

# Assignment\_10

## CODE:

```
package Com.College;
// Name: Aniket Singh
// PRN: 21070126013
// Batch: AIML-A1
/*
The above program demonstrates the usage of the Thread class and its various methods in Java. Specifically, the program creates a new thread using the Runnable interface, sets its priority, interrupts it, and waits for it to complete using join(). The program also shows how to sleep a thread and yield the main thread. The problem statement for this program could be to understand the basics of multithreading in Java and how to use the Thread class and its methods to manage threads effectively.
*/

import java.lang.Thread;
import java.lang.Runnable;

public class Assignment_10 implements Runnable {
    public static void main(String[] args) {
        // Create a new As_10 object
        Assignment_10 example = new Assignment_10();

        // Create a new thread with the Runnable object
        Thread thread = new Thread(example);

        // Start the thread
        thread.start();

        // Call various Thread methods
        System.out.println("-> Main thread starting...");
        System.out.println("-> New thread created: " + thread.getName());
        System.out.println("-> New thread ID: " + thread.getId());
        System.out.println("-> New thread priority: " + thread.getPriority());

        // Sleep for 1 seconds
        try {
            System.out.println("-> Main thread is sleeping...");
            Thread.sleep(1000);
            System.out.println("-> Main thread is awake!");
        } catch (InterruptedException e) {
            e.printStackTrace();
        }

        // Interrupt the new thread
        thread.interrupt();
        System.out.println("-> New thread interrupted: " + thread.isInterrupted());

        // Set thread priority to 7
        thread.setPriority(7);
        System.out.println("-> New thread priority (after setting): " + thread.getPriority());

        // Yield the main thread
        Thread.yield();
        System.out.println("-> Main thread yielded...");
    }
}
```

```

        try {
            // Wait for the thread to complete
            thread.join();
            System.out.println("-> New thread completed.");
        } catch (InterruptedException e) {
            e.printStackTrace();
        }

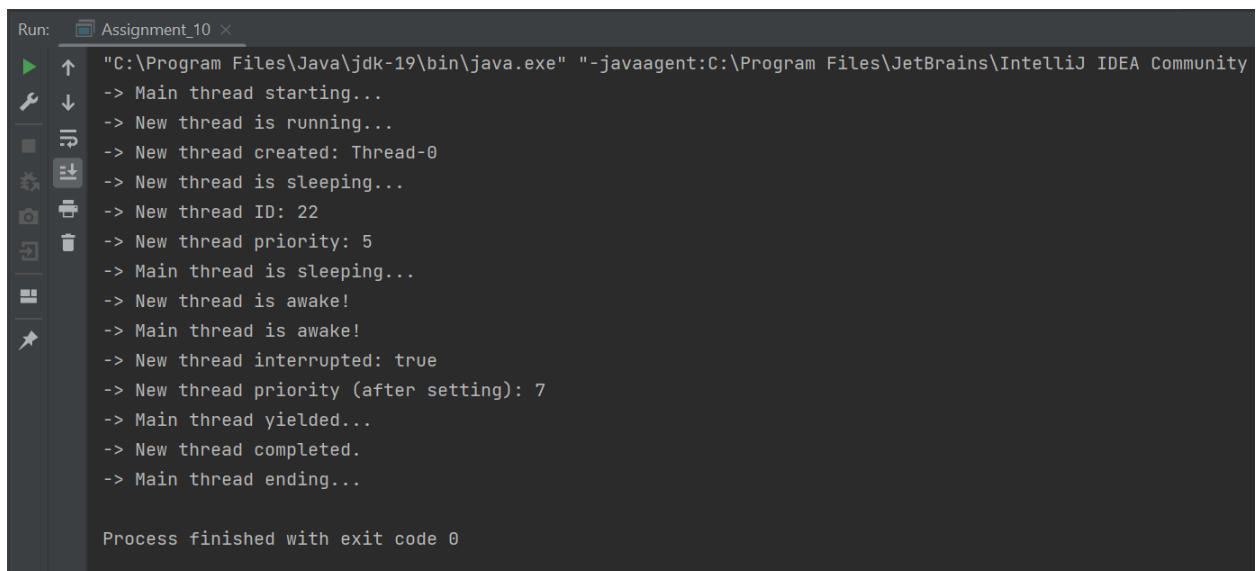
        System.out.println("-> Main thread ending...");
    }

    @Override
    public void run() {
        // The code that runs in the new thread
        System.out.println("-> New thread is running...");

        // Sleep for 1 seconds
        try {
            System.out.println("-> New thread is sleeping...");
            Thread.sleep(1000);
            System.out.println("-> New thread is awake!");
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}

```

## OUTPUT:



The screenshot shows the Run console in IntelliJ IDEA for a project named "Assignment\_10". The console displays the output of the Java program, which includes the main thread starting, creating a new thread, the new thread sleeping and waking up, the main thread sleeping and waking up, the new thread being interrupted, the main thread yielding, the new thread completing, and the main thread ending. The process finished with exit code 0.

```

Run: Assignment_10 x
"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community
-> Main thread starting...
-> New thread is running...
-> New thread created: Thread-0
-> New thread is sleeping...
-> New thread ID: 22
-> New thread priority: 5
-> Main thread is sleeping...
-> New thread is awake!
-> Main thread is awake!
-> New thread interrupted: true
-> New thread priority (after setting): 7
-> Main thread yielded...
-> New thread completed.
-> Main thread ending...

Process finished with exit code 0

```


## GITHUB:





java\_Assignments/Assignment\_10 at main · AniketSingh1m/java\_Assignments

Contribute to AniketSingh1m/java\_Assignments development by creating an account on GitHub.

AniketSingh1m/  
**java\_Assignments**



 [https://github.com/AniketSingh1m/java\\_Assignments/tree/main/Assignment\\_10](https://github.com/AniketSingh1m/java_Assignments/tree/main/Assignment_10)

 1 Contributor  
 0 Issues  
 0 Stars  
 1 Fork