



CHAPTER 16

1. 만약 컴퓨터에 CPU가 하나뿐이라면 어떻게 여러 개의 스레드가 동시에 실행될 수 있는가?

2. 스레드를 생성하는 방법을 간단히 설명하고 각 방법의 장점과 단점을 설명하라.

(1) Thread 클래스를 상속받는 방법

(2) Runnable 인터페이스를 구현하는 방법

3. 스레드의 몸체를 포함하는 메소드는 무엇인가?

① run();

② start();

③ stop();

④ main();

4. 다음은 스레드를 생성하여 실행하는 코드이다. 비어 있는 부분에 어떤 코드를 넣어야 할까?

```
class Test implements Runnable
{
    public static void main(String args[])
    {
        /* 비어있는 부분 */
    }
    public void run() {}
}
```

5. 다음의 메소드는 어떠한 경우에 사용되는가?

(1) sleep()

(2) yield()

(3) start()

6. 다음 프로그램의 출력은?

```
public class Test implements Runnable
```

```

{
    public static void main(String [] args)
    {
        Test t = new Test();
        Thread x = new Thread(t);
        x.start();
    }
    public void run()
    {
        for(int i = 0; i < 3; ++i)
        {
            System.out.print(i + "..");
        }
    }
}

```

7. 다음 프로그램의 출력은?

```

public class Test {
    private int count = 1;

    public synchronized void sub() {
        for (int i = 0; i < 10; i++)
            System.out.println(count++);
    }

    public static void main(String[] args) {
        Test demo = new Test();
        Thread a1 = new A(demo);
        Thread a2 = new A(demo);
        a1.start();
        a2.start();
    }
}

class A extends Thread {
    Test demo;

    public A(Test td) {
        demo = td;
    }

    public void run() {
        demo.sub();
    }
}

```

8. (1) 출력을 예상하여 보라.

```
class Job implements Runnable {
    public void run() {
        byte n = 0;
        while(true)
            System.out.println(""+n++);
    }
}

public class Test {
    public static void main(String[] args) {
        Thread t = new Thread(new Job());
        t.start();
    }
}
```

(2) Thread 클래스를 상속받는 형태로 다시 작성하라.

(3) Job 클래스의 생성자를 작성하고 매개 변수로 스레드의 이름을 문자열 형태로 받도록 하라. 스레드가 실행되면서 스레드의 이름을 출력한다.

```
System.out.println(name+n++);
```

(4) 동일한 스레드를 하나 더 생성하여 동시에 실행하여 보자. 어떤 출력이 나타나는가? 출력을 구분하기 위하여 스레드의 이름을 다르게 하라.

9. 다음 프로그램의 출력을 예상하여 보라.

```
class MyJob implements Runnable {
    public void run() {
        try {
            System.out.println("A");
            Thread.sleep(1000);
            System.out.println("B");
        } catch (InterruptedException e) {
            System.out.println("C");
        }
        System.out.println("D");
    }
}

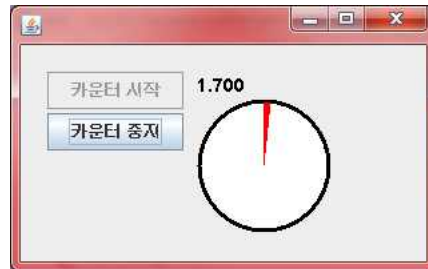
public class Test {
    public static void main(String[] args) {
        Thread t = new Thread(new MyJob());
        t.start();
    }
}
```

```

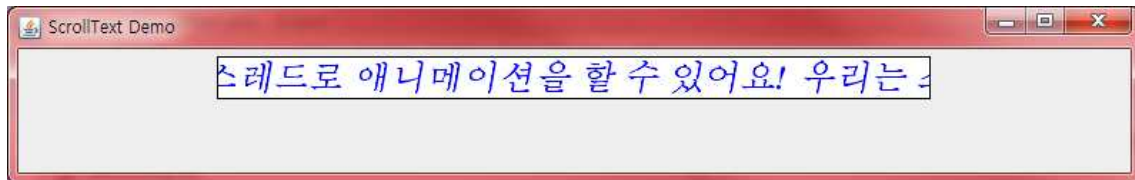
        t.interrupt();
    }
}

```

10. "Start" 버튼을 누르면 스레드가 동작하여서 카운터를 증가시키고 "Stop" 버튼을 클릭하면 카운터 증가가 중지되는 애플리케이션을 작성하여 보자. 스레드를 사용한다.



11. 거리의 광고판을 보면 글자들이 가로로 스크롤되는 것을 많이 볼 수 있다. 자바의 스레드 기능을 이용하여서 글자들이 스크롤되도록 프로그램을 작성하라.



12. 화면의 사각형이 축소되었다가 다시 확대되는 반복적인 애니메이션을 작성하여 보자. 역시 스레드를 이용한다.

