Oops

Constructor : constructor is a type special method which help to create the object.

Pts

1. Constructor have same name as class itself. But in Typescript or angular we need create the function with name as constructor.
2. Constructor no return type not even void also.
3. Constructor get call automatically when we create the object.

Parameter constructor

If local variable or parameter variable and instance variable have same name then local variable or parameter variable hide the visibility of instance variable.

But if we want to refer to instance variable then we need to use this.instancevariable.

In the life of the object if we want to perform any task only one time. That type of task we need to write inside a constructor.

If the life of the object if we want to perform any task more than one time that type of task we need to write inside method.

Encapsulation : Binding or wrapping data or variable and code or method/functions in a single unit is known as Encapsulation.

Ex : class.

Java provided one of the pre-defined class ie JavaBean.

JavaBean class is known as pure encapsulation class.

1. All variable in java class must be private.
2. For each variable we need to write setter and getter methods

Inheritance : Inheritance is use to inherits property and behaviour of old class to new class.

class OldClass { super class, base class or parent class.

property

behaviour

}

Sub class, derived class or child class.

class NewClass extends OldClass{

property

behaviour

}

Types of inheritance

1. Single inheritance : One super class and one sub class.
   class A { }

   class B extends A {}
2. Multilevel inheritance : one super class and n number of sub classes they are connected one by one
   class A { }
   class B extends A {}
   class C extends B {}
3. Hierarchical inheritance : one super class and n number of sub class directedly connected to super class.
   class A { }
   class B extends A {}
   class C extends A {}
4. 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. In java using class we can't extends more than one class.

this type of inheritance java support using interface.

Super class must be generic

```
class Employee {
        id,name,salary
}
class Manager extends Employee{
    numberOfEmp;
}
class ProjectManager extends Manager{
    clientId;
}
class Developer extends Employee{
    projectName;
}
```

Polymorphism : One name many forms or many implementation.

In java we can achieve two type of polymorphism

1. Compile time polymorphism or early binding or static binding
   Example : Method overloading
   The method java same name but different parameter list ie type of parameter list or number of parameter list must be different.
2. Run time polymorphism or late binding or dynamic binding
   Example : Method overriding