



Scanner 클래스를 이용한 콘솔 입력

- 1) 표준 출력장치(콘솔)를 참조하기 위하여 **System.out** 클래스를 사용한다. `print()`, `println()`, `printf()` 메소드를 사용한다.
- 2) 표준 입력장치(키보드)를 참조하기 의해 **System.in** 클래스를 사용한다. 콘솔 입력은 자바에서 직접적으로 지원하지 않지만 **System.in**으로부터 입력을 읽기 위한 객체를 생성하는 **Scanner** 클래스를 사용한다.

```
Scanner input = new Scanner(System.in);
```

System.in

키보드로부터 직접 읽는 자바의 표준 입력 스트림

키 값을 바이트(문자 아님)로 리턴

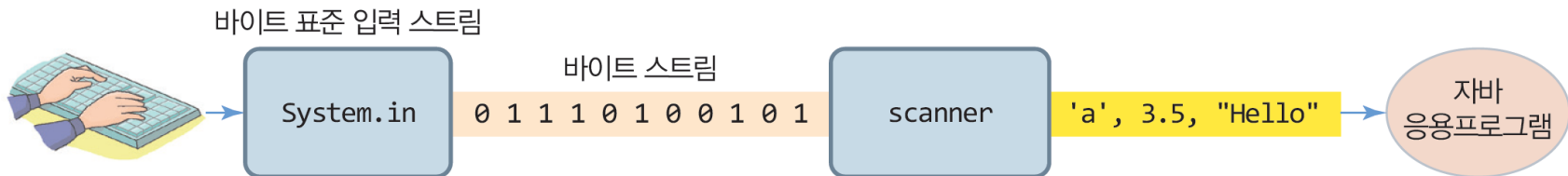
System.in을 사용할 때 문제점

키 값을 바이트 데이터로 넘겨주므로 응용프로그램이 문자 정보로 변환해야 함

Scanner 클래스를 이용한 콘솔 입력

Scanner 클래스

- System.in에게 키를 읽게 하고, 읽은 바이트를 문자, 정수, 실수, 불린, 문자열 등 다양한 타입으로 변환하여 리턴
 - java.util.Scanner 클래스



- Scanner에서 키 입력 받기
 - Scanner는 입력되는 키 값을 공백으로 구분되는 아이템 단위로 읽음
 - 공백 문자 : 'wt', 'wf', 'wr', ' ', 'wn'
- 개발자가 원하는 다양한 타입의 값으로 바꾸어 읽을 수 있음



Scanner 클래스를 이용한 콘솔 입력

1) Scanner 객체를 위한 메소드

| 메소드 | 설명 |
|--------------|----------------------------------------------------------------|
| nextByte() | byte 타입의 정수 하나를 읽는다. |
| nextShort() | short 타입의 정수 하나를 읽는다. |
| nextInt() | int 타입의 정수 하나를 읽는다. |
| nextLong() | long 타입의 정수 하나를 읽는다. |
| nextFloat() | float 타입의 실수 하나를 읽는다. |
| nextDouble() | double 타입의 실수 하나를 읽는다. |
| next() | 공백 이전에 종료되는 하나의 문자열을 읽는다. |
| nextLine() | 한 줄의 문자열을 읽는다. 즉 'Wn'을 포함하는 한 라인을 읽고 'Wn'을 버린 나머지 문자열 리턴한다. |
| close | Scanner의 사용 종료 |



Scanner 클래스를 이용한 콘솔 입력

1) import 문장

import java.util.Scanner; // Scanner 클래스 포함

- 입력 기능을 가지고 있는 클래스를 포함시키는 문장이다.
- 클래스들은 패키지라는 단위로 그룹핑되어 있다.
- 패키지들을 자바 클래스 라이브러리(Java Class Library)라고 부른다.

2) 객체 생성(추후 상세 설명)

Scanner input = new Scanner(System.in);



Scanner 클래스를 이용한 콘솔 입력

3) 사용자로부터 입력

`x = input.nextInt(); // 사용자로부터 첫 번째 숫자를 읽는다.`

Scanner 객체인 `input`을 이용하여 사용자로부터 정수를 읽어 들이는 문장이다. `input`을 통하여 `nextInt()`라고 하는 메소드를 호출하게 된다.

4) 출력메소드

`System.out.print("첫번째 숫자를 입력하시오: "); // 입력 안내 출력`

`println()`은 문자열을 출력한 후에 줄을 바꾸는 메소드이고
`print()`는 문자열 출력 후에 줄을 바꾸지 않는다.



소스 코드 입력

```
package exam_input;
```

```
// 사용자가 입력한 두 개의 숫자를 더해서 출력한다.  
import java.util.Scanner; // Scanner 클래스 포함
```

```
public class TwoNumberSum {  
    // 메인 메소드에서부터 실행이 시작된다.  
    public static void main(String args[]) {  
        Scanner input = new Scanner(System.in);  
        int x; // 첫 번째 숫자 저장  
        int y; // 두 번째 숫자 저장  
        int sum; // 합을 저장
```



소스 코드 입력

```
// 입력 안내 출력
```

```
System.out.print("첫번째 숫자를 입력하시오: ");
```

```
x = input.nextInt(); // 사용자로부터 첫 번째 숫자를 읽는다.
```

```
// 입력 안내 출력
```

```
System.out.print("두번째 숫자를 입력하시오: ");
```

```
y = input.nextInt(); // 사용자로부터 두 번째 숫자를 읽는다.
```

```
sum = x + y; // 두 개의 숫자를 더한다.
```

```
System.out.println("두 수의 합 : "+sum); // 합을 출력한다.
```

```
input.close();
```

```
}
```

```
}
```




Scanner를 이용한 키 입력 연습

Scanner를 이용하여
이름, 도시, 나이, 체중,
독신 여부를 입력 받고
다시 출력하는
프로그램을 작성하라.

```
import java.util.Scanner;

public class ScannerEx {
    public static void main(String args[]) {
        System.out.println("이름, 도시, 나이, 체중, 독신 여부를 빈칸으로 분리하여 입력하세요");
        Scanner scanner = new Scanner(System.in);

        String name = scanner.next(); // 문자열 읽기
        System.out.print("이름은 " + name + ", ");

        String city = scanner.next(); // 문자열 읽기
        System.out.print("도시는 " + city + ", ");

        int age = scanner.nextInt(); // 정수 읽기
        System.out.print("나이는 " + age + "살, ");

        double weight = scanner.nextDouble(); // 실수 읽기
        System.out.print("체중은 " + weight + "kg, ");

        boolean single = scanner.nextBoolean(); // 논리값 읽기
        System.out.println("독신 여부는 " + single + "입니다.");

        scanner.close(); // scanner 닫기
    }
}
```

이름, 도시, 나이, 체중, 독신 여부를 빈칸으로 분리하여 입력하세요.

Kim Seoul 20 65.1 true

이름은 Kim, 도시는 Seoul, 나이는 20살, 체중은 65.1kg, 독신 여부는 true입니다.

클래스 생성

File->New->Class 메뉴 선택

New Java Class

Java Class
Create a new Java class.

Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static

Superclass:

Interfaces:

Which method stubs would you like to create?

- ☐ public static void main(String[] args)
- ☐ Constructors from superclass
- ☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))

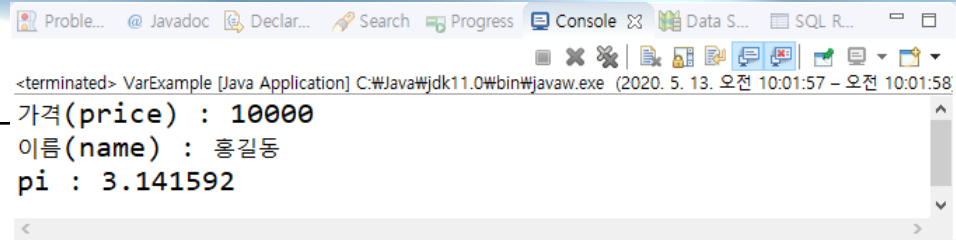
- ☐ Generate comments

소스 코드 입력

```
package exam_var;
```

```
public class VarExample {  
    public static void main(String[] args) {  
        var price = 10000;  
        var name = "홍길동";  
        var pi = 3.141592;
```

```
        System.out.println("가격(price) : " + price);  
        System.out.println("이름(name) : " + name);  
        System.out.println("pi : " + pi);  
    }  
}
```





- 자바의 기본 구조를 익히기 위해 자바의 입출력 방법이나 오류를 미리 익혀야 한다.
 - 자바의 입출력은 라이브러리 클래스를 이용합니다.
 - 현 시점에서 입출력과 관련된 라이브러리를 이해할 필요는 없습니다.
 - 입출력 방법을 상용 구문처럼 외워서 사용하면 됩니다.
 - 프로그램을 설명하는 주석문에 대해 학습합니다.
 - 프로그램 작성시 발생하는 오류에 대해 학습합니다.



● 실습 예제

예제 2.2

SecondProgram.java

```
01: public class SecondProgram {  
01: public static void main(String[] args) {  
02:     System.out.print("두 번째로 작성해 보는 ");  
03:     System.out.println("자바 프로그램입니다 ");  
04:     System.out.println("처음 시작하는 " + "자바 프로그램입니다");  
05: }  
06: }
```

◀-내용을 출력한 다음 줄을 바꾸지 않는다

실행 결과

두 번째로 작성해 보는 자바 프로그램입니다
처음 시작하는 자바 프로그램입니다



프로그램에 값을 입력하기 위한 방법

- 프로그램 실행 시 값을 입력할 수 있는 방법 중에서 Scanner 라이브러리 클래스를 이용한다
- 실스 예제

예제 2.3

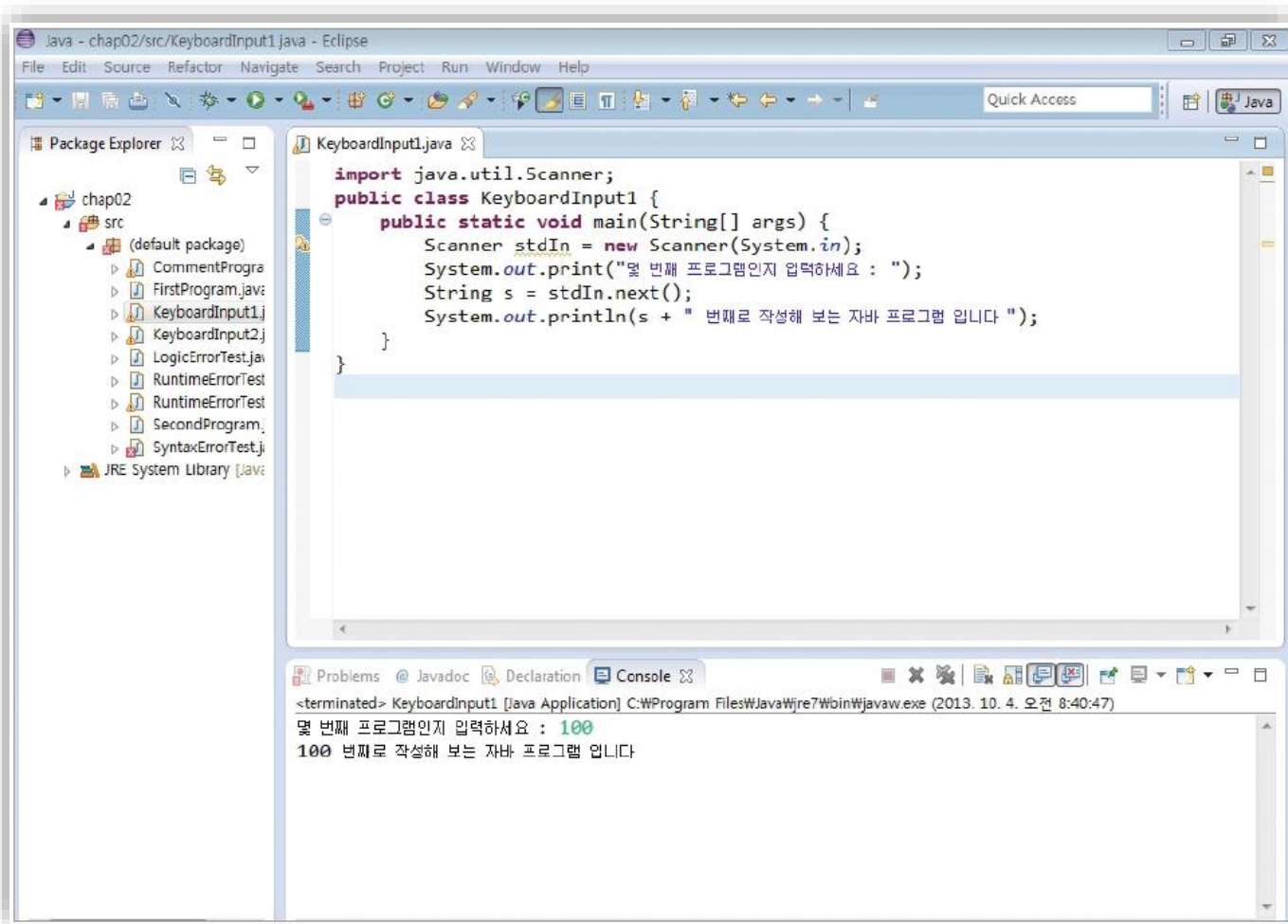
KeyboardInput1.java

```
01: import java.util.Scanner; ← 라이브러리 클래스 포함.
02: public class KeyboardInput1 {
03:     public static void main(String[] args) {
04:         Scanner stdIn = new Scanner(System.in); ← 키보드를 입력을 위해 Scanner 객체 생성
05:         System.out.print("몇 번째 프로그램인지 입력하세요 : ");
06:         String s = stdIn.next(); ← 사용자의 입력을 문자열로 받는다
07:         System.out.println(s + " 번째로 작성해 보는 자바 프로그램 입니다 ");
08:     }
09: }
```



프로그램에 값을 입력하기 위한 방법

- 프로그램을 실행시키고 Console 창에서 데이터를 입력한다





프로그램에 값을 입력하기 위한 방법

- 실습 예제 : Scanner 클래스의 nextInt(), nextDouble() 메소드 사용

예제 2.4

KeyboardInput2.java

```
01: import java.util.Scanner;
02: public class KeyboardInput2 {
03:     public static void main(String[] args) {
04:         Scanner stdIn = new Scanner(System.in);
05:         System.out.print("이름과 나이, 몸무게를 공간(space)으로 구분하여 입력 : ");
06:         String name = stdIn.next(); ←----- 키보드로부터 이름을 문자열로 입력
07:         int i = stdIn.nextInt(); ←----- 키보드로부터 나이를 정수로 입력
08:         double d = stdIn.nextDouble(); ←----- 키보드로부터 몸무게를 실수로 입력
09:         System.out.println(name + "씨의 나이는 " + i + "세이고");
10:         System.out.println("몸무게는 " + d + "Kg 입니다");
11:     }
12: }
```




프로그램에 값을 입력하기 위한 방법

- 실습 예제 : 실행 시 데이터 입력은 공간으로 구분한다

The screenshot shows the Eclipse IDE with a Java project named 'chap02'. The Package Explorer on the left shows the project structure. The main editor displays the code for 'KeyboardInput2.java'. The code imports 'java.util.Scanner' and defines a 'main' method. It uses 'Scanner' to read input from 'System.in', splitting it by spaces. It then reads a name, an integer age, and a double weight, and prints them out with formatted strings.

```
import java.util.Scanner;
public class KeyboardInput2 {
    public static void main(String[] args) {
        Scanner stdIn = new Scanner(System.in);
        System.out.print("이름과 나이, 몸무게를 공간(space)으로 구분하여 입력 : ");
        String name = stdIn.next();
        int i = stdIn.nextInt();
        double d = stdIn.nextDouble();
        System.out.println(name + "씨의 나이는 " + i + "세이고");
        System.out.println("몸무게는 " + d + "Kg 입니다");
    }
}
```

The Console window at the bottom shows the output of the program. It displays the prompt '이름과 나이, 몸무게를 공간(space)으로 구분하여 입력 : ' followed by the user input '홍길동 22 62.5'. The program then outputs '홍길동씨의 나이는 22세이고' and '몸무게는 62.5Kg 입니다'.



주석문과 프로그램 오류

● 주석문

- 프로그램의 구조와 동작 방법을 설명하기 위한 부분
- 프로그램의 실행에는 영향을 미치지 않는다
- 주석문은 향후에 다른 개발자가 프로그램을 수정하거나 보완할 때 매우 유용하다

자바의 주석문은 다음과 같이 3가지 형태로 많이 사용됩니다.

- ① `/* */` : 여러 줄의 문장을 주석으로 처리할 수 있습니다.
- ② `// :` 한 줄의 문장을 주석으로 처리합니다.
- ③ `/** */` : 여러 줄의 문장을 주석으로 처리하고, javadoc 도구에 의해 사양서를 만들 수 있는 주석입니다.



주석문과 프로그램 오류

● 예제 : 주석문을 사용한 프로그램

예제 2.5

CommentProgram.java

```
01: /**
02:  * 이 클래스는 세 개의 값을 입력받아 첫 번째 값은 문자열값으로, 두 번째 값은
    정수값으로
03:  * 세 번째 값은 실수값으로 출력하는 프로그램입니다.
04:  * @author 김충석
05:  * @see KeyboardInput2.java
06:  */
07: import java.util.Scanner;    // Scanner 클래스를 포함
08: public class CommentProgram {
09:     public static void main(String[] args) {
10:         Scanner stdIn = new Scanner(System.in);
11:         // 다음 문장은 콘솔 창에서 입력을 요청하는 문장입니다
12:         System.out.print("이름과 나이, 몸무게를 공간(space)으로 구분하여 입력 : ");
13:         /* 표준 입력으로 문자열과 정수,
14:          * 실수를 입력받아 적합한 형의 변수에 값을 저장 */
15:         String name = stdIn.next();
16:         int i = stdIn.nextInt();
17:         double d = stdIn.nextDouble();
18:         System.out.println(name + "씨의 나이는 " + i + "세이고");
19:         System.out.println("몸무게는 " + d + "Kg 입니다"); // 저장된 값을 출력
20:     }
21: }
```



주석문과 프로그램 오류

- 프로그래밍 오류

- 능숙한 프로그래머도 오류를 피해갈 수는 없다

- 오류의 종류

- 컴파일 시간에 발생하는 **구문(Syntax) 오류** : 이클립스에 의해 오류가 자동 표시된다
- 실행 시 발생하는 **실행 시간(Runtime) 오류** : 프로그램 실행 시 JVM에 의해 오류를 발생시킨다
- 논리적 문제로 발생하는 **논리(Logic) 오류** : 프로그래머에 의해 발견되어야 수정할 수 있다



주석문과 프로그램 오류

● 프로그래밍 오류 : 구문 오류 - 예제

예제 2.6

SyntaxErrorTest.java

```
01: public class SyntaxErrorTest {  
02:     public static void main(String args[]) {  
03:         i = 30 ; ← 변수를 선언하지 않고 사용  
04:         System.out.println( i ) ← 문장 종료 기호가 없음  
05:     }  
06: }
```

실행 결과 오류 메시지 출력

Exception in thread "main" java.lang.Error: Unresolved compilation problems:

i cannot be resolved to a variable

i cannot be resolved to a variable

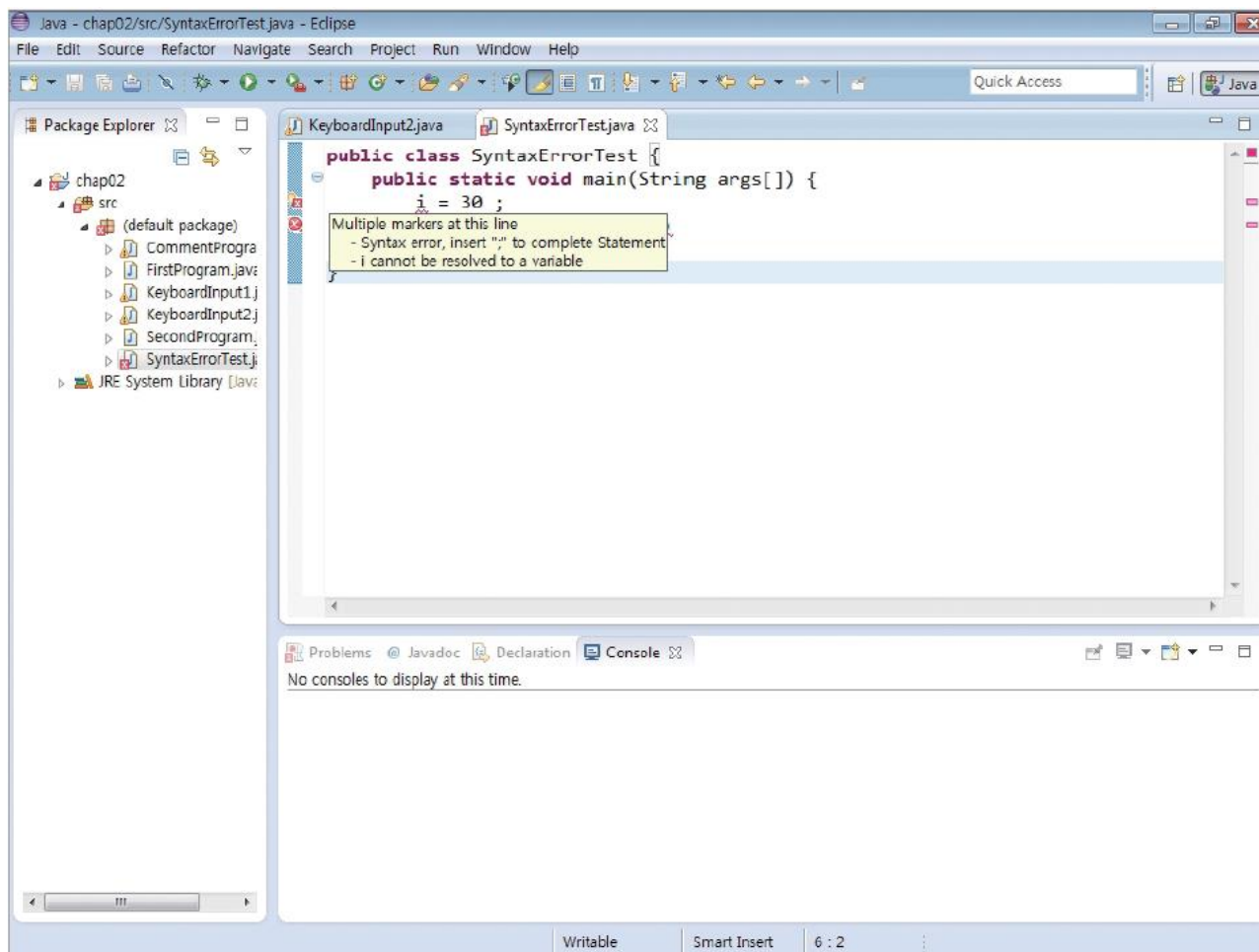
Syntax error, insert ";" to complete Statement

at SyntaxErrorTest.main(SyntaxErrorTest.java:3)



주석문과 프로그램 오류

- 프로그래밍 오류 : 구문 오류(구문 오류는 이클립스에 적색으로 표시된다)





주석문과 프로그램 오류

● 프로그래밍 오류

- 0으로 나누거나, 배열의 첨자가 벗어났거나, 사용자의 입력 값이 잘못 입력 되었거나 등등
- 실행 시간 오류는 프로그램을 비 정상적으로 종료 시킨다

예제 2.7

RuntimeErrorTest.java

```
01: public class RuntimeErrorTest {  
02:     public static void main(String args[]) {  
03:         int i = 3 / 0 ;  
04:         int j = 5 / 0 ;  
05:         System.out.println(i + j);  
06:     }  
07: }
```




주석문과 프로그램 오류

● 실행 시간 오류 메시지 출력

- 3번 라인에서 실행 시간 오류 발생하여 프로그램 종료되고 아래와 같은 메시지 출력

실행 결과

실행 시간 오류 메시지 출력

```
Exception in thread "main" java.lang.ArithmeticException: / by zero  
at RuntimeErrorTest.main(RuntimeErrorTest.java:3)
```

- 3번 라인을 수정한 다음 다시 실행시키면 다시 4번 라인에서 실행 시간 오류 메시지 출력

실행 결과

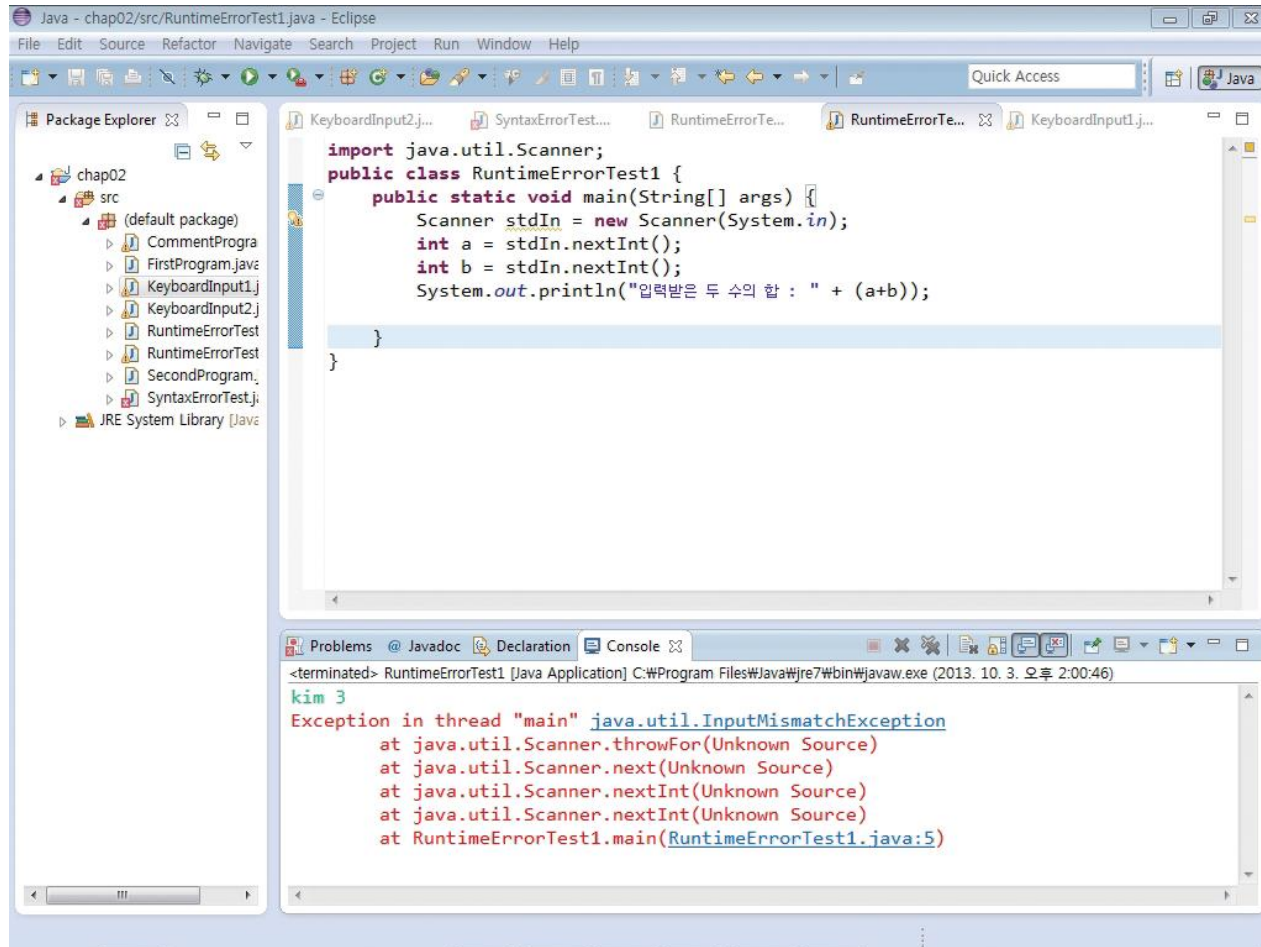
3번 라인의 오류를 수정한 다음 실행

```
Exception in thread "main" java.lang.ArithmeticException: / by zero  
at RuntimeErrorTest.main(RuntimeErrorTest.java:4)
```



주석문과 프로그램 오류

● 실행 시간 오류 : 이클립스의 실행 시간 오류





주석문과 프로그램 오류

● 논리 오류

- 프로그램 작성자의 의도와 다른 결과를 나타내는 오류
- 프로그램의 실행이 종료되어도 나타나지 않기 때문에 오류로 취급되지 않는 경우도 있지만, 프로그램을 작성한 사람이 의도한 결과를 출력하지 않았기 때문에 오류로 취급
- 논리 오류는 프로그램 작성자나 수정하는 사람에 의해 발견되어야 한다



주석문과 프로그램 오류

● 논리 오류의 예 : 예제

예제 2.8

LogicError.java

```
01: public class LogicErrorTest {  
02:     public static void main(String args[]) {  
03:         int i = 300 ;  
04:         int j = 500 ;  
05:         j += i + j ;  
06:         System.out.println("합계는 = " + j) ;  
07:     }  
08: }
```

» 설명

05 작성자의 의도는 $j = i + j$ 인데 잘못 기술됨

실행 결과

합계는 = 1300



■ 학습정리

● 자바 프로그램의 형태

- ① 자바 응용 프로그램
- ② 자바 애플릿
- ③ 자바 서블릿(Servlet)
- ④ JSP(Java Server Page)
- ⑤ 자바 빈스(Beans)

● 학습을 위한 준비

① 표준 출력

- `System.out.print()` : 내용을 모니터로 출력하고 줄을 바꾸지 않습니다.
- `System.out.println()` : 내용을 모니터로 출력하고 줄을 바꿉니다.

② 키보드로부터의 입력

- `java.util.Scanner` 클래스를 이용하여 키보드로부터 입력을 받습니다. 한 줄에 여러 개의 데이터를 입력받기 위해서는 공간을 구분자로 사용합니다.



■ 학습정리

- `next()` : 문자열을 입력받는 메소드
- `nextInt()` : 정수를 입력받는 메소드
- `nextDouble()` : 실수를 입력받는 메소드

③ 자바의 주석문 : 자바에서는 3가지 형태의 주석문이 제공되고 있습니다.

- `/* */` : 여러 줄의 문장을 주석으로 처리할 수 있습니다.
- `// :` 한 줄의 문장을 주석으로 처리합니다.
- `/** */` : 여러 줄의 문장을 주석으로 처리하고, javadoc 도구에 의해 사양서를 만들 수 있는 주석입니다.

④ 자바의 오류

- 구문 오류 : 프로그램 구조를 잘못 사용할 때 발생하는 오류로서 쉽게 발견될 수 있습니다.
- 실행 시간 오류 : 실행 시간에 발생하는 오류로서 프로그램이 비정상적으로 종료될 수 있는 오류입니다.
- 논리 오류 : 프로그램의 컴파일과 실행이 오류 없이 수행되지만, 실행 결과가 작성자의 의도와는 다른 결과를 출력하는 오류입니다. 논리 오류는 발견과 수정이 쉽지 않은 오류입니다.



중간 점검 문제

1. 프로그램에서 데이터를 저장하는 공간은 _____ 이다.
2. 변수에 값을 저장하는 연산을 _____ 이라고 한다.
3. 실행 도중에 값이 변하지 않는 수를 _____ 이라고 한다.



오류

- 컴파일 오류

컴파일 오류는 컴파일시에 발견되는 오류

문법 규칙에 따르지 않으면 구문 오류(syntax error)

- 실행 오류

실행 오류는 프로그램이 실행되는 도중에 발생하는 오류

– (예) 0으로 나누려고 시도

- 논리 오류

논리 오류는 컴파일되고 실행도 되지만 의도하지 않는 결과를 발생하는 경우

(예) 값이 예상과는 다르게 잘못 계산되는 경우

- 디버깅(debugging): 논리 오류를 찾고 고치는 과정



중간 점검 문제

1. 컴파일시에 오류가 없는데도 원하는 결과가 나오지 않는다면 _____ 오류가 있는 것이다.
2. 만약 실행 도중에 0으로 나누었다면 _____ 오류가 발생한다.