



실 습 문 제

제3장

1. [변수 선언 및 대입 연산] 변수를 선언하고 값을 대입하는 과정을 살펴보자.

```
#include <stdio.h>

int main(void)
{
    int x;
    int y;
    int result;           // ①

    x = 200;              // ②
    y = 100;              // ③
    result = x + y;
    printf("%d\n", result);

    return 0;
}
```

- (1) 위의 프로그램에 주석을 붙이도록 하라. 첫 부분에 전체 프로그램을 설명하는 주석을 둔다. 각 줄 끝에는 그 줄에 실린 코드를 설명하는 주석을 붙인다. 수정한 프로그램을 다시 컴파일하고 실행하여 보자.
 - (2) 각 변수들을 본문에서처럼 상자로 그리고, 각 줄이 실행된 다음에 상자의 내용이 어떻게 변화되는지를 그려보라.
 - (3) 덧셈 연산을 뺄셈, 곱셈, 나눗셈 연산으로 변경하여 보고 각각의 결과를 출력하여 보라.
 - (4) 문장 ①과 문장 ②를 순서를 바꾸면 어떻게 되는가? 그 이유는 무엇인가?
 - (5) 문장 ②와 ③을 삭제한 다음에, 변수 x를 200으로, 변수 y를 100으로, 변수 result를 0으로 초기화하도록 변수 선언 문장을 변경하여 보라.
 - (6) 변수 x, y, result는 모두 그 자료형 같다. 한 줄에서 모두 선언하도록 변경하여 보라.
 - (7) 변수 x, y, result를 모두 실수형인 float형으로 변경한 뒤에, 덧셈, 뺄셈, 곱셈, 나눗셈 연산을 하여 출력하여 보라.
2. [printf()] printf()를 이용하여 변수의 값을 화면에 출력하여 본다.



```
#include <stdio.h>

int main(void)
{
    int x = 200;
    int y = 100;

    printf("Hello C Programmers!\n");    // ①

    printf("%d\n", x);                  // ②
    printf("%d %d\n", x, y );          // ③
    printf("%d\n", x + y );            // ④

    return 0;
}
```

실행결과(기록하기)

- (1) 위의 프로그램을 오류없이 입력하고, 컴파일하여 실행하자. 어떤 결과가 출력되는가?
- (2) 출력되는 문자열을 "Hello Everyone!"으로 변경하고, 만약 여기서 "Hello Everyone!과 같이 끝에 따옴표가 생략되면 어떻게 되는지를 확인하라.
- (3) printf()의 %d 형식 지정자를 이용하여 x, y, x+y를 동시에 화면에 출력하여 보라.
- (4) 다음과 같은 출력 화면이 생성되도록 프로그램을 수정하라.

```
Hello C Programmers!
x = 10
x = 10    y = 20
x + y = 30
```

- (5) 형식 지정자와 변수의 자료형을 맞추지 않았을 경우, 어떤 일이 발생하는지를 알아보자. 문장 ②를 다음과 같이 변경하여 실행 결과를 살펴보자.

```
printf("%f\n", x);    // ②
```

- (6) 형식 지정자의 개수와 변수의 개수를 다르게 하면 역시 오류가 발생한다. 다음 문장에서 형식 지정자는 2개이지만 변수는 1개이다. 어떤 식으로 출력되는지 관찰하라.

```
printf("%d %d\n", x);    // ②
```

- (7) 변수 x와 y를 모두 실수형으로 변경하여서 실행 결과를 출력하여 보자.

3. [scanf()] scanf()를 이용하여 사용자로부터 값을 입력받아서 변수에 저장해본다.

```
#include <stdio.h>

int main(void)
{
    int x;

    printf("정수를 입력하십시오: ");
    scanf("%d", &x); // ①
    printf("입력된 정수는 %d입니다.\n", x);

    return 0;
}
```

실행결과(기록하기)

(1) 문장 ①에서 변수 앞에 있는 &기호를 제거하여서 실행 결과를 관찰하라. 다시 원상태로 회복시켜라.

```
scanf("%d", x); // ①
```

(2) 문장 ①에서 %d를 실수형을 받아들이는 형식 지정자 %f로 변경하여서 실행 결과를 관찰하라. 다시 원상태로 회복시켜라.

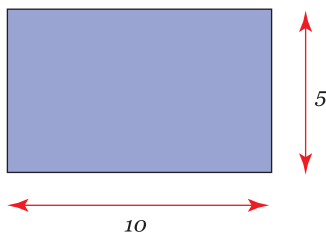
```
scanf("%f", &x); // ①
```

(3) scanf()가 한번에 2개의 입력을 받도록 다음과 같이 변경하여 보라. 변수 y를 정수형으로 선언하자. scanf()가 완료된 다음에 변수 y의 값도 출력하여보자.

```
scanf("%d %d", &x, &y);
```

(4) 변수 x의 자료형을 실수로 바꾸고 scanf()와 printf()의 형식 지정자를 모두 %f를 바꾸어서 실수가 입력되게끔 변경하여 보자.

4. [응용] 가로 길이가 10이고 세로 길이가 5인 직사각형의 둘레와 면적을 구하는 프로그램을 작성한다.





```

/* 직사각형의 둘레와 면적 */
#include <stdio.h>

int main(void)
{
    int w;                // 정수형 변수 w 선언
    _____          // 정수형 변수 h 선언
    _____          // 정수형 변수 peri 선언
    _____          // 정수형 변수 area 선언

    w = 10;              // 변수 w에 10을 대입
    _____          // 변수 h에 5를 대입
    _____          // w와 h를 이용하여 둘레를 구하여 peri에 대입
    _____          // w와 h를 이용하여 면적을 구하여 area에 대입

    printf("%d\n", peri); // 변수 peri의 값을 출력한다.
    _____          // 변수 area의 값을 출력한다.

    return 0;
}

```

실행결과

```

직사각형의 둘레 = 30
직사각형의 면적 = 50

```

- (1) 위의 빈 칸을 채운 다음에, 비주얼 C++를 이용하여 프로젝트를 만들고, 프로젝트 안에 소스 파일을 만들어서 위의 프로그램을 입력한다. 컴파일하여 실행하여 위와 같은 결과가 나오는지 먼저 확인하라.
 - (2) 위의 프로그램에서 모든 변수는 따로 따로 선언되어 있다. 이들 변수를 한 줄에 모아서 선언해보라.
 - (3) 한번의 `printf()` 호출로 변수 `peri`와 `area`의 값이 동시에 출력되도록 변경하라.
 - (4) 모든 변수를 실수형으로 바꾸고 1.001과 6.123처럼 실수를 입력하여서, 면적과 둘레가 정확하게 계산되는지를 살펴보자. 출력할 때도 실수형을 지정하는 형식 지정자를 사용하여야 한다.
 - (5) 변수들에 값을 대입하는 문장 대신에, 사용자로부터 값을 받아서 사각형의 면적과 둘레를 계산할 수 있도록 프로그램을 변경하라.
- 5. [응용]** 이번에는 실용적인 프로그램을 만들어보자. 미국에서는 거리를 표시하는데 마일을 사용한다. 마일을 미터로 환산하는 프로그램을 만들어보자. 사용자로부터 마일단위로 거리를 입력 받아서 변수에 저장한다. 이 변수에 1609를 곱하여 미터로 변환한다. 미터로 변환된 값을 화면에 출력한다.

```
마일을 입력하시오: 2 Enter
2마일은 3218미터입니다.
```

이번에는 주석부터 시작하여 거의 모든 것을 스스로 작성하여 보자.

```
_____ // 여기에 프로그램을 설명하는 주석을 넣는다.
_____ // stdio.h 헤더 파일을 삽입한다.

int main(void)
{
    _____ // 사용자에게 받은 마일 값을 저장하는 변수 mile 선언
    _____ // 환산된 미터값을 저장하는 변수 meter 선언

    _____ // "마일을 입력하시오:"라는 문자열을 출력
    _____ // 사용자로부터 마일값을 입력받아 mile에 저장

    _____ // mile에 1609를 곱하여 meter에 대입
    _____ // 변수 meter의 값을 mile과 함께 출력

    _____ // main 함수를 종료하고 0값을 반환한다.
}
```