



Subject: SBL-OOPJ

Class: SE-Data Science

Semester: III

A.Y. 2022-2023

Experiment No. 11

❖ **Aim :** Write a Java program to create thread and exploring built-in methods for it.

Theory :

There are two ways to create a thread:

1. By extending Thread class
2. By implementing Runnable interface.

Thread class:

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

Commonly used methods of Thread class:

1. **public void run():** is used to perform action for a thread.
2. **public void start():** starts the execution of the thread. JVM calls the run() method on the thread.
3. **public void sleep(long milliseconds):** Causes the currently executing thread to sleep (temporarily cease execution) for the specified number of milliseconds.
4. **public void join():** waits for a thread to die.
5. **public void join(long milliseconds):** waits for a thread to die for the specified milliseconds.
6. **public int getPriority():** returns the priority of the thread.
7. **public int setPriority(int priority):** changes the priority of the thread.
8. **public String getName():** returns the name of the thread.
9. **public void setName(String name):** changes the name of the thread.
10. **public Thread currentThread():** returns the reference of currently executing thread.
11. **public int getId():** returns the id of the thread.
12. **public Thread.State getState():** returns the state of the thread.
13. **public boolean isAlive():** tests if the thread is alive.
14. **public void yield():** causes the currently executing thread object to temporarily pause and allow other threads to execute.
15. **public void suspend():** is used to suspend the thread(deprecated).
16. **public void resume():** is used to resume the suspended thread(deprecated).



-
17. **public void stop();** is used to stop the thread(deprecated).
 18. **public boolean isDaemon();** tests if the thread is a daemon thread.
 19. **public void setDaemon(boolean b);** marks the thread as daemon or user thread.
 20. **public void interrupt();** interrupts the thread.
 21. **public boolean isInterrupted();** tests if the thread has been interrupted.
 22. **public static boolean interrupted();** tests if the current thread has been interrupted.

Java Thread Example by extending Thread class

FileName: Multi.java

```
1. class Multi extends Thread{
2.     public void run(){
3.         System.out.println("thread is running...");
4.     }
5.     public static void main(String args[]){
6.         Multi t1=new Multi();
7.         t1.start();
8.     }
9. }
```

Output:

thread is running...

Using the Thread Class: Thread(String Name)

We can directly use the Thread class to spawn new threads using the constructors defined above.

FileName: MyThread1.java

```
1. public class MyThread1
2. {
3.     // Main method
4.     public static void main(String args[])
5.     {
6.         // creating an object of the Thread class using the constructor Thread(String name)
7.         Thread t= new Thread("My first thread");
```



PARSHVANATH CHARITABLE TRUST'S

A.P. SHAH INSTITUTE OF TECHNOLOGY

Department of Computer Science and Engineering
Data Science

- 8.
9. // the start() method moves the thread to the active state
10. t.start();
11. // getting the thread name by invoking the getName() method
12. String str = t.getName();
13. System.out.println(str);
14. }
15. }

Output:

My first thread