Programming Assignment Unit 3

Godfrey Ouma

University of the People

CS 1103: Programming 2

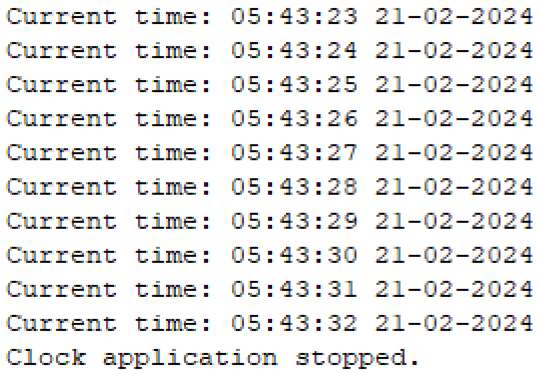
Vikas Thada

February 22, 2024

```
package textio;
import java.util.Calendar;
class ClockDemo implements Runnable {

    //Use volatile for visibility across threads
        private volatile boolean running = true;

        @Override
        public void run() {
            while (running) {
                try {
                    //Update time and print
                    Calendar now = Calendar.getInstance();
                    int hour = now.get(Calendar.HOUR_OF_DAY);
                    int minute = now.get(Calendar.MINUTE);
                    int second = now.get(Calendar.SECOND);
                    int day = now.get(Calendar.DAY_OF_MONTH);
                    int month = now.get(Calendar.MONTH)+1; //Months are 0-indexed
                    int year = now.get(Calendar.YEAR);
                    System.out.printf("Current time: %02d:%02d:%02d %02d-%02d-%04d\n",
hour, minute, second, day, month, year);

                    //Update every second
                    Thread.sleep(1000);

                    //Update every second
                    }
                catch (InterruptedException e) {
                    //Handle interruption gracefully
                    e.printStackTrace();
                    running = false;
                }
            }
        }

        public void stop() {
            running = false;
        }
    }
public class Clock{

    public static void main(String[] args) {
        //Create clock and display thread
        ClockDemo Obj = new ClockDemo();
        ThreadGroup clockGroup = new ThreadGroup("ClockGroup");
        Thread displayThread = new Thread(clockGroup,Obj, "DisplayThread");

        //Set higher priority
        displayThread.setPriority(Thread.MAX_PRIORITY);

        //Start threads
        displayThread.start();

        //Run for 10 seconds
        try {
            Thread.sleep(10000);
        } catch (InterruptedException e) {
            e.printStackTrace();
```

```java
        }
        // Stop clock thread
        Obj.stop();

        //Wait for display thread to finish
        try {
            displayThread.join();
        } catch (InterruptedException e) {
            e.printStackTrace();
        }

        System.out.println("Clock application stopped.");

    }

}
```

Console



```
Current time: 05:43:23 21-02-2024
Current time: 05:43:24 21-02-2024
Current time: 05:43:25 21-02-2024
Current time: 05:43:26 21-02-2024
Current time: 05:43:27 21-02-2024
Current time: 05:43:28 21-02-2024
Current time: 05:43:29 21-02-2024
Current time: 05:43:30 21-02-2024
Current time: 05:43:31 21-02-2024
Current time: 05:43:32 21-02-2024
Clock application stopped.
```