***OOP’s And Interview Questions In java***

***Questions 1; Immutability in java Class:***

*Class Student {  
  
private String name;*

*Private String age;*

*Private Address address;*

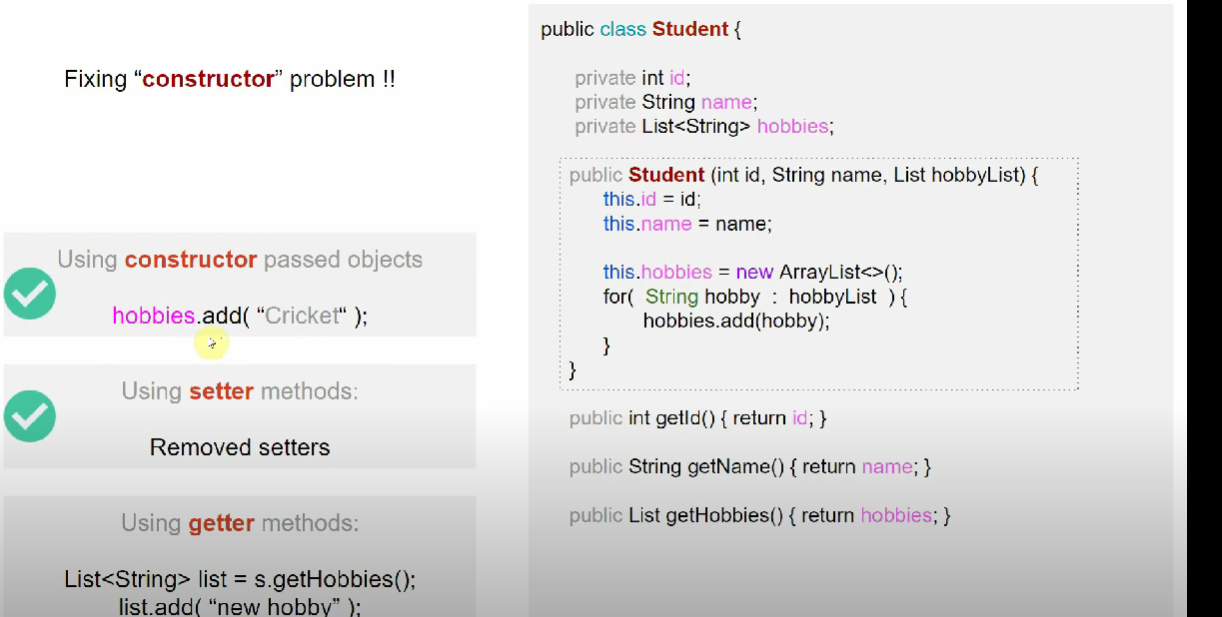
*Public Student(String name, String age, Address add){*

*This.name = name;*

*This.age = age;*

*This.address = new Address(add.getLine1());*

*}*

**

*Public List getHobbies(){*

*List newList = new ArrayList<>();*

*For(string hobby : hobbies){*

*newList.add(hobby);*

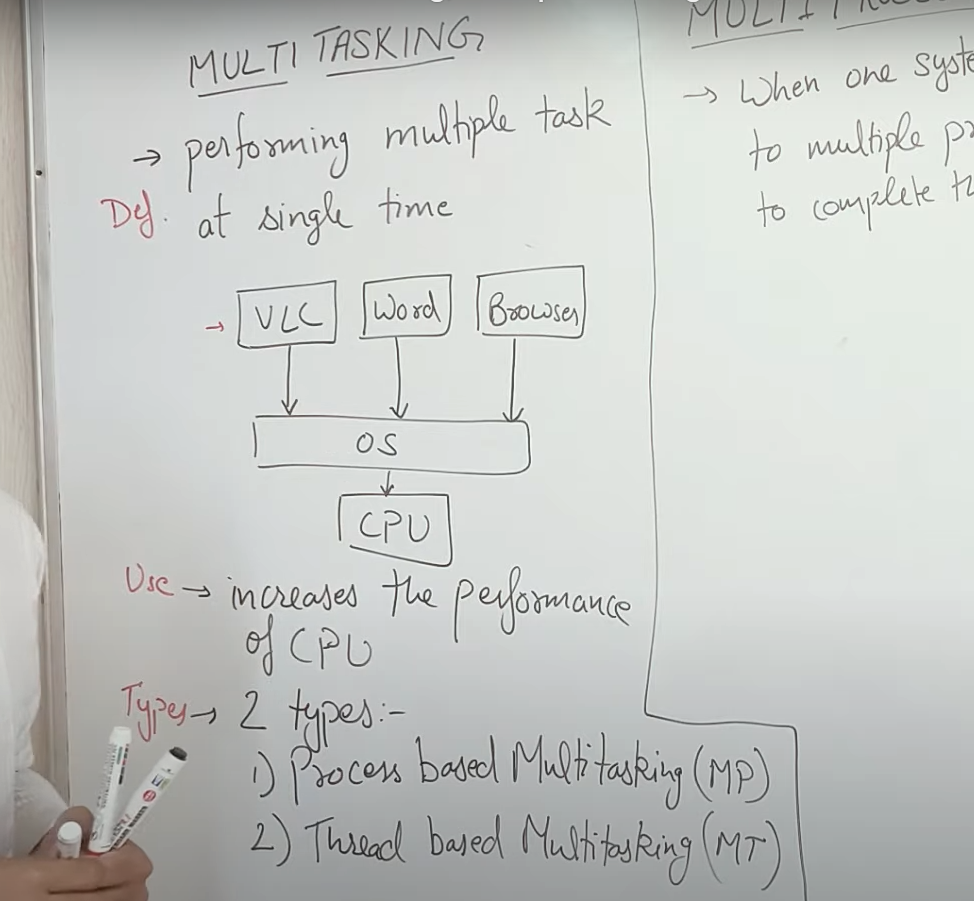
*}*

*Return newList;*

*}*

*This is how we can play safe to add the value using get method*

*MULTI-THREADING :*

**

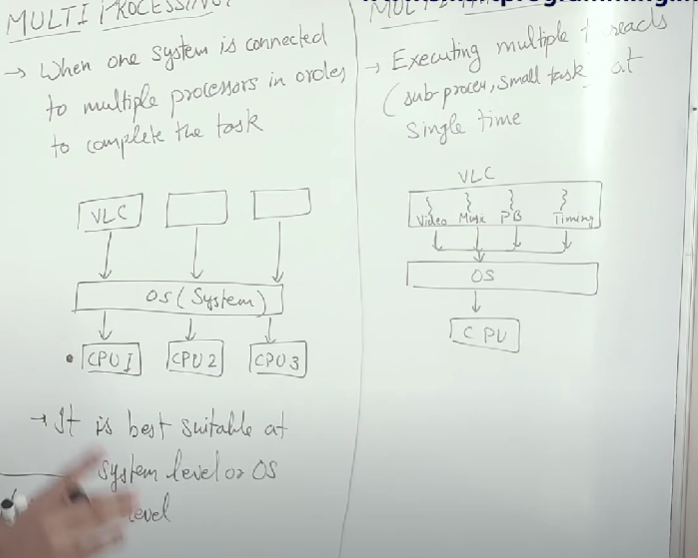
*#MULTI\_PROCESSING:   
  
when one system is connected to multiple CPU in order to achieve the task is called as multiprocessors!!*

* *It’s Best suitable at OS level.*

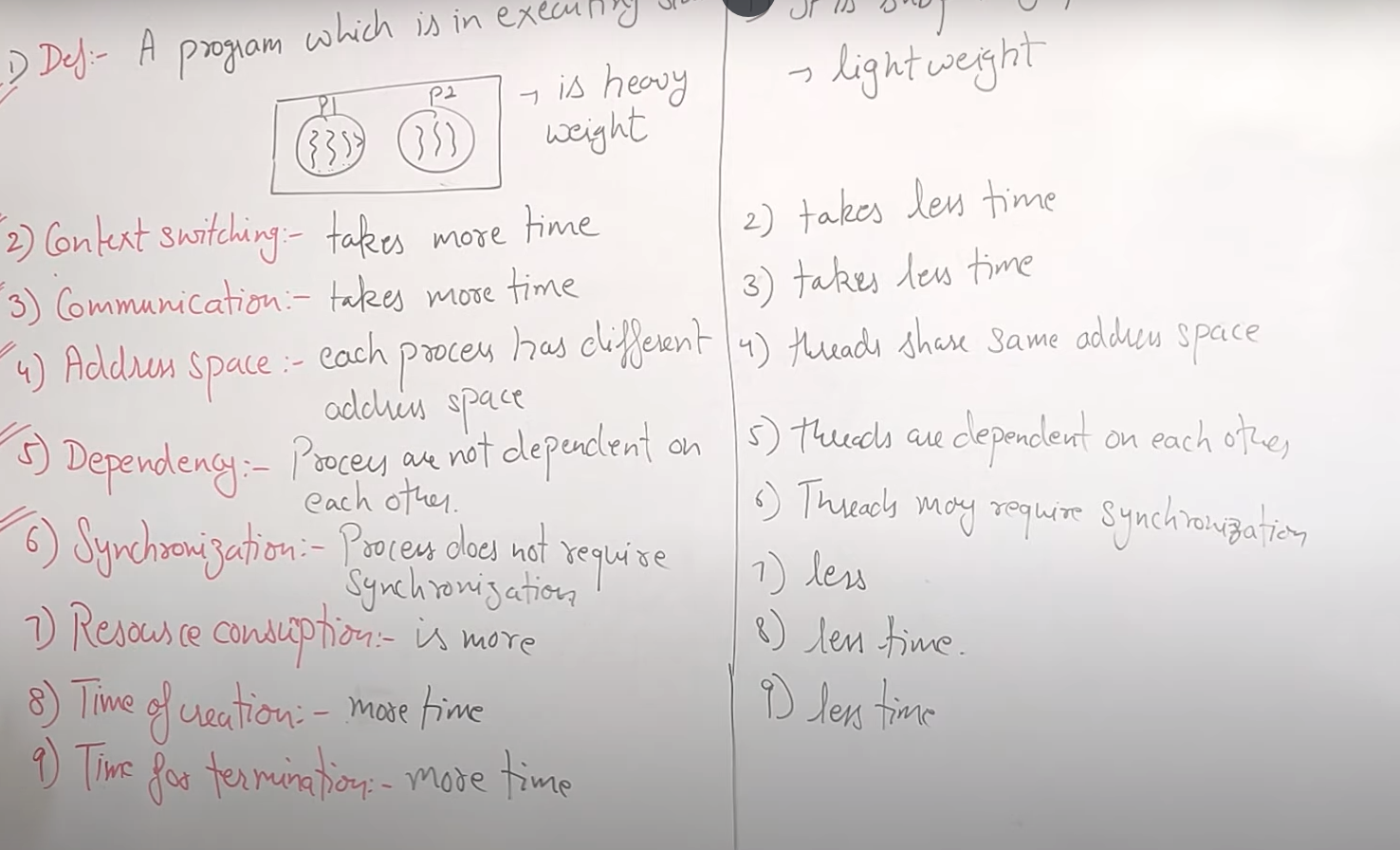
*#Multi-Threading*

*We have one process called media player in that we have small task or jobs that is running at the same time and all of these are called multi threading*

*Multiple threads are working parallelly to complete the job in a single process is called multi threading*

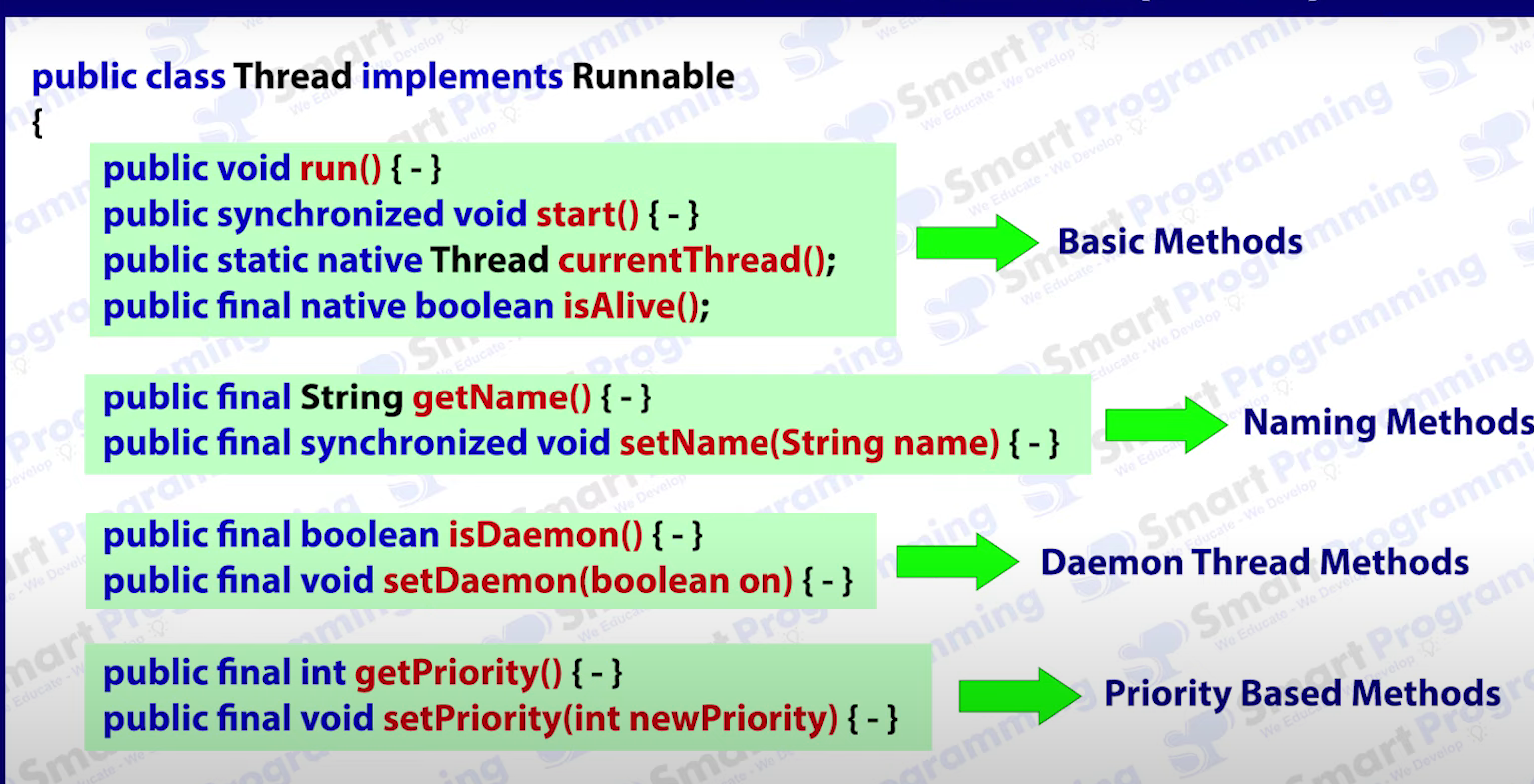
**

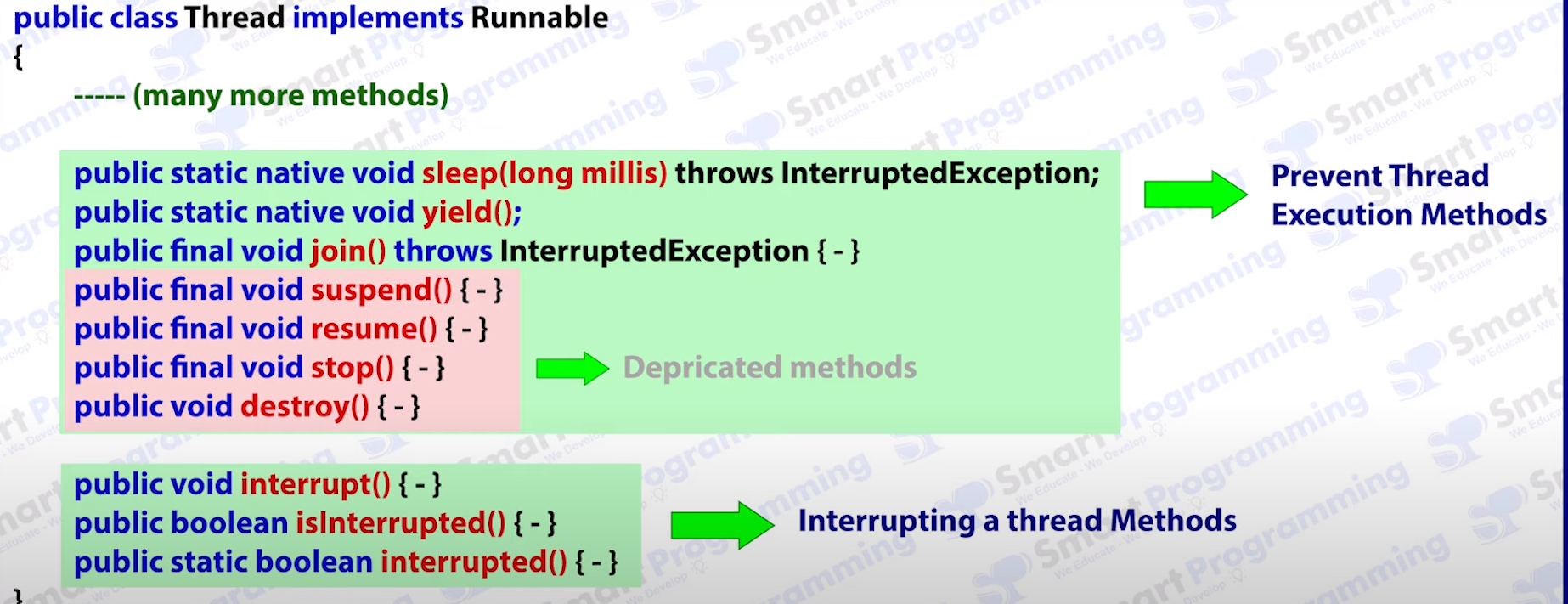
*#Difference Between process and threads!*

**

*#THREAD LIFE CYCLE:  
  
2 ways to create thread in java*

1. *Thread(Class)*
2. *Runnable(interface)*

**

**

*This thread class is in java.lang packages!!*

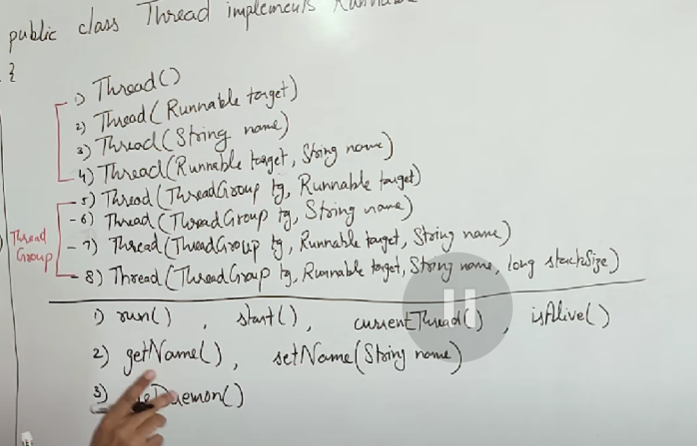
*Through start() method run() calls internally*

*For every thread it will call a stack of thread !!*

*Better way to implement thread is Using interface Runnable*

*Extending thread class is not a good approach if one class is extending another class the multiple inheritance is not allowed in java*

*Constructors of thread class :*

**

*IsAlive() – Check weather thread is in dead state or not*

*getPriority()*

*setPriority()*

*sleep() // prevent the thread execution*

*yield()*

*join()*

*//Depreciated method*

*suspend()*

*resume()*

*stop()*

*destroy()*

*//Thread interruption method*

*Interrupt()*

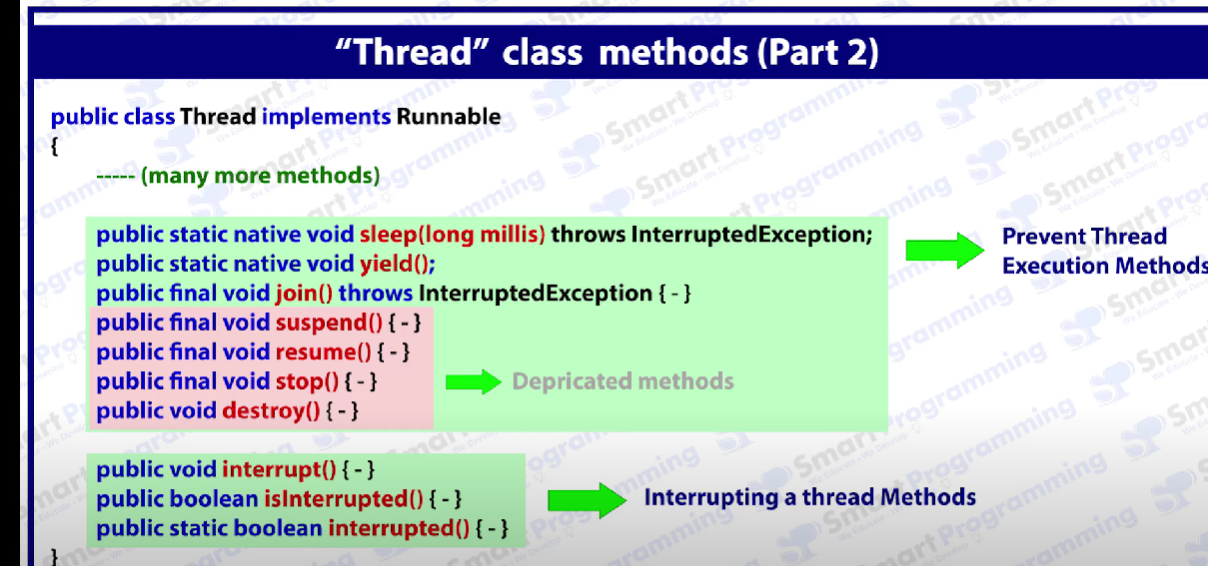
*isInteruptted()*

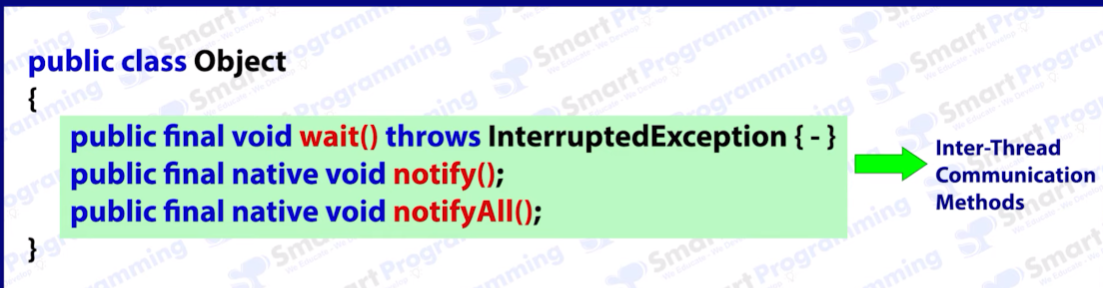
*interrupted()*

*wait() notify()*

**

*CurrentThread and isAlive is of* ***Native*** nature means there implementation is not present in java





In simple program we always have one thread is called main() thread !!

This thread Is created by JVM

***#Daemon Threads:***   
Which runs in the background !!

Usecase : provide the service to the thread

Daemon threads have methods -> public final void Set Daemon(Boolean b){  
}

Public final Boolean isDaemon(){

Return value;

}

Life of a daemon thread is the thread which is running in the background if the thread is in the dead state then daemon thread also goes into the dead state

Daemon thread are the low priority threads which provide the services to the main thread

Ex; Garbage collection using finalize method()

1. It inherits the property of main thread
2. Priority of normal thread same assign to daemon thread
3. If last thread is daemon thread then jvm will kill that thread and goes into the shutdown state
4. If the Main() thread doesn’t work any task then service(Daemon thread will not work)

**Priorities In Thread In Java:-**

Priority are represent in the range from 1 to 10 in threads

1 – min priority

5- norm priority

10 – max priority

Java provides *the* value

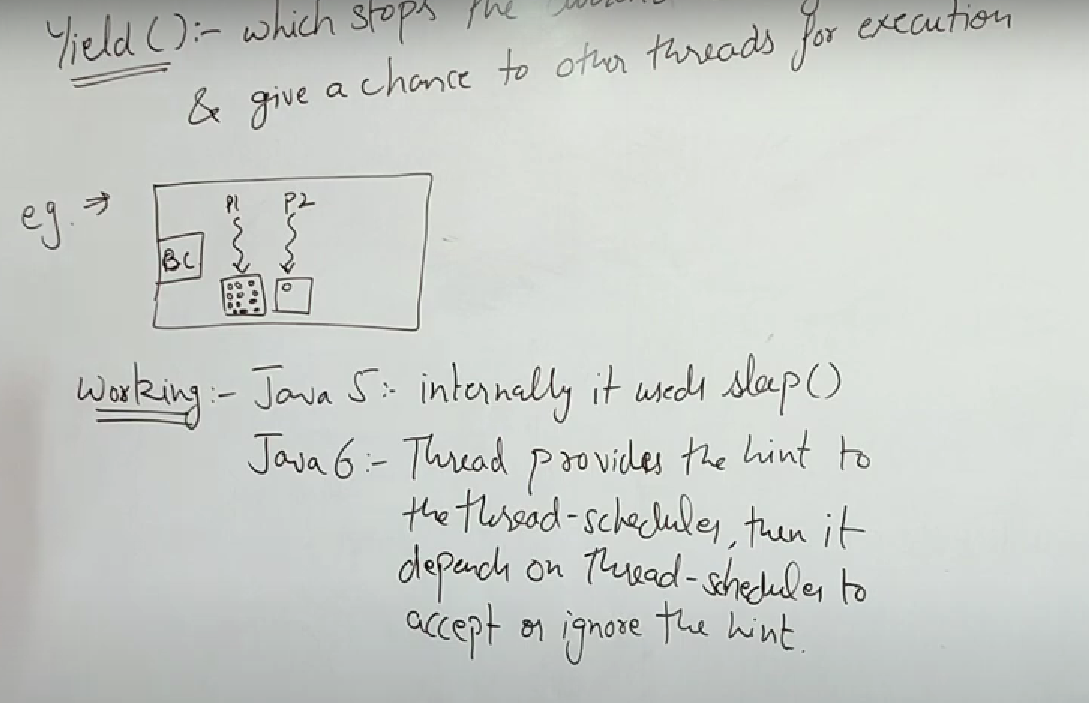
***SLEEP() In Java Multi Threads :***

*One method in thread is native method of sleep(long milliseconds) throws Interrupted Exceptions*

*One method is in java Sleep() throws interrupted exceptions*

***Yeild() In Java:***

Which stops the current thread and give chance to other thread for executions



In java 5 internally Yeild method uses Sleep()