INHERITANCE IN JAVA

Inheritance can be defined as the process where one class acquires the properties (methods and fields) of another.

The class that inherits the properties of others is known as subclass (derived class, child class) and the class whose properties are inherited is known as superclass (base class, parent class). Inheritance represents the **IS-A relationship**.

USE OF INHERITANCE

- For Method Overriding.
- For Code Reusability.

SYNTAX OF INHERITANCE

```
class Subclass-name extends Superclass-name
{     //methods and fields
}
```

Example:
class Dog extends Animal

Ex:
class Child extends Parent
{
Statement/s
}

extends is the keyword used to inherit the properties of a class.

Example:(Without extends keyword)

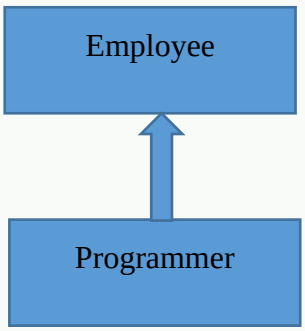
```
class Employee{
          float salary=40000;
         }

         class Programmer {
          int bonus=10000;

           public static void main(String args[]){
            Programmer p=new Programmer();
            System.out.println("Programmer salary is:"+p.salary);
            System.out.println("Bonus of Programmer is:"+p.bonus);
         }
         }
```

```
C:\Windows\System32\cmd.exe
F:\Java Code 2020>javac Programmer.java
Programmer.java:10: error: cannot find symbol
          System.out.println("Programmer salary is:"+p.salary);
                                                      ^
  symbol:   variable salary
  location: variable p of type Programmer
1 error
```

After using the keyword **extends**




```
class Employee{
          float salary=40000;
          }

          class Programmer extends Employee {
           int bonus=10000;


           public static void main(String args[]){
             Programmer p=new Programmer();
             System.out.println("Programmer salary is:"+p.salary);
             System.out.println("Bonus of Programmer is:"+p.bonus);
          }
          }
```

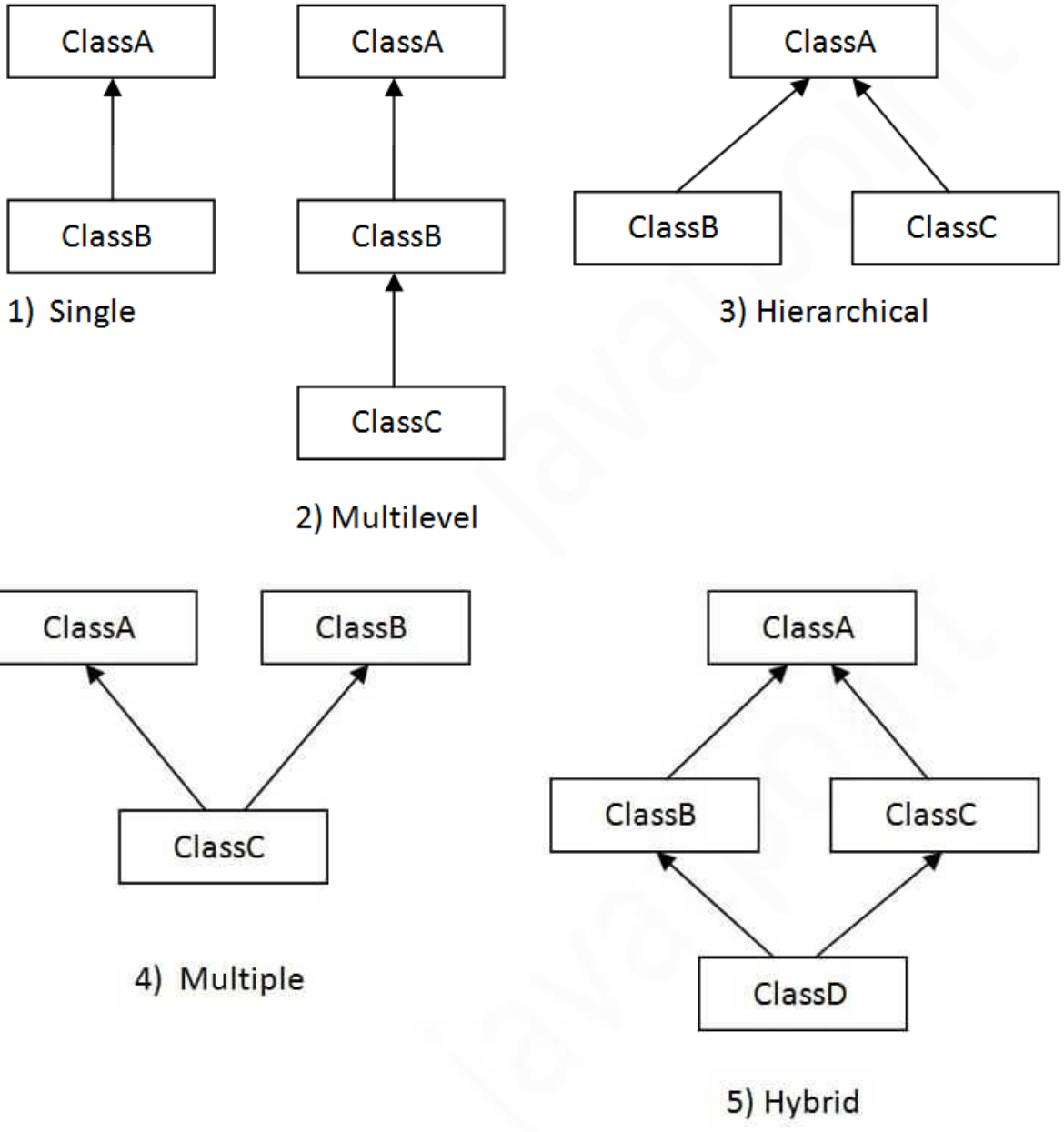
```
F:\Java Code 2020>javac Programmer.java

F:\Java Code 2020>java Programmer
Programmer salary is:40000.0
Bonus of Programmer is:10000
```

In the above example, Programmer object p can access the field of own class as well as of Employee class i.e. code reusability.

TYPES OF INHERITANCE

There are various types of inheritance as demonstrated below.



.

When one class inherits multiple classes, it is known as multiple inheritances.

Multiple inheritance is not supported in Java through the class.

WHY MULTIPLE INHERITANCE IS NOT SUPPORTED IN JAVA?

To reduce the complexity and simplify the language, multiple inheritances are not supported in Java.

example:

```
class A{
void message(){System.out.println("Hello");}
}

class B{
void message(){System.out.println("Welcome");}
}

class C extends A,B{

 public static void main(String args[]){
   C obj=new C();
   obj.msg();//Now which message() method would be invoked?
 }
}
```

Output: **Compile Time Error**

Example:

```
class Parent
{
public void m1()
{
System.out.println("Parent Method");
}
}

class Child extends Parent
{
public void m2()
{
System.out.println("Child Method");
}
}

class TestPC
{
public static void main(String args[])
{
  Parent p=new Parent();
  p.m1();
  p.m2();

  Child c = new Child();
  c.m1();
  c.m2();

  Parent p1=new Child();
  p1.m1();
  p1.m2();

  Child c1 = new Parent();
  c1.m1();
  c1.m2();
  }
  }
```

Case 0:
Parent p=new Parent();
p.m1();

```
F:\Java Code 2020>javac TestPC.java

F:\Java Code 2020>java TestPC
Parent Method
```

Case 1:
Parent p=new Parent();
p.m2();

```
F:\Java Code 2020>javac TestPC.java
TestPC.java:22: error: cannot find symbol
  p.m2();
   ^
  symbol:   method m2()
  location: variable p of type Parent
1 error
```

Case 3:
Child c = new Child();
c.m1();
c.m2();

Whatever the parent class has by default available to the child, means child class reference can call both parent and child class methods.

```
F:\Java Code 2020>javac TestPC.java

F:\Java Code 2020>java TestPC
Parent Method
Child Method
```

Case 4:
Parent p1=new Child();--------1
p1.m1();

```
F:\Java Code 2020>javac TestPC.java

F:\Java Code 2020>java TestPC
Parent Method
```

Case 5:
Parent p1=new Child(); ----1
p1.m2();

Whatever the child class has by default is not available to the parent class (**Case 5, Case 1**), hence parent class reference can call only parent class methods(**case 0,case 4**), and cannot call child class methods(**Case 5,Case 1**),.

Parent class reference can be used to hold child class objects(**case 4 -----1**,**case 5**---1).By using that reference we can call only parent class methods(**case 4**) but we cannot call child class methods(**case 5**).

```
F:\Java Code 2020>javac TestPC.java
TestPC.java:22: error: cannot find symbol
  p1.m2();
    ^
  symbol:   method m2()
  location: variable p1 of type Parent
1 error
```

Case 6:
Child c1 = new Parent();

We can't use child class references to hold parent class objects(**Case 6**).

```
F:\Java Code 2020>javac TestPC.java
TestPC.java:21: error: incompatible types: Parent
cannot be converted to Child
  Child c1 = new Parent();
             ^
```

Object class in Java:

- Object class is present in java.lang package. Every class in Java is directly or indirectly derived from the Object class.
- If a Class does not extend any other class, then it is a direct child class of Object and if extends another class then it is indirectly derived. Therefore, the Object class methods are available to all Java classes. Hence Object class acts as a root of the inheritance hierarchy in any Java Program.
- The common functionality, which is required for any java class, is defined in the Object class and by keeping that class as a superclass, its functionality by default is available to every java class.