* Static method can be accessed directly by using the class name
* Non static methods can be accessed only by class reference
* Static methods cannot access non static members directly
* Non static methods can access static members directly
* An object is an instance of a class. The term ‘object’, however refers to an actual instance of a class. Every object must belong to a class
* Objects have a life span but class do not
* **Is-a** relationship is expressed using extends key word (inheritance can be achieved using extends key word)
* **Has-a** relationship means an instance of one class has an instance of another class

**Animal is parent object, cow is child objects**

* Animal a =new cow()

In the above example only the methods present in animal class can be accessed

All the methods of cow class cannot be accessed

* Incase if the methods of parent class is over ridden in child class then child class implementation is implemented
* All the non over ridden methods of child class cannot be accessed
* Cow c = new Animal() X

Parent object cannot referenced to child object

Animal object cannot be created and referenced it to cow, since animal is parent object of cow

* Cow c = new cow()

All the methods of cow class and animal class can be accessed, since cow is extending animal

* Local variables should be initialized to some value at the time of creation
* **Parameters** are the variables declared in a function
* **Arguments**  are the value passed to the parameters
* **Constructors cannot be inherited**
* Parameterized constructors are used to initialize the value of instance variables at the time of object creation
* A class can have more than one constructor but they should vary with **number of parameters or type of parameters**
* One constructor can be called from an another constructor(this is called as constructor chaining)
* We can also call same class constructor using this(), instead of (Animal(‘cow ’) or Animal() 🡺 **this(‘cow’) or this()**)
* **This()** keyword is used to refer the current class objects
* **Super()** can be used to access parent class constructor, super must always be the first line of the constructor
* **Super and this cannot be present inside the same constructor.**
* **Static blocks get executed before the main method**
* Non static methods gets executed at the time of object creation
* The overriding method should have same parameters as that of overridden method
* The return type should be same or a subtype of return type declared in overridden method
* Access level of over riding method cannot be more restrictive than over ridden method
* A method declared static cannot be over ridden but can be redeclared
* Abstract class cannot be instantiated
* If a method is declared as abstract then it is mandatory to declare class as abstract
* Instance block will be executed when ever a new object is created
* Static block gets executed before the main method or it gets executed before class gets loaded