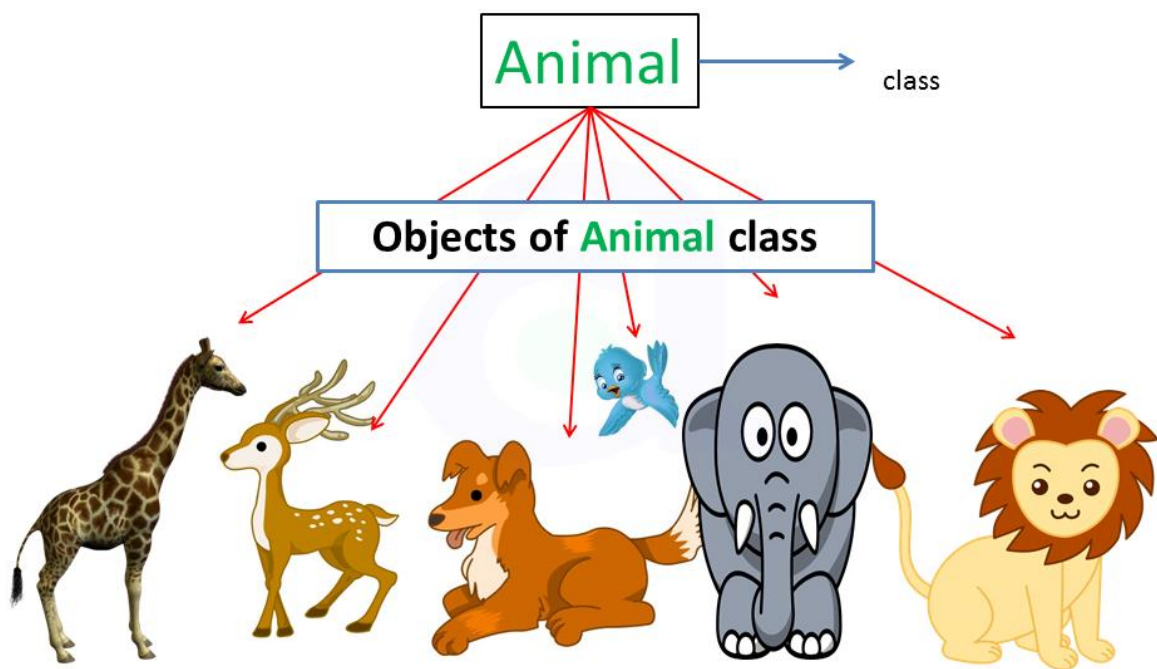
Example: A cat is an object as it has states like color, name etc. as well as behaviors like running, eating etc.



Class: -

- Class is collection of objects, methods, constructors, variables, statements, datatypes etc.
- Class is a blueprint or a set of instructions to build a specific type of object.
- Class in Java determines how an object will behave and what the object will contain.
- It is a logical entity, which doesn't consume any space.
- A class can also be defined as a blueprint or protocol from which objects are created.

Class and Object Relation -



=====

Inheritance

1. Inheritance in Java is a mechanism in which one class acquires all the properties and behaviours of another class with the help of extends keyword.
2. Inheritance is one of the most important Object-oriented programming language system.
3. Also, we can say that subclass can acquire the properties of superclass with the help of extends keyword
4. Inheritance represents the **IS-A relationship** which is also known as a *parent-child* relationship.

Why use inheritance in java

1. For Code Reusability.
2. For code optimization

Terms used in Inheritance

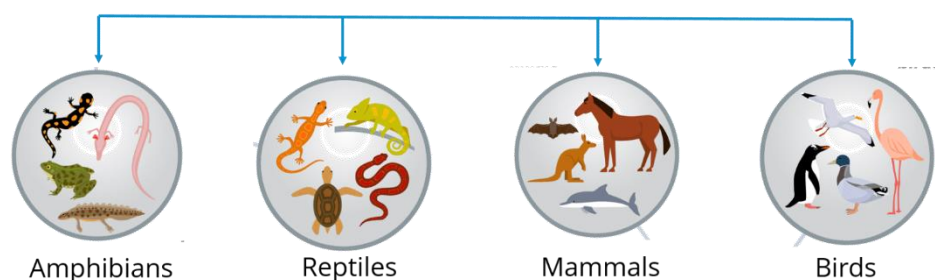
- **Sub Class/Child Class:** Subclass is a class which inherits the other class. It is also called a *derived class, extended class, or child class*.
- **Super Class/Parent Class:** Superclass is the class from where a subclass inherits the features. It is also called a *base class or a parent class*.

Super Class



Animals

Child Class



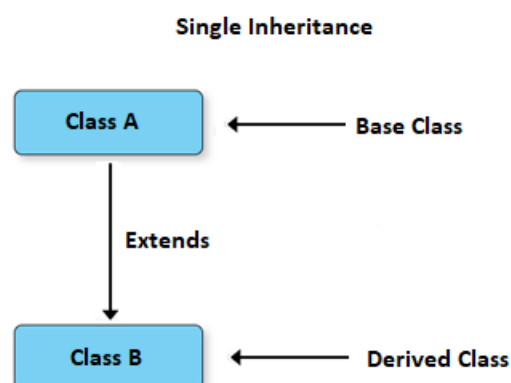
Types of Inheritance

1. Single Level Inheritance
2. Multi-Level Inheritance
3. Multiple Inheritance
4. Hierarchical Inheritance
5. Hybrid Inheritance

=====

Single Level Inheritance

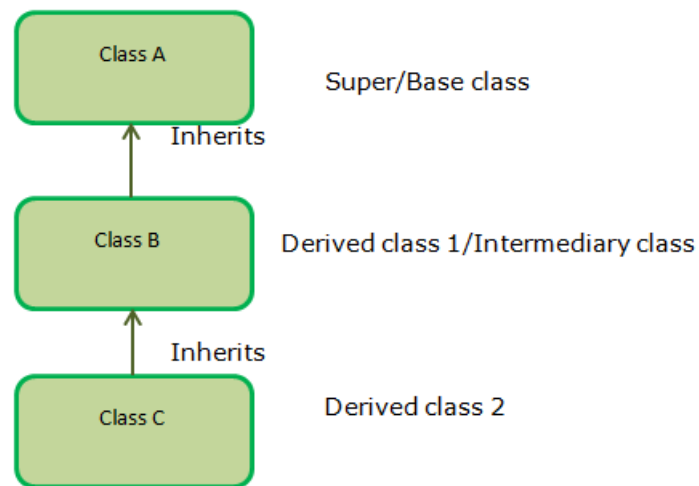
- The process of inheritance in which a class acquires/inherits the properties of another class by using extends keyword is called as Single Level Inheritance.
- It is an operation where inheritance takes place between two classes only to perform Single Level Inheritance



Eg. ClassB inherits classA.

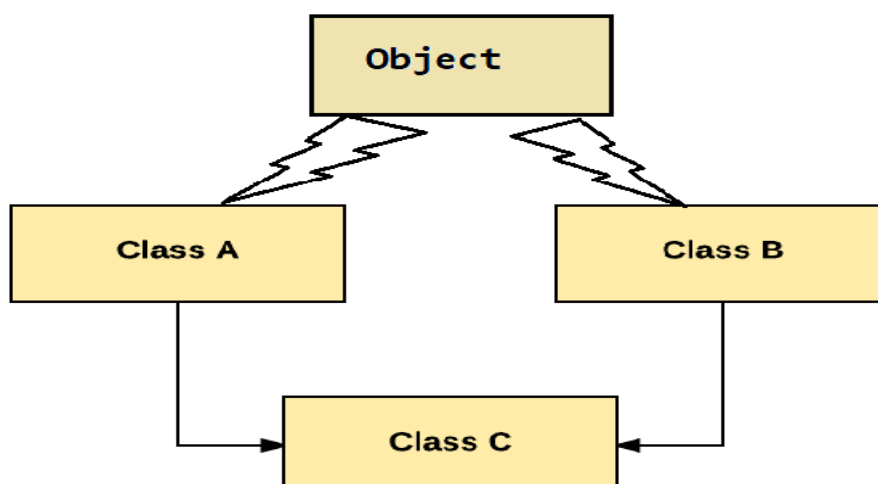
Multi-Level Inheritance

- Multilevel inheritance takes place between three or more classes.
- The process of inheritance in which one subclass acquires the properties of another superclass with the help of extends keyword and this phenomenon continues so that we called as ***"Multi level Inheritance"***
- In short when there is a chain of inheritance it is also known as multilevel inheritance.



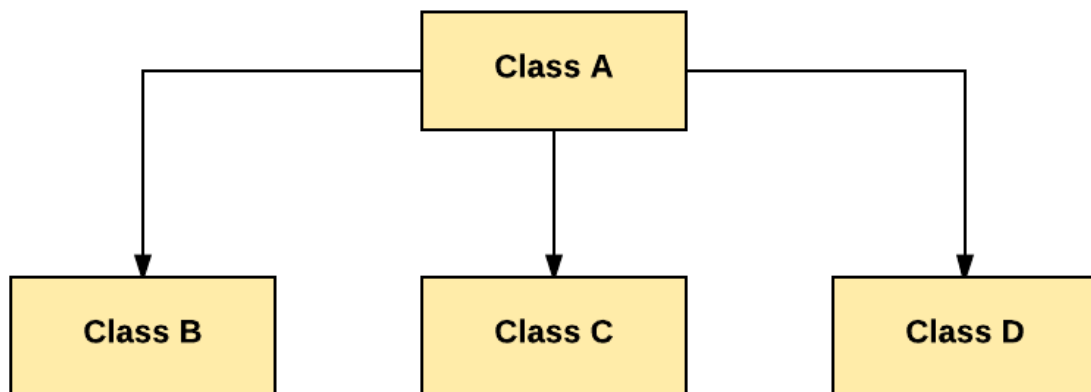
Multiple Inheritance

- The Process of inheritance in which one subclass acquires the properties of two or more super-classes at a same time with the help of extends keyword is called as Multiple Inheritance.
- Java doesn't support multiple inheritance because it results diamond ambiguity problem.
- It means when we try to inherits properties from more than two classes at the same time, the diamond like structure gets created in JVM and object variable gets confused for who need to inherited the property of superclass into subclass.
- Super Class of all the classes is object class so their diamond ambiguity forms
- To reduce the complexity of language multiple inheritance not supported by java, but we can achieve with the help of interface.



Hierarchical Inheritance

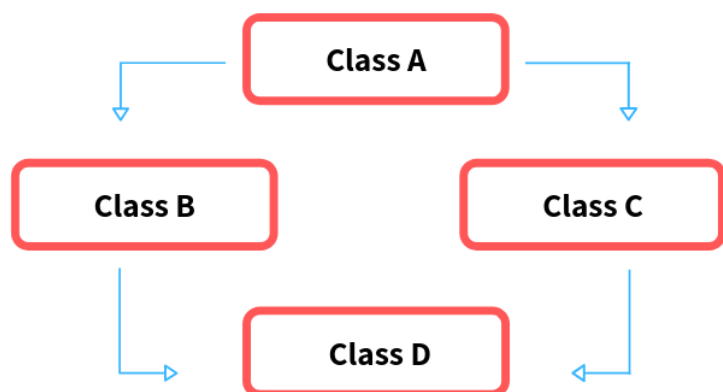
- The process of inheritance in which multiple subclasses acquires the properties one superclass with help of extends keyword is called as Hierarchical Inheritance.
- Hierarchical Inheritance takes place between one superclass and multiple subclasses.
- The two or more classes inherits a single class it is known as Hierarchical Inheritance



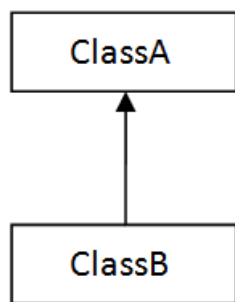
Hybrid Inheritance

- Hybrid Inheritance is a combination of Inheritances. Since in Java Multiple Inheritance is not supported directly so hybrid inheritance is also not support but we can achieve Hybrid inheritance through Interfaces.

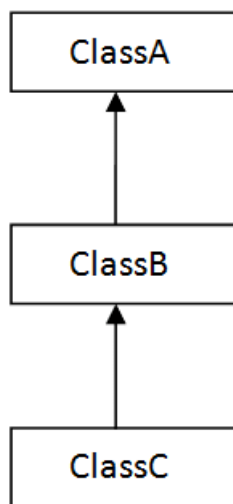
**COMBINATION OF
SINGLE
AND
MULTIPLE
INHERITANCE.**



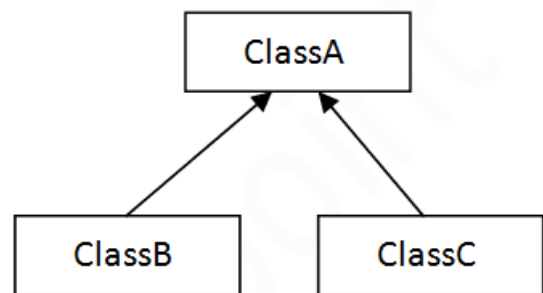
=====



1) Single

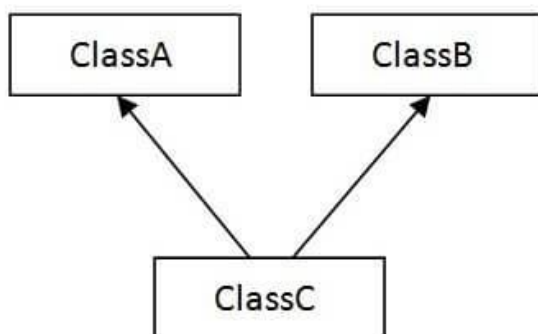


2) Multilevel

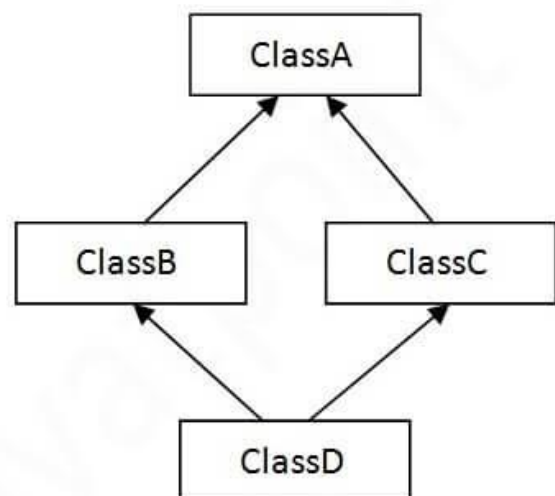


3) Hierarchical

Note: Multiple inheritance and Hybrid inheritance is not supported in Java through class.



4) Multiple



5) Hybrid

What is the difference between an object-oriented programming language and object-based programming language?

Object-based programming language follows all the features of OOPs except Inheritance. JavaScript and VBScript are examples of object-based programming languages.

Who is the superclass of all the classes?

=====