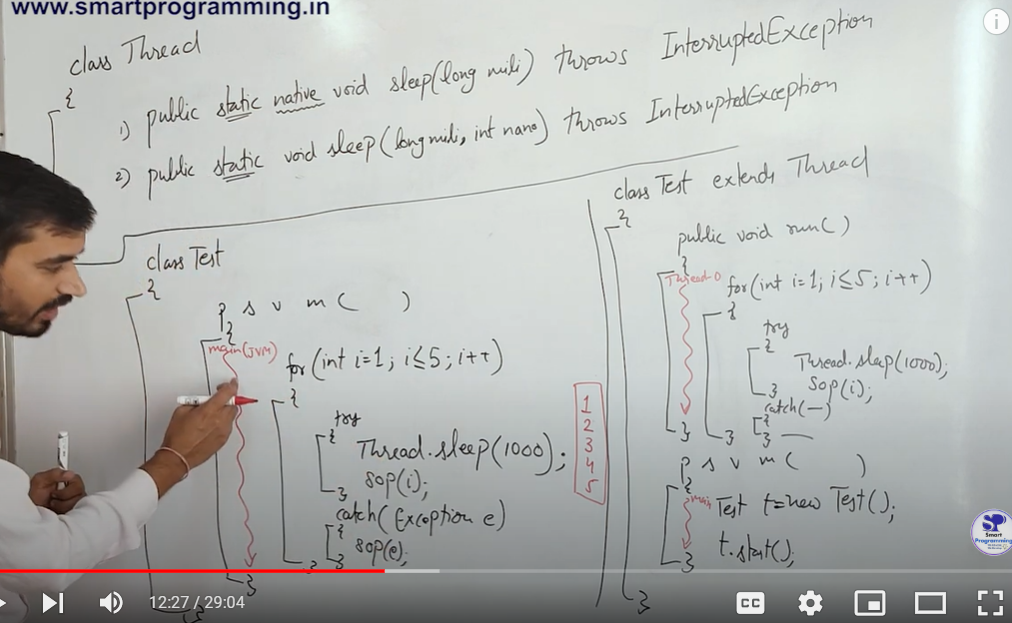
THREAD SLEEP METHODS



Two methods>

1)Public Static native void sleep( long mills) throws InterruptedExcepion

2) 1)Public Static void sleep( long mills , int nano ) throws InterruptedExcepion

-> so both are static methods and hence can be accessed using Thread Class .One is native that means its implementation is provided in other language and other one is implemented by JAVA. Both throws Interrupted Exception . So we need to handle the exception .

3) Use of nano in 2nd method ->

Example : Sleep for 1 millisecond and 500,000 nanoseconds (which equals 1.5 milliseconds)

->Thread.sleep(1, 500000); -> The accuracy of the sleep time for nanoseconds is highly platform-dependent.

q> How to handle exception ?

run() method:

It is part of the Thread class. . When you override run(), you cannot add throws InterruptedException because the original run() method in Thread class does not declare it. You must handle InterruptedException using a try-catch block within the run() method.

main() method:

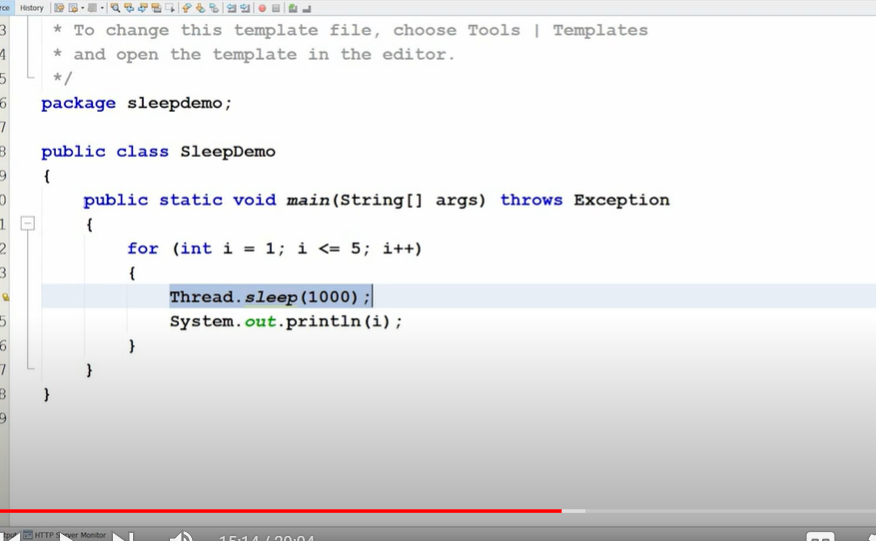
It is the entry point of your program, and it is not overriding any method.You can freely declare throws InterruptedException (or any other checked exception) because there are no restrictions on checked exceptions in the main() method.

Q>Thread.sleep(x); which thread ?? Current running thread that is if a thread.sleep(x); is used inside run() method , the sleep method will be called for run method();

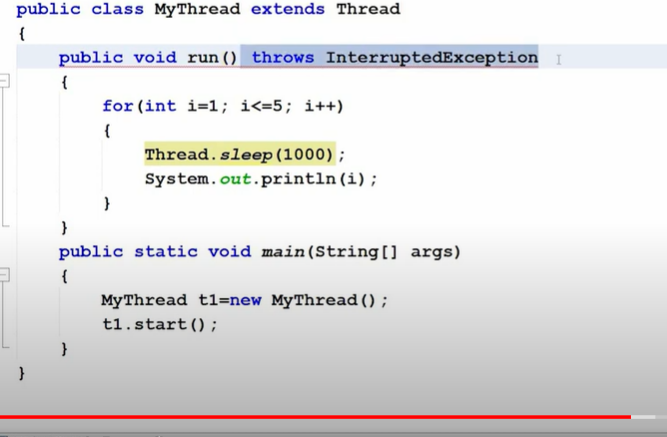
q>we can use try , catch block or throws exception or throws InterruptedException to handle exception.

Exception indicates that the method may throw any checked exception, including InterruptedException, IOException, SQLException, etc . InterruptedException means that the method specifically throws anInterruptedException, which occurs when a thread is interrupted while it is sleeping, waiting, or doing other blocking operations. So Exception Covers all checked exceptions, making it less specific and harder to understand which exact exception is being thrown.

A>>

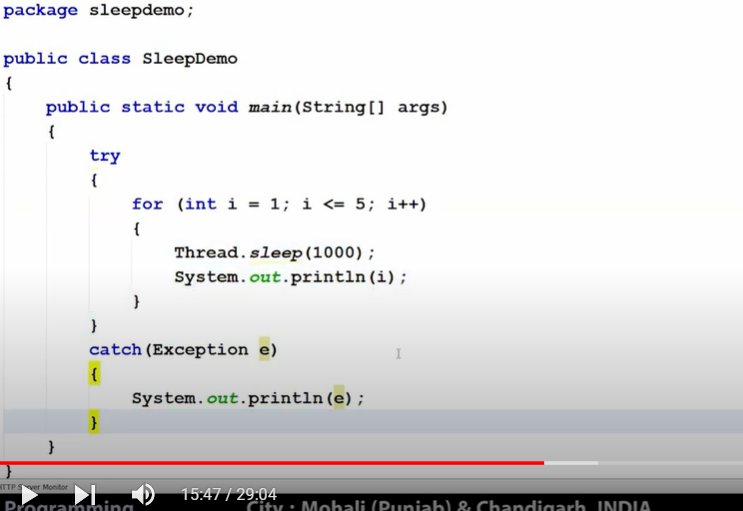


B>>

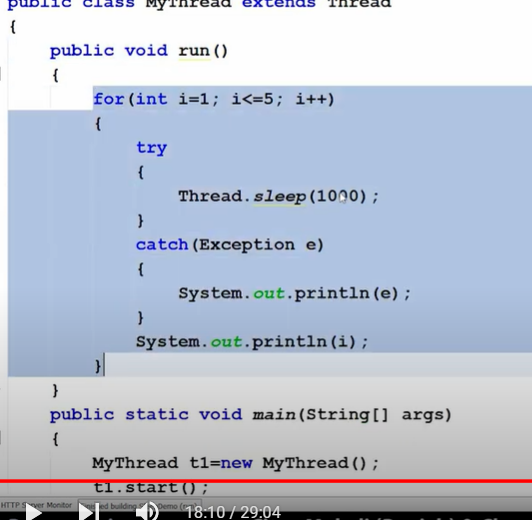


Code a no error : RSN -> we are overriding run() method from Thread Class and in Thread class throws InterruptedException is not declared in run() method.

c>>Using Try Catch Block



D>>Try Catch inside run method();



q>>

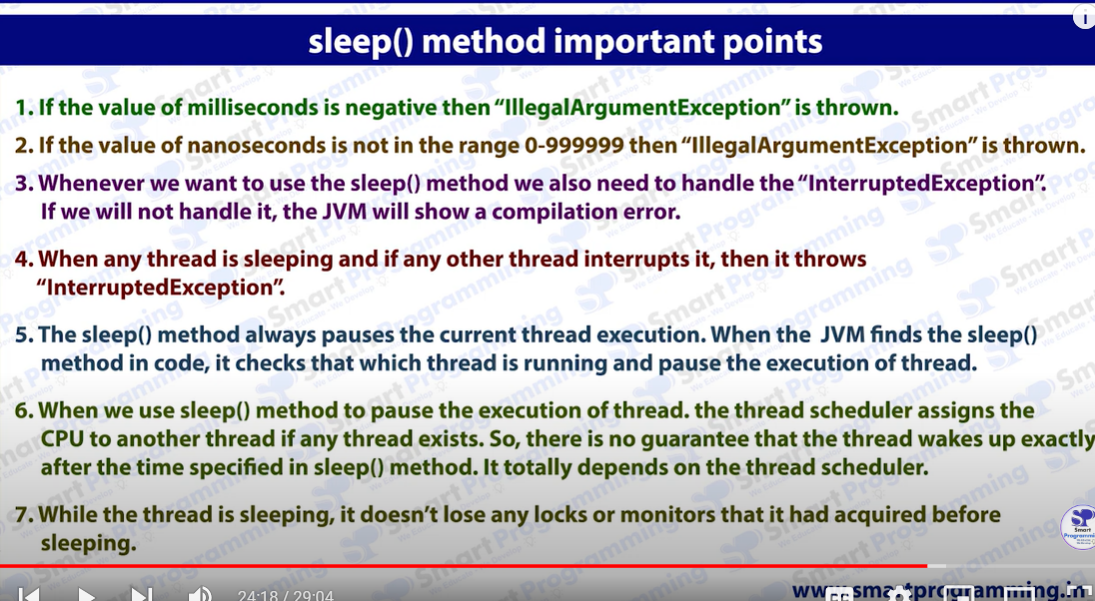
Thread.sleep(-1);

Error: timeout value is negative

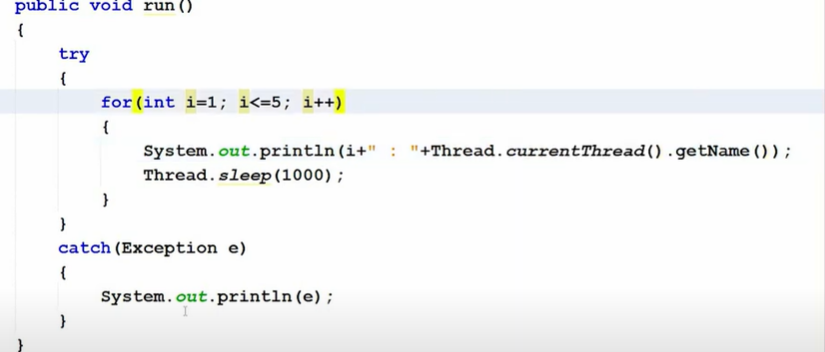
Thread.sleep(100,999999999999999);

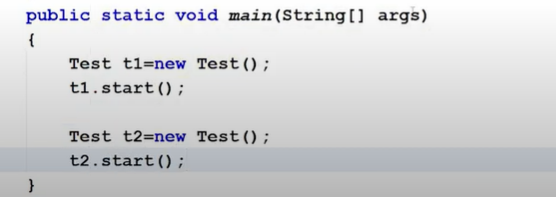
Error : nanosecond timeout value out of range

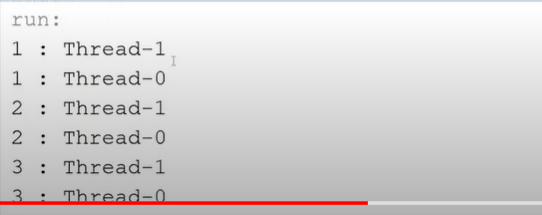
E>>



F>>





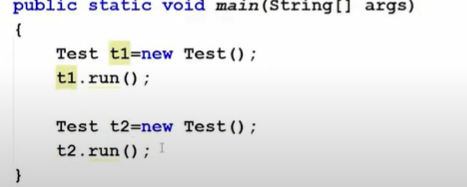


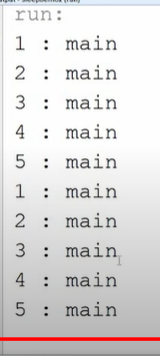
EVERY CONSECUTIVE 2 LINE IN THIS EXAMPL E PRINTS TOGETHER SINCE BOTH THREAD STARTS TOGETHER ( MILLISECOND DIFFERENCE) AND BOTH TAKING SLEEP CONSECUTIVELY FOR 1 SEC.OUPUT LOOKS LIKE

T= 0 : 1:Thread-1 1: Thread-0

T = 1: 1:Thread-1 1: Thread-0 2:Thread-1 2: Thread-0

G>>





SAME RUN() METHOD OF F , JUST START REPLACED WITH RUN()

* CHANGES :: THREAD.CURRENTTHREAD.GETNAME() – MAIN ?
* SINCE RUN() USED IT WILL BE TRAETED AS REGULAR FUNCTION AND HENCE THE RUN() METHOD WILL BE INVOKED MY MAIN THREAD.
* SINCE REGULAR FUNCION FIRST T1.RUN() WILL PRINT 1-5 THEN T2.RUN() WILL PRINT 1-5.

USING SLEEP() ,START() METHOD WE CAN RUN THINGS CONCURRENTLY LIKE CODE E . SO IF WE HAVE A VIDEO PLAYER WE CAN SYNCHRONIZE VIDEO , MUSIC , PROGRESS BAR AND RUN THEM PARALLELY.