1. Problem statement: Write a simple Timer that can periodically print a timeout message.

2. code with comments:

```
package com.prac.go;


import java.util.Date;

import java.util.Timer;

import java.util.TimerTask;


public class TestMethod extends TimerTask {


  @Override
  public void run() {
    System.out.println("Timer task started at:"+new Date());
    completeTask();
    System.out.println("Timer task finished at:"+new Date());
  }


  private void completeTask() {
    try {
      //assuming it takes 20 secs to complete the task
      Thread.sleep(20000);
    } catch (InterruptedException e) {
      e.printStackTrace();
    }
  }


  public static void main(String args[]){
    TimerTask timerTask = new TestMethod();
    //running timer task as daemon thread
```

```java
    Timer timer = new Timer(true);

    timer.scheduleAtFixedRate(timerTask, 0, 10*1000); // Note: Java Timer object is scheduled to run
the task every 10 seconds

    System.out.println("TimerTask started");

    //cancel after sometime

    try {

        Thread.sleep(120000);

    } catch (InterruptedException e) {

        e.printStackTrace();

    }

    timer.cancel();

    System.out.println("TimerTask cancelled");

    try {

        Thread.sleep(30000);

    } catch (InterruptedException e) {

        e.printStackTrace();

    }

  }


}
```

3. Explanation of the code:

Running timer task as daemon thread. Java Timer object is scheduled to run the task every 10 seconds

cancel after sometime

4. Result flow in detail:

Running timer task as daemon thread. Java Timer object is scheduled to run the task every 10 seconds

cancel after sometime

5. Output screenshot:

```java
1  package com.prac.go;
2
3  import java.util.Date;
4  import java.util.Timer;
5  import java.util.TimerTask;
6
7  public class TestMethod extends TimerTask {
8
9      @Override
10     public void run() {
11         System.out.println("Timer task started at:"+new Date());
12         completeTask();
13         System.out.println("Timer task finished at:"+new Date());
14     }
15
16     private void completeTask() {
17         try {
18             //assuming it takes 20 secs to complete the task
19             Thread.sleep(20000);
20         } catch (InterruptedException e) {
21             e.printStackTrace();
22         }
23     }
24
25     public static void main(String args[]){
26         TimerTask timerTask = new TestMethod();
27         //running timer task as daemon thread
28         Timer timer = new Timer(true);
```

Problems @ Javadoc Declaration Console ⊠

TestMethod [Java Application] C:\Program Files\Java\jdk1.8.0_111\bin\javaw.exe (26-Nov-2018, 11:28:47 PM)

```
Timer task finished at:Mon Nov 26 23:29:08 IST 2018
Timer task started at:Mon Nov 26 23:29:08 IST 2018
Timer task finished at:Mon Nov 26 23:29:28 IST 2018
Timer task started at:Mon Nov 26 23:29:28 IST 2018
```