120 Java의 정석定石 3판 - 연습문제 풀이

# Chapter

쓰레드

thread

## [ 연습문제 ]

[13-1] 쓰레드를 구현하는 방법에는

Thread클래스로부터 상속받는 것과

Runnable인터페

이스를 구현하는 것 두 가지가 있는데, 다음의 코드는 Thread클래스를 상속받아서 쓰레드 를 구현한 것이다. 이 코드를 Runnable인터페이스를 구현하도록 변경하시오.

class Exercise13\_1 {

public static void main(String args[]) { Thread1 th1 = new Thread1();

th1.start();

}

}

class Thread1 extends Thread { public void run() {

for(int i=0; i < 300; i++) { System.out.print('-');

}

}

}

[연습문제]**/ch13/Exercise13\_1.java**

[13-2] 다음 코드의 실행결과로 옳은 것은?

class Exercise13\_2 {

public static void main(String[] args) { Thread2 t1 = new Thread2(); t1.run();

for(int i=0; i < 10; i++) System.out.print(i);

}

}

class Thread2 extends Thread { public void run() {

for(int i=0; i < 10; i++) System.out.print(i);

}

}

[연습문제]**/ch13/Exercise13\_2.java**

1. 01021233454567689789처럼 0부터 9까지의 숫자가 섞여서 출력된다.
2. 01234567890123456789처럼 0부터 9까지의 숫자가 순서대로 출력된다.
3. IllegalThreadStateException이 발생한다.

[13-3] 다음 중 쓰레드를 일시정지 상태(WAITING)로 만드는 것이 아닌 것은? (모두 고르 시오)

1. suspend()
2. resume()
3. join()
4. sleep()
5. wait()
6. notify()

[13-4] 다음 중 (모두 고르시오)

interrupt()에 의해서 실행대기 상태(RUNNABLE)가 되지 않는 경우는?

1. sleep()에 의해서 일시정지 상태인 쓰레드
2. join()에 의해서 일시정지 상태인 쓰레드
3. wait()에 의해서 일시정지 상태인 쓰레드
4. suspend()에 의해서 일시정지 상태인 쓰레드

[13-5] 다음의 코드를 실행한 결과를 예측하고,

직접 실행한 결과와 비교하라.

만일 예

측한 결과와 실행한 결과의 차이가 있다면 그 이유를 설명하라.

class Exercise13\_5

{

public static void main(String[] args) throws Exception

{

Thread3 th1 = new Thread3(); th1.start();

try {

Thread.sleep(5\*1000);

} catch(Exception e) {}

throw new Exception("꽝~!!!");

}

}

class Thread3 extends Thread { public void run() {

for(int i=0; i < 10; i++) { System.out.println(i);

try {

Thread.sleep(1000);

} catch(Exception e) {}

}

} // run()

}

[연습문제]**/ch13/Exercise13\_5.java**

[13-6] 다음의 코드를 실행한 결과를 예측하고,

직접 실행한 결과와 비교하라.

만일 예

측한 결과와 실행한 결과의 차이가 있다면 그 이유를 설명하라.

class Exercise13\_6

{

public static void main(String[] args) throws Exception

{

Thread4 th1 = new Thread4(); th1.setDaemon(true); th1.start();

try {

th1.sleep(5\*1000);

} catch(Exception e) {}

throw new Exception("꽝~!!!");

}

}

class Thread4 extends Thread { public void run() {

for(int i=0; i < 10; i++) { System.out.println(i);

try {

Thread.sleep(1000);

} catch(Exception e) {}

}

} // run()

}

[연습문제]**/ch13/Exercise13\_6.java**

[13-7] 다음의 코드는 쓰레드 th1을 생성해서 실행시킨 다음 6초 후에 정지시키는 코드

이다. 그러나 실제로 실행시켜보면 쓰레드를 정지시킨 다음에도 몇 초가 지난 후에서야

멈춘다. 그 이유를 설명하고, 선하시오.

쓰레드를 정지시키면 지체없이 바로 정지되도록 코드를 개

class Exercise13\_7

{

static boolean stopped = false;

public static void main(String[] args)

{

Thread5 th1 = new Thread5(); th1.start();

try {

Thread.sleep(6\*1000);

} catch(Exception e) {}

stopped = true; // 쓰레드를 정지시킨다.

System.out.println("stopped");

}

}

class Thread5 extends Thread { public void run() {

// Exercise13\_7.stopped의 값이 false인 동안 반복한다. for(int i=0; !Exercise13\_7.stopped; i++) {

System.out.println(i);

try {

Thread.sleep(3\*1000);

} catch(Exception e) {}

}

} // run()

}

[연습문제]**/ch13/Exercise13\_7.java**

0

1

2

stopped

[실행결과]

[13-8] 다음의 코드는 텍스트기반의 타자연습게임인데 WordGenerator라는 쓰레드가

Vector에 2초마다 단어를 하나씩 추가하고, 사용자가 단어를 입력하면 Vector에서 일치하 는 단어를 삭제하도록 되어 있다. WordGenerator의 run()을 완성하시오.

import java.util.\*;

class Exercise13\_8

{

Vector words = new Vector();

String[] data = {"태연","유리","윤아","효연","수영","서현","티파니","써니","제시카"}; int interval = 2 \* 1000; // 2초

WordGenerator wg = new WordGenerator();

public static void main(String args[])

{

Exercise13\_9 game = new Exercise13\_9(); game.wg.start();

Vector words = game.words;

while(true) {

System.out.println(words);

String prompt = ">>"; System.out.print(prompt);

// 화면으로부터 라인단위로 입력받는다. Scanner s = new Scanner(System.in); String input = s.nextLine().trim();

int index = words.indexOf(input); if(index!=-1) {

words.remove(index);

}

}

} // main

class WordGenerator extends Thread { public void run() {

**/\***

**(1)** 아래의 로직에 맞게 코드를 작성하시오**.**

1. **interval(2**초**)**마다 배열 **data**의 값 중 하나를 임의로 선택해서
2. **words**에 저장한다**.**

**\*/**

} // end of run()

} // class WordGenerator

} // Exercise13\_9

[연습문제]**/ch13/Exercise13\_8.java**

[]

>>

[서현]

>>서현 [수영, 윤아]

>>수영 [윤아, 유리]

>>유리

[윤아, 티파니]

>>티파니

[윤아, 윤아, 유리]

>>윤아 [윤아, 유리]

>>유리 [윤아, 효연]

>>효연

[윤아, 티파니]

>>윤아 [티파니, 윤아]

>>티파니

[윤아, 수영, 써니]

>>

[실행결과]

[13-9] 다음은 사용자의 입력을 출력하고 종료하는 프로그램을 작성한 것으로, 10초 동

안 입력이 없으면 자동종료되어야 한다.

그러나 실행결과를 보면,

사용자의 입력이

10초

안에 이루어졌음에도 불구하고 프로그램이 즉시 종료되지 않는다. 사용자로부터 입력받는 즉시 프로그램이 종료되도록 수정하시오.

String input = JOptionPane.showInputDialog("아무 값이나 입력하세요.");

System.out.println("입력하신 값은 " + input + "입니다."); th1.interrupt(); // 쓰레드에게 작업을 멈추라고 요청한다.

}

}

class Exercise13\_9\_1 extends Thread { public void run() {

int i = 10;

while(i!=0 && !isInterrupted()) {

{

import javax.swing.JOptionPane;

class Exercise13\_9 {

public static void main(String[] args) throws Exception Exercise13\_9\_1 th1 = new Exercise13\_9\_1(); th1.start();

[연습문제]**/ch13/Exercise13\_9.java**

System.out.println(i--);

try {

Thread.sleep(1000); // 1초 지연

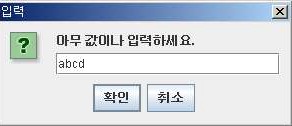
} catch(InterruptedException e) {}

}

System.out.println("카운트가 종료되었습니다.");

} // main

}



10

9

8

입력하신 값은 abcd입니다. 7

6

5

4

3

2

1

카운트가 종료되었습니다.

[실행결과]