MULTITHREADING:

**package** multithreading14;

**public** **class** Multithreading **extends** Thread{

**public** **void** run(){

System.***out***.println("running thread name is:"+Thread.*currentThread*().getName()); // name of the thread

System.***out***.println("running thread priority is:"+ Thread.*currentThread*().getName()+ " :::" + Thread.*currentThread*().getPriority()); //what is current prirotiy of thread

System.***out***.println("running thread state is:"+ Thread.*currentThread*().getName()+ " :::" + Thread.*currentThread*().getState()); // it will print state of the thread running

System.***out***.println("The thread group is:" + Thread.*currentThread*().getName()+ " :::" + Thread.*currentThread*().getThreadGroup()); // a group in whihc thread is assgined

System.***out***.println("The thread id is:"+ Thread.*currentThread*().getName()+ " :::" + Thread.*currentThread*().getId()); // CPU scheduler has given a unique ID to each thread

System.***out***.println("Is my thread alive or not? : "+ Thread.*currentThread*().getName()+ " :::" + Thread.*currentThread*().isAlive()); // it will heck if thread is alive or dead

System.***out***.println("Is my thread Daemon or not? : "+ Thread.*currentThread*().getName()+ " :::" + Thread.*currentThread*().isDaemon()); // this thread always run in background

}

**public** **static** **void** main(String args[]){

Multithreading m1=**new** Multithreading();

Multithreading m2=**new** Multithreading();

Multithreading m3=**new** Multithreading();

m1.setName("Apq");

m2.setName("Pqr");

m3.setName("Hmn");

m1.setPriority(Thread.***MIN\_PRIORITY***);

m2.setPriority(Thread.***MAX\_PRIORITY***);

m3.setPriority(Thread.***NORM\_PRIORITY***);

m2.setDaemon(**true**);

m1.start();

m2.start();

m3.start();

}

}