**Flow Control Statements:**

1. Selection Statements
   1. If -- we write only Boolean conditions eg: if(age>40)
   2. If-else
   3. Switch
   4. Else-if
2. Iteration Statements
   1. For
   2. For-each
   3. While – Condition must be Boolean & Mandatory
   4. Do-While
3. Transfer Statements
   1. Break
   2. Continue
   3. Return
   4. Try

**Class Vs Object:-**

* **Class is a logical entity it contains logics of the application whereas object is physical entity it is representing memory.**
* Class is blue print it decides object creation without class we are unable to create object.
* **Based on single class (blue print) it is possible to create multiple objects but every object occupies memory.**
* We are declaring the class by using class keyword but we are creating object by using **new** keyword.

**Class {**

* + 1. **Variables**
    2. **Methods**
    3. **Constructors**
    4. **Instance Blocks**
    5. **Static Blocks**

**}**

1. **Variables:**

**V**ariables are used to store the values by using these values we are achieving project requirements.

* All variables must have a type. Eg: Primitive type, array type, class type or enum type.

1. **Local Variables:**
   1. The variables which are declared inside a **method or constructor or block** those variables are Local.
   2. It is possible to access LV only inside a method or constructor or blocks only. Outside other methods are not able to access.
   3. Memory allocated when method starts and memory released when method completed.
   4. LV stored in stack memory.
2. **Instance Variables:**
   1. The Variables which are declared inside a class but outside of methods those are called IV.
   2. The Scope of IV is inside a class having global visibility.
   3. Memory allocated during object creation & memory released when object is destroyed.
   4. IV stored in Heap memory.
3. **Static Variables:**
   1. The Variables which are declared inside the class but outside of the methods with static modifier those are called SV.
   2. The Scope of SV is inside a class having global visibility.
   3. Memory allocated during .class file loading and memory released at .class file unloading time.
   4. SV stored in non-heap memory.

Note: In Java for IV and SV JVM will assign default values but for the LV the JVM won’t provide default values. So user must assign values to LV otherwise Exception came.

***Instance vs. Static variables:-***

*In case of instance variables the JVM will create separate memory for each and every object it means separate instance variable value for each and every object.*

*In case of static variables irrespective of object creation per class single memory is allocated, here all objects of that class using single copy.*

1. **Methods:**

Methods are used to write the business logics of the application

2.1 Instance Method

Note: For the Instance members is allocated during **object creation** hence access the Instance Members by using Object-name (reference variable)

2.2 Static Method

Note: For the Static member’s memory allocated during **.class file loading** hence access the static members by using class-name.

**Method Signature**: Method name corresponding parameters list is called MS.

**Public void m1(int a, int b) throws Exception{}**

* Note1 : Methods can have any return type like **Primitive type(int,byte,short,long… etc), Class type, Array type, Enum type and Interface type.**