

1. Class

- a. Skeleton that wraps data members into single unit called class. It's blue print.
- b. Variables
- c. Methods
- d. Constructors: do not have any return type or void and should have name same as class name.
 - i. Default
 - 1. Present in class by default. If any parametrized constructor is defined then we need to define default constructor explicitly.
 - ii. Overloaded
 - 1. Defining the constructors with different parameter or default different parameter types
 - 2.
 - a. Demo(int i)
 - b. Demo(byte i , int k)
- e. Class => Constructor => used to create obj => obj is used to access data member

2. Object

- a. new keyword
- b. Memory Assignment

3. this keyword (important)

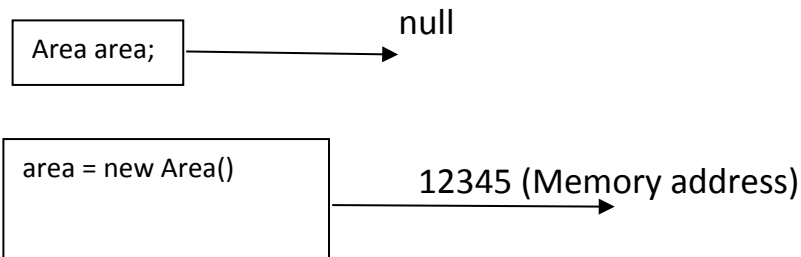
- a. refers to instance of class within the same class (similar to creating object using new). This keyword can not be used within static context. (will cover later with example)
- b. used to call the constructors (calling over loaded constructor from other constructor within same class)
- c. distinguish between local and instance variables

d. can not be used within the static context

e. Calling constructor (overloaded or any other) and this should be the first statement inside the constructor.

```
Area area;
```

```
area= new Area();
```



1. Static Keyword

- a. Used to define class members (variables or methods)
- b. Directly can be accessed by class name
- c. Non-static cannot be directly used within static context. It can be used with help of an object
- d. This keyword can not be used inside static block of code.

2. Initialization Block

- a. Static init block
 - i. Called only once by default when class is loaded
 - ii. Used to set **final** variables value or used to call initial set up or pre-requisite required
- b. Non- static init block
 - i. Called by default on constructor calling
 - ii. Executes before constructor gets executed

3. Sequence of calling

- a. Load class (static block)
- b. Call cons using new keyword (non- static block)

4. Sequence of execution

- a. Static block
 - b. Non-static block
 - c. Calling constructor if and only if **this()** is used and completes execution
 - d. Constructor
5. Call by Reference
- a. When an object is passed as an argument to a method.
6. Object Class
- a. Super class for all the class
 - b. toString() is called by default in println()
1. Types of Classes
- a. Nested class: Class inside a class
 - i. Inner
 - Class within another class
 - ii. Local
 - Class inside method
 - b. Static class
 - Inner class with static declaration
 - c. Anonymous inner class
 - Un-named class
 - Makes use of **Lambda expression(java 8) feature**
 - Class with no name and class keyword
 - **Used for functional interface**