

컴퓨터 프로그래밍 (Computer Programming)

이 선 순



12. 파일 입출력



목차

01. 예외 처리

02. 표준입출력

03. 파일입출력

02

표준 입출력

02 표준 입출력

■ 표준 입출력

- JAVA에서는 화면엔 출력하고 입력 받는 표준 입출력 클래스를 미리 정의해 둠
- 이 클래스는 프로그램이 시작될 때 생성되므로 따로 만들 필요는 없음
- 표준 입력은 키보드로 입력하는 것이고 표준 출력은 화면에 출력하는 것임

■ System.out

- 시스템의 표준 출력 장치로 출력

■ System.in

- 시스템의 표준 입력 장치에서 읽음

■ System.err

- 빨간색으로 오류 메시지를 출력할 때 사용함
- out, in, err 모두 정적(static) 변수임. 지금까지 System 클래스를 생성하지 않고도 System.out을 사용할 수 있었던 이유는 out변수가 System클래스의 정적 변수이기 때문임.

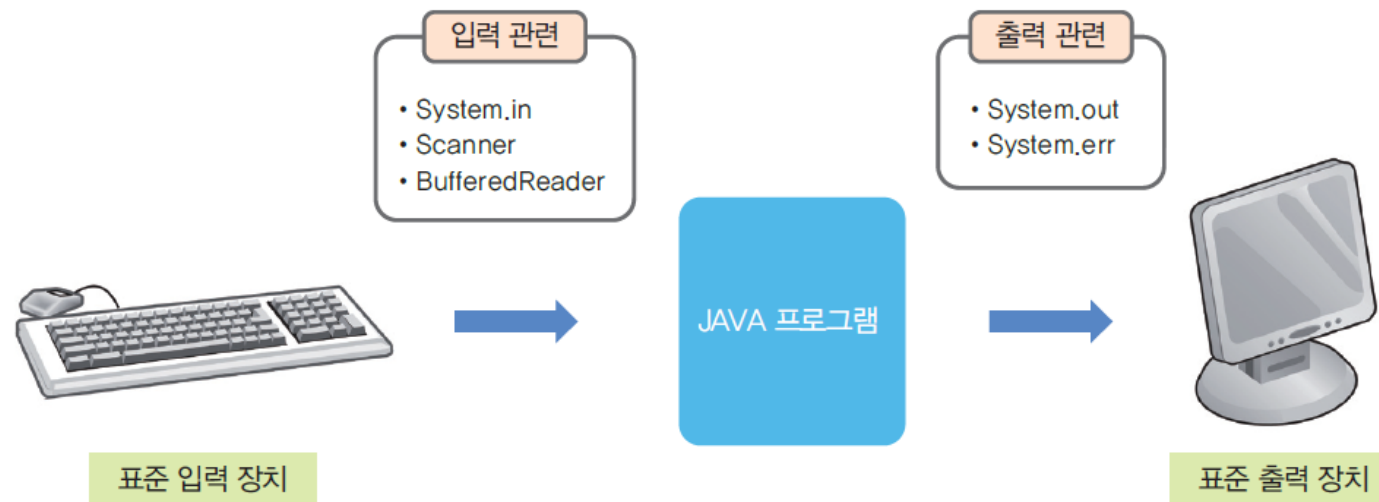


그림 10-7 표준 입출력의 개념

■ 표준 출력 : System.out.printf()

- 서식을 지정해서 출력하는 메소드

구문	설명
System.out.printf("서식", 출력할 매개변수...)	서식에 맞춰 표준 출력 장치(화면)에 출력해준다.

서식	설명
%d	정수형(int)
%c	문자형(char)
%s	문자열(String)
%x	16진수 정수(int)
%o	8진수 정수(int)
%f	실수형(float, double)
%e	공학 계산용 형식

02 표준 입출력

■ 표준 출력 : System.out.printf()

■ [실습10-5] 서식화된 출력 메소드 사용 예

```
1 public class Ex10_05 {  
2     public static void main(String[] args) {  
3         int a = 1234;  
4         float b = 100.12345f;  
5         char ch = 'K';  
6         String s = "IT_Cookbook";  
7  
8         System.out.printf("정수의 10진수 ==> %d\n", a);  
9         System.out.printf("정수의 16진수 ==> %X\n", a); //정수를 10진수, 16진수, 8진수로 출력  
10        System.out.printf("정수의 8진수 ==> %o\n", a);  
11        System.out.printf("실수 ==> %10.3f\n", b); //실수를 일반 방식 및 공학용으로 출력  
12        System.out.printf("실수(공학용) ==> %e\n", b);  
13        System.out.printf("문자 ==> %c\n", ch); //문자와 문자열을 출력  
14        System.out.printf("문자열 ==> %s\n", s);  
15    }  
16 }
```

Problems @ Javadoc Declaration Console

<terminated> Ex10_05 [Java Application] C:\Program Files\Java\jdk-15.0.2\bin

```
정수의 10진수 ==> 1234  
정수의 16진수 ==> 4D2  
정수의 8진수 ==> 2322  
실수 ==> 100.123  
실수(공학용) ==> 1.001235e+02  
문자 ==> K  
문자열 ==> IT_Cookbook
```

02 표준 입출력

- `System.out.println()` : 괄호 안의 내용을 그대로 출력한 후 한 행을 띄움
- `System.out.print()` : 괄호 안의 내용을 출력한 후 한 행을 띄지 않고 유지

■ 표준 입력 : Scanner

- Java.util 패키지에 있는 입력 클래스
- 문자 뿐만 아니라 정수, 실수 등 다른 자료형도 읽을 수 있음
- 콘솔 화면뿐 아니라 파일이나 문자열을 생성자의 매개변수로 받아 자료를 읽어 올 수 있음

```
Scanner s = new Scanner(System.in);  
변수 = s.메소드( )
```

제공 메소드	설명
nextByte()	정수를 입력받는다(범위 : -128~+127).
nextShort()	정수를 입력받는다(범위 : -32768~+32767).
nextInt()	정수를 입력받는다(범위 : 약 -21억~+21억).
nextLong()	정수를 입력받는다(범위 : 약 -900경~+900경).
nextFloat()	실수를 입력받는다(정밀도는 소수점 아래 약 일곱 자리).
nextDouble()	실수를 입력받는다(정밀도는 소수점 아래 약 열다섯 자리).
next()	한 단어를 입력받는다.
nextLine()	한 줄을 입력받는다.

02 표준 입출력

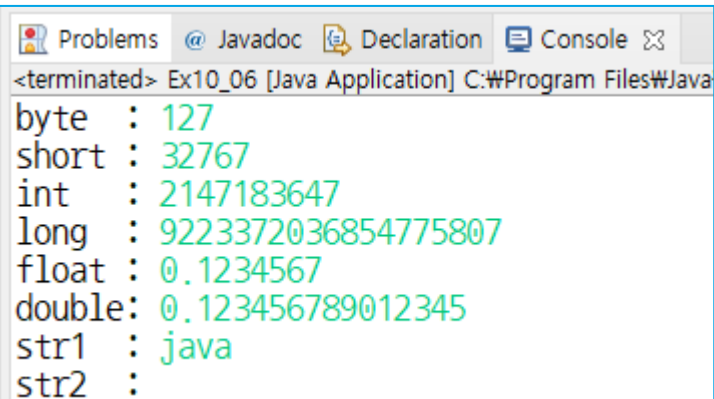
■ 표준 입력 : Scanner

■ [실습10-6]표준 입력 사용 예

```
1 import java.util.Scanner;
2
3 public class Ex10_06 {
4     public static void main(String[] args) {
5         Scanner s = new Scanner(System.in);
6
7         byte a;
8         short b;
9         int c;
10        long d;
11        float e;
12        double f;
13        String str1, str2;
14
15        System.out.print("byte : ");
16        a = s.nextByte();
17        System.out.print("short : ");
18        b = s.nextShort();
19        System.out.print("int : ");
20        c = s.nextInt();
21        System.out.print("long : ");
22        d = s.nextLong();
23        System.out.print("float : ");
24        e = s.nextFloat();
25        System.out.print("double: ");
26        f = s.nextDouble();
27        System.out.print("str1 : ");
28        str1 = s.next();
29        System.out.print("str2 : ");
30        str2 = s.nextLine();
31
32    }
33 }
34
35 }
```

Scanner 형의 변수 s를 선언

다양한 변수 선언



```
Problems @ Javadoc Declaration Console
<terminated> Ex10_06 [Java Application] C:\Program Files\Java
byte : 127
short : 32767
int : 2147183647
long : 9223372036854775807
float : 0.1234567
double: 0.123456789012345
str1 : java
str2 :
```

02 표준 입출력

■ 표준 입력 : Scanner

- [실습10-6]표준 입력 사용 예
 - 20행

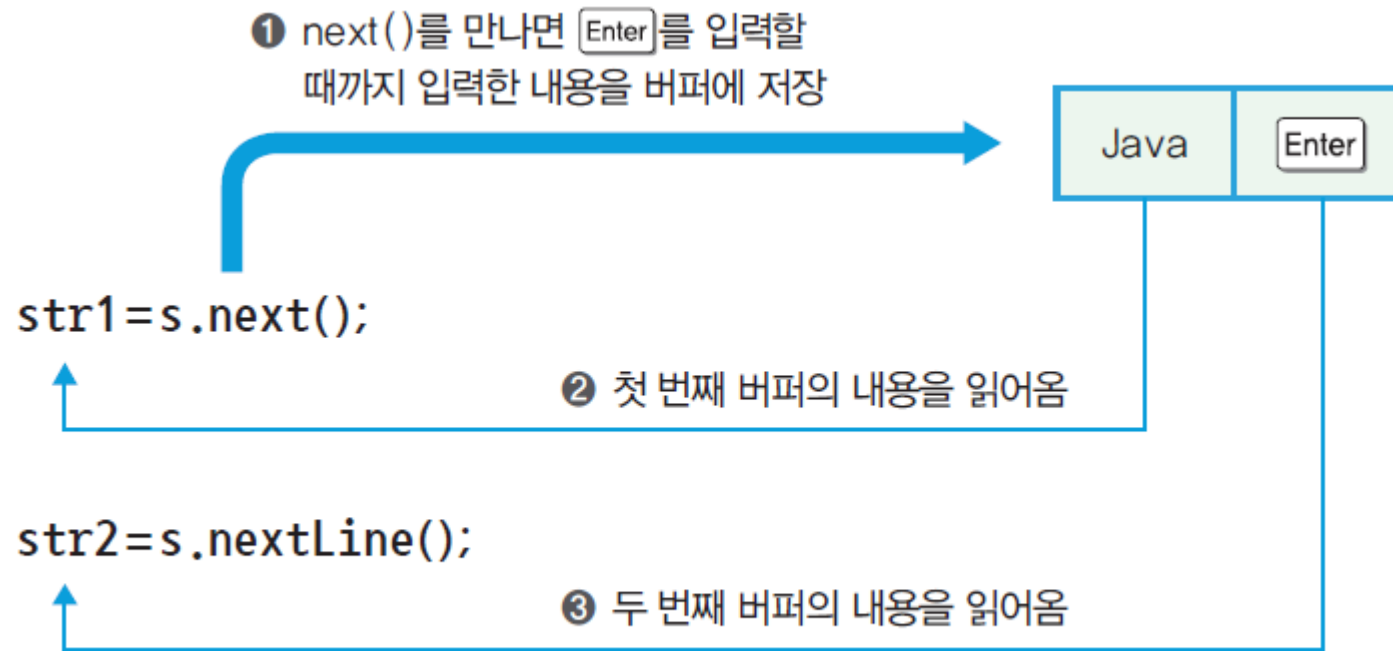


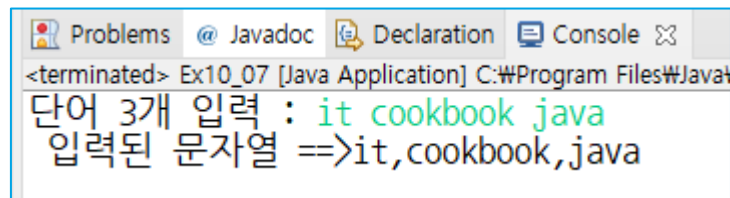
그림 10-10 next()의 작동

02 표준 입출력

■ 표준 입력 : Scanner

■ [실습10-7]next()의 작동 예

```
1 import java.util.Scanner;
2
3 public class Ex10_07 {
4     public static void main(String[] args) {
5         Scanner ss = new Scanner(System.in);
6         String str1, str2, str3;
7
8         System.out.print("단어 3개 입력 : ");
9         str1 = ss.next();
10        str2 = ss.next(); //버퍼에서 3개의 단어를 읽어들인다.
11        str3 = ss.next();
12
13        System.out.print(" 입력된 문자열 ==>");
14        System.out.print(str1 + "," + str2 + "," + str3);
15
16        ss.close();
17    }
18 }
```



Problems @ Javadoc Declaration Console

<terminated> Ex10_07 [Java Application] C:\Program Files\Java\jre6\bin\java.exe

단어 3개 입력 : it cookbook java

입력된 문자열 ==>it,cookbook,java

- 8행에서 3개의 단어를 공백으로 분리해서 입력한 후 [Enter]를 누르면 버퍼의 단어가 차례대로 str1, str2, str3 에 공백으로 분리되어 들어간다.

■ 하나의 문자 입력 : System.in.read()

- [Enter]를 누를 때까지 키를 입력받고 입력한 결과가 "hanbit"이면 통과하는 프로그램
- [실습10-8] System.in.read() 사용 예

```
1 import java.io.IOException;
2
3 public class Ex10_08 {
4     public static void main(String[] args) {
5
6         String userName = "hanbit";
7         String input = "";
8
9         int key;
10
11         try {
12             System.out.print("이름 ==> ");
13
14             while ((key = System.in.read()) != 13) {
15                 input += Character.toString((char) key);
16             }
17
18             if (userName.equals(input))
19                 System.out.println(input + "님 어서오세요 ~");
20             else
21                 System.out.println(input + "님 등록이 안되었습니다~");
22
23         } catch (IOException e) {
24             e.printStackTrace();
25         }
26     }
27 }
28 }
```

사용자 이름을 hanbit으로 고정
입력받을 input 변수를 준비

System.in.read()메소드는 try~catch문으로 묶어야 한다

입력한 키가[Enter](아스키코드값 :13)가 아니면 무한 반복.
그리고 입력한 키는 key 변수에 저장

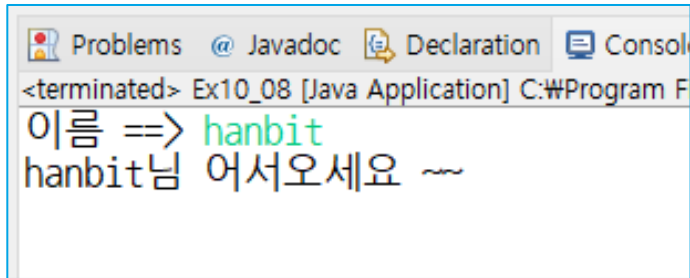
입력한 정수형 키를 문자열로 변경해서 input에 덧붙인다.

사용자 이름인 hanbit과 입력한 값이
동일한지 확인하여 메시지를 출력

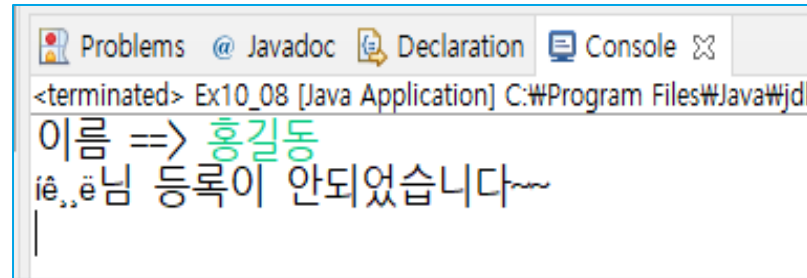
02 표준 입출력

■ 하나의 문자 입력 : System.in.read()

- [Enter]를 누를 때까지 키를 입력받고 입력한 결과가 "hanbit"이면 통과하는 프로그램
- [실습10-8] System.in.read() 사용 예



```
<terminated> Ex10_08 [Java Application] C:\Program F  
이름 ==> hanbit  
hanbit님 어서오세요 ~
```

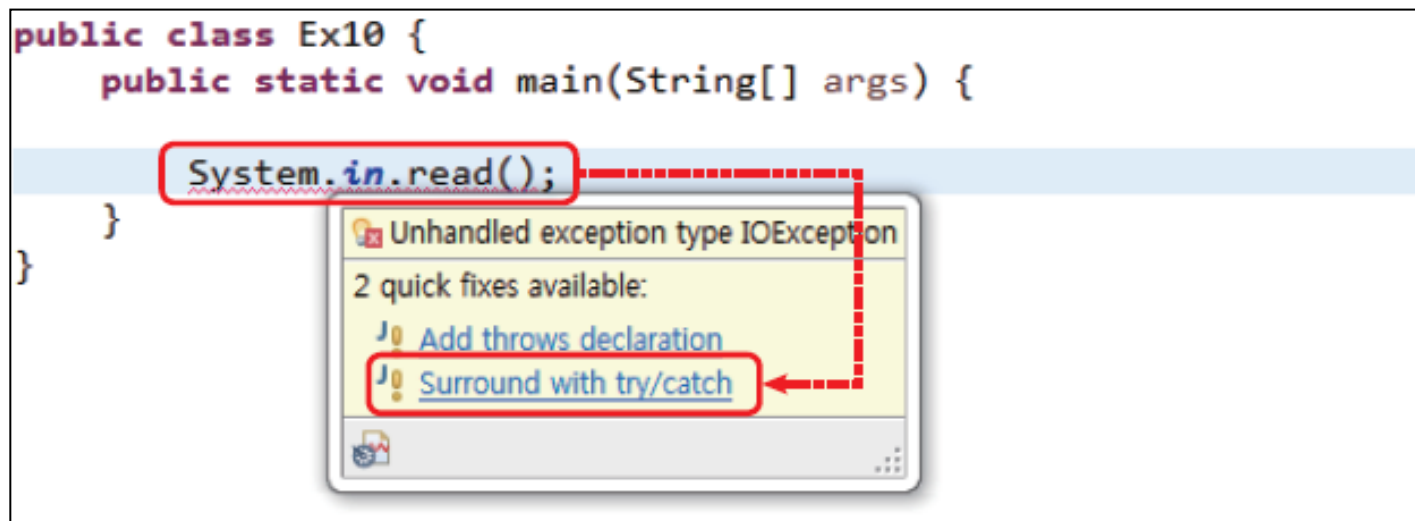


```
<terminated> Ex10_08 [Java Application] C:\Program Files\Java\jdk  
이름 ==> 홍길동  
이름 등록이 안되었습니다~  
|
```

- ❖ 코드가 실행되면 'hanbit' 이라고 입력
- ❖ 다른 이름 입력하면 결과가 안나옴

■ 이클립스의 자동완성 기능

- try~catch 문을 사용할 때 직접 코딩하여 오류를 처리할 부분을 묶어도 되지만, 이클립스에서는 자동으로 완성해주는 기능을 사용하면 더 편리함
- System.in.read();를 입력하면 빨간 줄이 생기고 마우스를 가져가면 풍선 도움말이 나옴. 그 중에서 [Surround with try/catch]를 선택하면 자동으로 try~catch 문으로 묶어준다. 상당히 편리한 기능이니 적극 활용하자.



질문은 이메일을 이용해주세요.
ds.june2@gmail.com

감사합니다