Course Name: A Level (2nd Sem) **Subject: JAVA**
Topic :Multithreading in Java(contd) **Date: 22-04-20**

Difference between multitasking, multithreading and multiprocessing:

Multitasking: Ability to execute more than one task at the same time is known as multitasking.

Multithreading: It is a process of executing multiple threads simultaneously. Multithreading is also known as Thread-based Multitasking.

Multiprocessing: It is same as multitasking, however in multiprocessing more than one CPUs are involved. On the other hand one CPU is involved in multitasking.

How to Create a thread in Java:

There are two ways to create a thread in Java:

1) By extending Thread class.

2) By implementing Runnable interface.

First we will discuss about Thread class.

Thread class:

Thread class provides constructors and methods to create and perform operations on a thread.Thread class extends Object class and implements Runnable interface.

Some common methods of Thread class:

Method	Meaning
public void run():	is used to perform action for a thread.
public void start():	starts the execution of the thread.JVM calls the run() method on the thread.
public void sleep(long miliseconds):	Causes the currently executing thread to sleep (temporarily cease execution) for the specified number of milliseconds.
public void join():	waits for a thread to die.
public void join(long miliseconds):	waits for a thread to die for the specified miliseconds.
public int getPriority():	returns the priority of the thread.
public int setPriority(int priority):	changes the priority of the thread.
public String getName():	returns the name of the thread.
public void setName(String name):	changes the name of the thread.
public Thread currentThread():	returns the reference of currently executing thread.
public int getId():	returns the id of the thread.
public Thread.State getState():	returns the state of the thread.
public boolean isAlive():	tests if the thread is alive.
public void suspend():	is used to suspend the thread
public void resume():	is used to resume the suspended thread
public void stop():	is used to stop the thread
public boolean isDaemon():	tests if the thread is a daemon thread.
public void setDaemon(boolean b):	marks the thread as daemon or user thread.

1) Creating a Thread By extending Thread class

```
class Thread1 extendsThread{
public void run()
{
System.out.println("running thread..");
}
public static void main(String args[]){
Thread t1=new Thread1();
t1.start();
 }
}
```

A task is always represented by run method

The start method starts a particular thread

Explaination:

public void run() method is used to perform action for a thread.Any task which is to be processed is put inside the run method.we can say that run() method is a task itself.

start() method of Thread class is used to start a newly created thread.The start() method makes the thread moves from New state to the Runnable state.A method is always started and its run() method gets called.

Exercise:

1. How to create thread by extending a Thread class?
2. What will happen if we use run() method instead of start() method to run a method of thread class.
3. What is the significance of run() and start() method of Thread class.