Multithreading

Every Java application has by default 1 thread that is **main** thread.

Uses of Multi-threading:

1]Parallel Processing

2] Threads are used to create Games and Animation.

There are two ways by which you can implement threading:

I. By extending Thread Class
II. By implementing Runnable Interface

Note: Runnable Interface is a **Functional Interface**

An interface with exactly one abstract method is called **@FunctionalInterface.** The major benefit of java 8 functional interfaces is that we can use lambda expressions to instantiate them and avoid using bulky anonymous class implementation.

Example: one-method interfaces such as Runnable, Callable, Comparator, ActionListener, and others.

Important Methods:Thre

```
 I.    Run
 II.   t.start();//To Start a thread
 III.  Sleep
 IV.   t.isAlive();// returns true/false// To test if a thread is still alive
 V.    t.join();//to wait till a thread dies
 VI.   Thread t=Thread.currentThread();// To know the current running thread
 VII.  Thread.sleep(millis);//To stop the execution of a thread for a specific time
 VIII. String name=t.getName();//To get the name of a thread
 IX.   t.setName("new name");//To set a new name to a thread
 X.    int priority_no=t.getPriority();// To get the priority of a thread
 XI.   t.setPriority(int priority_no);// To set the priority of a thread
```

Note: Thread priorities can change from 1 to 10. We can also use the following constants to represent priorities:

```
  * The minimum priority that a thread can have
   public final static int MIN_PRIORITY = 1;

  * The default priority that is assigned to a thread.
   public final static int NORM_PRIORITY = 5;

  * The maximum priority that a thread can have.
public final static int MAX_PRIORITY = 10;
```

1. Which method is executed by the Thread by default?

Ans: public void run() method.

2. What is thread synchronization or Thread-safe?

Ans: When a thread is already acting on an object, preventing any other thread from acting on the same object is called “Thread Synchronization” or “Thread Safe”. The object on which threads are synchronized is called “synchronized object”. Thread synchronization is recommended when multiple threads are used on the same object(in multithreading).

Synchronized object

Synchronization:

Check normal operation: