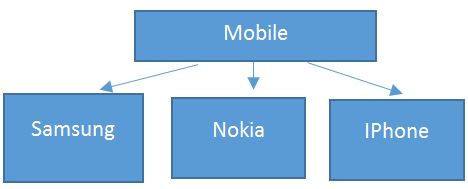
Object Oriented Programming is basically considered as a design methodology for creating a non-rigid application. In OOPS every logic is written to get our work done, but this is done based on entity which we call it as Objects. OOP allow us to decompose our problem in to small unit of work which are accessed via Objects. We build function around this objects. There are mainly four pillars (features) of OOP.  
  
If all this features fulfill our programming, then we can say it as perfect Object Oriented Programming.

1. Abstraction
2. Encapsulation
3. Inheritance
4. Polymorphism
5. 

In above diagram, each brand (Samsung, Nokia, IPhone) have their own list of features along with basic functionality of dialing, receiving a call & messaging.   
  
When we talk about OOP, as the word indicate it will talk about an object (a real world object)

**Class**  
A Class is a plan which describes the object. We call it as a blue print of how the object should be represented. Mainly a class would consist of a name, attributes & operations.

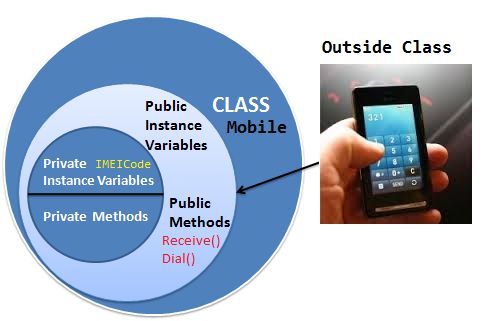
**Objects**  
Any real world entity which can have some characteristics or which can perform some work is called as Object. This object is also called as an instance i.e. - a copy of entity in programming language.

**Abstraction**  
Abstraction says, only show relevant details and rest all hide it. This is most important pillar in OOPS as it is providing us the technique to hide irrelevant details from User. If we consider an example of any mobile like Nokia, Samsung, IPhone.   
  
**Some features of mobiles**

1. Dialing a number call some method internally which concatenate the numbers and displays it on screen but what is it doing we don’t know.
2. Clicking on green button actual send signals to calling person’s mobile but we are unaware of how it is doing.

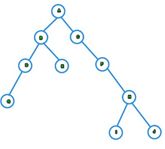
This is called abstraction where creating method which is taking some parameter & returning some result after some logic execution without understating what is written within the method

**Encapsulation**

  
  
Encapsulation is defined as the process of enclosing one or more details from outside world through access right. It says how much access should be given to particular details. Both Abstraction & Encapsulation works hand in hand because Abstraction says what details to be made visible & Encapsulation provides the level of access right to that visible details. i.e. – It implements the desired level of abstraction.

**Polymorphism**  
Polymorphism can be defined as the ability of doing the same operation but with different type of input.  
  
More precisely we say it as ‘many forms of single entity’. This play a vital role in the concept of OOPS.  
  
Let’s say Samsung mobile have the 5MP camera available i.e. – it is having a functionality of CameraClick(). Now same mobile is having Panorama mode available in camera, so functionality would be same but with mode. This type is said to be Static polymorphism or Compile time polymorphism. See the example below:

**Inheritance**

  
Ability to extend the functionality from base entity in new entity belonging to same group. This will help us to reuse the functionality which is defined before.