
work.com. workshop01 공통 패키지

[문제 1] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오.

- int, double, char, boolean 자료형을 각각 변수로 선언하고, 해당 변수의 값과 자료형을 출력하시오.
- 출력 시 변수의 자료형과 변수에 저장된 값을 함께 서술하는 형태로 출력하시오.

1. 구현 클래스

클래스명	메소드	설명
Test01	<u>+main(String args[]): void</u>	main 함수 안에서 모든 코드 작업 진행

2. 실행 결과

출력은 아래와 같다 .

정수형 변수의 값은 100이며, 자료형은 int 입니다.
실수형 변수의 값은 3.14이며, 자료형은 double 입니다.
문자형 변수의 값은 A이며, 자료형은 char 입니다.
논리형 변수의 값은 true이며, 자료형은 boolean 입니다.

[문제 2] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오.

변수 선언이 아래와 같을 때 실행 결과와 같이 출력 될 수 있도록 프로그램 완성 하시오.

```
ascii 값 'A' -> 65
ascii 값 'B' -> 66
ascii 값 '1' -> 49
ascii 값 '2' -> 50
```

```
class Test02 {
    public static void main(String[] args)
    { String s1 = "1";
      String s2 = "2";
      boolean bnx = false;
      char c1 = 'A';
      char c2 = 'B';
      char c3 = '1';
      char c4 = '2';
      int inx = 2;

      System.out.println(_____);
      System.out.println(_____);
      System.out.println(_____);
      System.out.println(_____);
      System.out.println(_____);
    }
}
```

1 구현 클래스

클래스명	메소드	설명
Test02	<u>+main(String args[]): void</u>	main 함수 안에서 모든 코드 작업 진행

2. 실행 결과

출력은 아래와 같다.

```
12
true
131
51
99
```

[문제 3] 비트 시프트 연산을 이용한 특정 자릿수 변환.

- 5 자리 숫자를 정수형 변수 num 에 저장한 후, 비트 시프트 연산(>>)을 이용하여 천의 자리 이하를 0 으로 변환하시오.
- 예를 들어 num = 45678 일 경우, 출력 값은 45000 이 되어야 한다.
- 비트 연산을 활용하여 천 단위로 반올림하지 않고 단순 변환한다.

```
class Test03{
    public static void main (String[] args){

        int num = 45678;

        int result = (num / 1000) << _____ ;

        System.out.println("기존 숫자: " + num);
        System.out.println("비트 연산 후 변환된 숫자: " + result);
    }
}
```

1. 구현 클래스

클래스명	메소드	설명
Test03	<u>+main(String args[]): void</u>	main 함수 안에서 모든 코드 작업 진행

2. 실행 결과

출력은 아래와 같다.

기존 숫자: 45678

비트 연산 후 변환된 숫자: 45000

[문제 4] 삼항 연산자를 이용한 영문자 판별 프로그램을 작성하자.

- 문자형 변수(char ch)를 선언하여 값을 할당한 후, 해당 문자가 대문자, 소문자 판별하는 결과를 출력하시오.
- 삼항 연산자(? :)를 사용하여 조건을 처리하시오.
- 출력 형식은 "입력한 문자 'x'는 영문자입니다." 또는 "입력한 문자 'x'는 영문자가 아닙니다."로 출력하시오.

```
class Test04 {
    public static void main(String[] args) {

        char ch = 'G'; // 다른 문자로 변경하여 테스트 가능

        // 삼항 연산자를 사용하여 영문자인지 판별
        String result = (ch >= 'A' && ch <= 'Z') || _____
                        ? "입력한 문자 " + ch + "는 영문자입니다."
                        : _____;

        // 결과 출력
        System.out.println(result);
    }
}
```

1. 구현 클래스

클래스명	메소드	설명
Test04	<u>+main(String args[]): void</u>	main 함수 안에서 모든 코드 작업 진행

2. 실행 결과

출력은 아래와 같다.

입력한 문자 'G'는 영문자입니다.

[문제 5] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오.

화씨를 섭씨로 변경하는 프로그램을 작성 하시오.

C(celcius): 섭씨, F(fahrenheit): 화씨

공식: $C = 5/9 * (F-32)$

```
class Test05 {
    public static void main(String[] args)
    { int fahrenheit = 100;

      float celcius = _____ 공식을 이용하여 프로그램을 완성 하시오 _____;
      System.out.println("Fahrenheit:"+fahrenheit);
      System.out.println("Celcius:"+celcius);
    }
}
```

1. 구현 클래스

클래스명	메소드	설명
Test05	<u>+main(String args[]): void</u>	main 함수 안에서 모든 코드 작업 진행

2. 실행 결과

출력은 아래와 같이 이루어진 다 .

```
Fahrenheit:100
Celcius:37.77778
```

[문제 6] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오.

변수 선언이 아래와 같이 되어 있을 때 실행 결과와 같이 출력 될 수 있도록 프로그램 하시오.

```
class Test06{
    public static void main(String[] args)
    { byte a = 10;
      byte b = 20;
      byte c = _____;
      char ch = 'A';
      ch = _____;
      float f = _____;
      long l = _____;
      float f2 = 0.1f;
      double d = 0.1;
      boolean result = (float)d _____;

      System.out.println("c="+c);
      System.out.println("ch="+ch);
      System.out.println("f="+f);
      System.out.println("l="+l);
      System.out.println("result="+result);
    }
}
```

1. 구현 클래스

클래스명	메소드	설명
Test06	<u>+main(String args[]): void</u>	main 함수 안에서 모든 코드 작업 진행

2. 실행 결과

출력은 아래와 같이 이루어진 다 .

```
c=30
ch=C
f=1.5
l = 27000000000
result=true
```

[문제 7] 삼항 연산자를 이용한 영문자 판별 프로그램 (Static 메소드 활용)

- 문자형 변수(char ch)를 선언하여 값을 할당한 후,
해당 문자가 대문자인지 소문자인지 판별하여 결과를 출력하시오.
- 삼항 연산자(? :)를 사용하여 조건을 처리하되, static 메소드를 사용하여 코드를
구조화하시오.
- isAlphabet(char ch), printResult(char ch) 두 개의 메소드를 만들어 호출하는
방식으로 구현하시오.

```

class Test07 {
    public static void main(String[] args) {
        // 테스트할 문자 변수 선언
        char ch = 'G'; // 다른 문자로 변경하여 테스트 가능

        // 결과 출력
        printResult(ch);
    }

    // 문자가 영문자인지 판별하는 static 메소드
    public static boolean isAlphabet(char ch) {
        return (ch >= 'A' && ch <= 'Z') || _____);
    }

    // 결과를 출력하는 static 메소드
    public static void printResult(char ch) {
        String result = isAlphabet(ch)
            ? "입력한 문자 '" + ch + "'는 영문자입니다."
            : _____;

        System.out.println(result);
    }
}

```

1. 구현 클래스

클래스명	메소드	설명
Test07	<code>+main(String args[]): void</code>	main 함수 안에서 모든 코드 작업 진행
	<code>isAlphabet(char ch): boolean</code>	문자가 영문자인지 판별하여 true/false 반환
	<code>+printResult(char ch): void</code>	영문자인지 판별하고 결과를 출력

2. 실행 결과

출력은 아래와 같이 이루어진 다 .

입력한 문자 'G'는 영문자입니다.