2022-2026-CSE-B

Aim:

Daemon thread is a low priority thread (in the context of **JVM**) that runs in background to perform tasks such as garbage collection etc., they do not prevent the **JVM** from exiting when all the user threads finish their execution.

JVM terminates itself when all user threads finish their execution, even while Daemon threads are running.

The code Thread.currentThread().isDaemon() will return true, if the current thread is a daemon thread and false if it is not a daemon thread.

Write a Java program to illustrate daemon threads.

Write a class DeamonThreadDemo which extends the Thread class. Override its run() method to check whether the current thread is either daemon or user thread and print "This is daemon thread" and "This is not a daemon thread" respectively.

Write the **main()** method in the class DeamonThreadDemo, which create three instances of class DaemonThreadDemo as t1, t2 and t3 and perform the below tasks in the given order:

- 1. invoke the setDaemon() method on t1 instance and pass true as the argument to set t1 as a
 daemon thread.
- 2. Invoke start() method on t1, t2 and t3 respectively.

Note: Please don't change the package name.

Source Code:

q11351/DaemonThreadDemo.java

```
package q11351;
public class DaemonThreadDemo extends Thread {
  public void run() {
      if (Thread.currentThread().isDaemon() ) {
         System.out.println("This is daemon thread");
      } else {
         System.out.println("This is not a daemon thread");
      }
   }
  public static void main(String[] args) {
      DaemonThreadDemo t1 = new DaemonThreadDemo();
      DaemonThreadDemo t2 = new DaemonThreadDemo();
      DaemonThreadDemo t3 = new DaemonThreadDemo();
      t1.setDaemon(true);
      t1.start();
      t2.start();
      t3.start();
   }
}
```

C	V	
ċ	ċ	
2	Z	
9	פ	
ò	ฮ	,
Ω		

ID: 22K61A0587

Test Case - 1
User Output
This is daemon thread
This is not a daemon thread
This is not a daemon thread