Chapter 03

데이터와 C

Section 1 기본 데이터형
Section 2 정수형과 부동소수점형의 차이
Section 3 printf( ), scanf( )

VS 2013 사용시 주의사항

- Visual Studio 2013
- C++11 표준 지원, 보안성이 함
  (scanf() 함수 , fopen() 함수 사
  소스 코드 맨 앞에 다음 문장을
  컴파일 오류 없앨 수 있음)
- #define _CRT_SECURE_NO_WARINGS
- #include <stdio.h>

```
1  #define _CRT_SECURE_NO_WARNINGS
2  #include <stdio.h>
3  // ...
4  int main(void)
5  {
6      // 몸체(Body);
7      // scanf() 사용
8      // fopen() 사용
9      return 0;
10 }
```

무엇을 학습할까요?

- 기본 데이터형
- 정수형과 부동소수점형
- printf( ) 함수
- scanf( ) 함수
- 실전 프로젝트 (연습문제 풀이, 총 8 문제)

기초 다지기

세미콜론 Semicolons **(;)** – 문장의 끝 표시

블록 ({ , }) – 블록 시작과 끝

scanf("%lf", &var); printf("%lf", var);

표준 입출력 함수

printf( ) / scanf( )

return 문장;

대입문 =

변환지정자	데이터형	함수
%c	char	printf/scanf
%d	int	printf/scanf
%f	double	printf
%lf	double	scanf

프로그래밍 연습 문제 p117

01

정수 오버플로, 부동소수점 수 오버플로와 언더플로가 발생했을 때 사용자의 시스템이 무엇을 하는지 알아보라.
즉, 이 문제를 확인할 수 있는 프로그램을 작성하라.
(가장 큰 값과 가장 작은 값에 대해 알아보기 위해 limits.h 와 float.h에 대한 4장의 논의를 살펴볼 수 있다.)

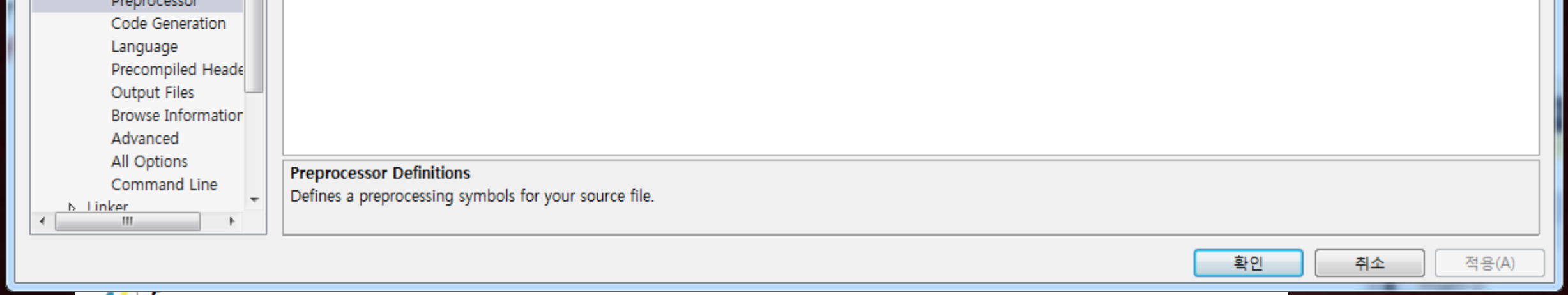
프로그래밍 연습 소스 01

```c
/* Programming Exercise 3-1 */
#include <stdio.h>
int main(void)
{
    int ibig = 2147483647; // big + 1: -2147483648
    float big = 1.0e37; // too big: big * big: 1.000000e+074
    float small = 1.0e-37; // too small:small / big: 1.000000e-074

    printf("ibig: %d, big + 1: %d\n", ibig, ibig + 1);
    printf("big: %e, too big: %e\n", big, big * big);
    printf("small: %e, too small: %e\n", small, small / big);

    return 0;
}
```

프로그래밍 연습 문제 p117

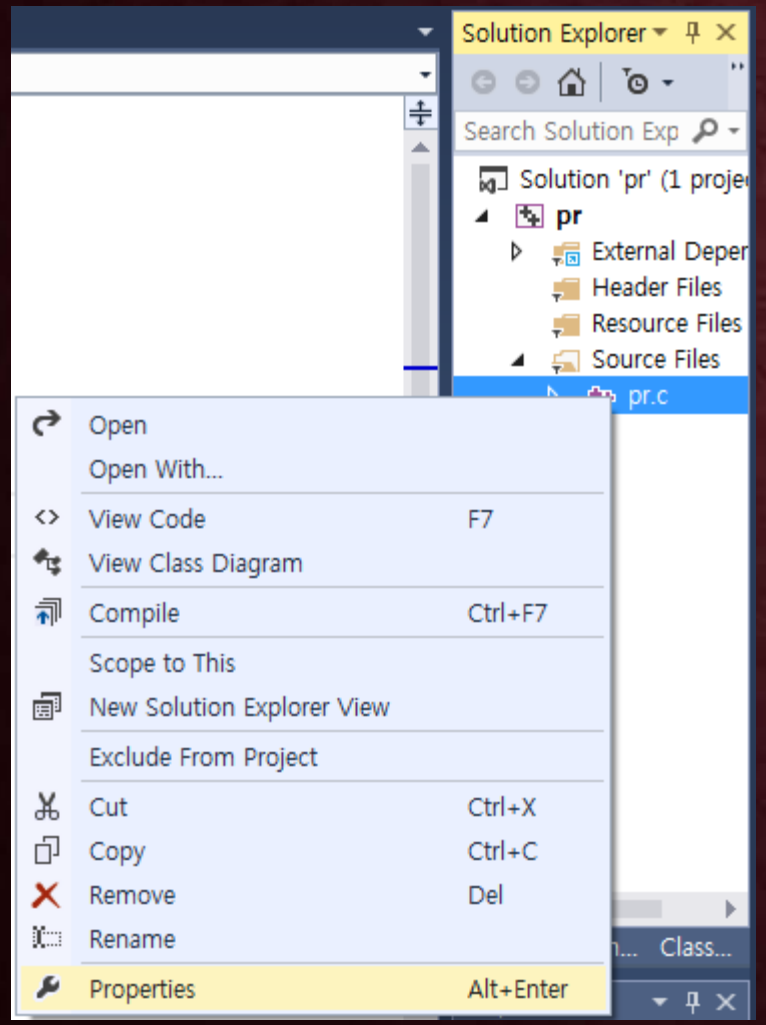
02 66과 같은 ASCII 코드값을 사용자에게 입력하도록 요청하고,
그 ASCII 코드값에 해당하는 문자를 출력하는 프로그램을 작성하라.

```c
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
// ...
int main(void)
{
    // 몸체(Body);
    // scanf() 사용
    // fopen() 사용
    return 0;
}
```



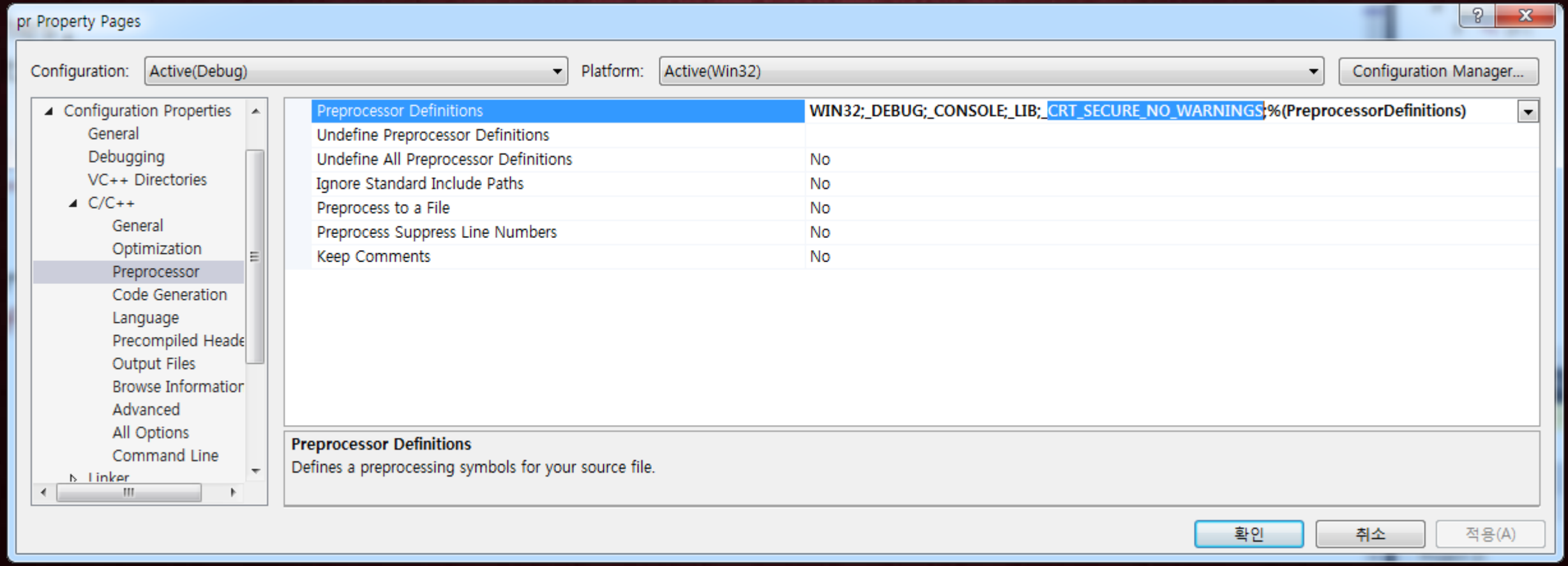

```c
{
    int ascii;

    printf("Enter an ASCII code: ");
    scanf("%d", &ascii);

    printf("%d is the ASCII code for %c.\n", ascii, ascii);

    return 0;
}
```

프로그래밍 연습 소스 02




```
#define _CRT_SECURE_NO_WARINGS
#include <stdio.h>
```

```
#define _CRT_SECURE_NO_WARINGS
#include <stdio.h>
```



프로그래밍 연습 문제 p117

03 경보음을 한 번 울리고, 다음과 같은 텍스트를 화면에 출력하는 프로그램을 작성하라.

갑작스런 소리에 깜짝 놀라 샐리는 외쳤다,

"호박대왕이다!"

프로그래밍 연습 소스 03

```c
/* Programming Exercise 3-3 */
#include <stdio.h>
int main(void)
{
    printf("\a갑작스런 소리에 깜짝 놀라 샐리는 외쳤다,\n");
    printf("\"호박대왕이다!\"\n");

    return 0;
}
```

프로그래밍 연습 문제 p117

04

부동소수점 수를 하나 읽고, 처음에는 소수점 표기로,
다음에는 지수 표기로, 그 다음에는 시스템이 지원한다면 p 표기로
출력하는 프로그램을 작성하라.

출력은 다음과 같이 되어야 한다. (실제로 화면에 표시되는 지수부의
자릿수는 시스템마다 다를 수 있다.)

```
부동소수점 수를 입력하시오: 64.25
고정소수점 표기: 64.250000
지수 표기: 6.425000e+01
p 표기: 0x1.01p+6
```

프로그래밍 연습 소스 04

(주의) %a: C99 지원, 16진수 소문자에서 사용

```c
/* Programming Exercise 3-4 */
#include <stdio.h>
int main(void)
{
    float num;

    printf("부동소수점 수를 입력하시오: ");
    scanf("%f", &num);

    printf("고정소수점 표기: %f\n", num);
    printf("지수 표기: %e\n", num);
    printf("p 표기: %a\n", num);

    return 0;
}
```

```c
#define _CRT_SECURE_NO_WARINGS
#include <stdio.h>
```

프로그래밍 연습 문제 p117

05

1년은 약 3.156×107초에 해당한다. 나이를 햇수로 입력 받아,
초 단위로 출력하는 프로그램을 작성하라.

프로그래밍 연습 소스 05

```c
/* Programming Exercise 3-5 */
#include <stdio.h>
int main(void)
{
    float sec_per_year = 3.156e7; /* e7:10000000 */
    float age;
    float age_in_sec;

    printf("나이를 햇수(몇년)로 입력: ");
    scanf("%f", &age);

    age_in_sec = age * sec_per_year;
    printf("당신의 나이는 %e 초이다.\n", age_in_sec);

    return 0;
}
```

```c
#define _CRT_SECURE_NO_WARINGS
#include <stdio.h>
```

프로그래밍 연습 문제 p118

06

물 분자의 질량은 약 3.0×10^{-23}그램이다.
물 1쿼트(quart)는 약 950그램이다.

쿼트 단위로 물의 양을 입력받아,
그 안에 들어 있는 물 분자의 개수를 출력하는 프로그램을 작성하라.

10^{-23}

프로그래밍 연습 소스 06

```c
/* Programming Exercise 3-6 */
#include <stdio.h>
int main(void)
{
    float mass_mol = 3.0e-23; /* 물분자 질량 */
    float mass_qt = 950; /* 1 쿼트(quart) = 950 grams */
    float quarts;
    float molecules;

    printf("쿼트 단위로 물의 양을 입력: ");
    scanf("%f", &quarts);

    molecules = quarts * mass_qt / mass_mol;
    printf("%f 쿼트의  물 분자의 개수 = %e \n", quarts, molecules);
    return 0;
}
```

```c
#define _CRT_SECURE_NO_WARINGS
#include <stdio.h>
```

프로그래밍 연습 문제 p118

07

1인치는 2.54센티미터이다.

키를 센티미터 단위로 입력받아 인치 단위로 출력하는 프로그램을 작성하라.

원한다면 인치 단위로 입력받아 센티미터 단위로 출력하는 프로그램을 작성해도 상관없다. (소스 코드는 인치 단위로 입력한 경우)

프로그래밍 연습 소스 07

```c
/* Programming Exercise 3-7 */
#include <stdio.h>
int main(void)
{
    float inches;
    float centimeters;

    printf("키를 센티미터 단위로 입력: ");
    scanf("%f", &inches);

    centimeters = 2.54 * inches;
    printf("%f 인치는 %f 센티미터이다.\n", inches, centimeters);

    return 0;
}
```

1 2 3

```c
#define _CRT_SECURE_NO_WARINGS
#include <stdio.h>
```

프로그래밍 연습 문제 p118

08

미국에서는 부피 단위로 pint를 사용한다. 1 pint는 2 cups이고,
1 cup은 8 ounces, 1 ounce는 2 tablespoons,
1 tablespoon은 3 teaspoons이다.

cup 단위로 부피가 얼마인지 묻고 그것을 pints, ounces, tablespoons, teaspoons으로 환산하여 표시하는 프로그램을 작성하라.
왜 이 경우에 부동소수점형이 정수형보다 더 적합한지 설명하라.

(1 Oz(Ounce) = 29.5ml,

영국에서는 $\frac{1}{20}$, 미국에서는 $\frac{1}{16}$ 파인트(pint)에 해당하는 액체의 양

식탁 숟가락의 약 1/3; 略 tsp

L = liter : 리터의 약자.(liter는 litre로 쓰기도 합니다.)

1cc = 1ml, c = cup : 컵의 약자, 1c = 200cc = 200ml

TS = talbe spoon : 테이블 스푼(식탁용 숟가락)의 약자,

약 1/2 온스 (약 3ts), 1TS = 15cc = 15ml

ts = tea spoon : 티스푼(찻스푼)의 약자, 1ts = 5cc = 5ml,

1/8 온스, 1TS= 15ml = 3ts = 3 * 5ml

1 Oz(Ounce) = 29.5ml
1 Dash(대시) = 5~6drop
1 ts(티스푼) = 1/8 온스
1 Ts(테이블 스푼) = 약 1/2 온스 (약 3ts)

1 Cup(컵) = 8온스(240ml)
1 Pint(핀트) = 16온스(2cup = 480ml)
1 Quart(쿼트) = 32온스(4cup = 960ml)
1 Gallon(갈론) = 128온스(4쿼터)

1 Pony(포니) = 1온스(30ml)
1 Finger(핑거) = 1온스(30ml),
손가락 대중(검지)

1 Jigger(지거) - 1 1/2온스(45ml)
1 Gill(길) = 4온스(120ml)
1 Split(스플리트) = 6온스(180ml)

1 Magnum(마그넘) = 1.5 (liter)
1 Jeroboam(제로보움) = 3 (liter)

1 합 = 180ml
1 승 = 1.8
1cc = 1ml
lump = 각설탕 단위

프로그래밍 연습 소스 08

```
#include <stdio.h>
int main(void)
{
    float pints, cups, ounces, tablespoons, teaspoons;
    printf("PLease enter the volume in cups: ");
    scanf("%f", &cups);

    pints = cups/2.0;
    ounces = 8.0 * cups;
    tablespoons = 2.0 * ounces;
    teaspoons= 3.0 * tablespoons;
    printf("%f cups are equivalent to each of the following:\n", cups);
    printf("%f pints\n", pints);
    printf("%f tablespoons\n", tablespoons);
    printf("%f teaspoons\n", teaspoons);
    return 0;
}
```

```
#define _CRT_SECURE_NO_WARINGS
#include <stdio.h>
```

KeyPoint 핵심 개념

- 기본 데이터형
- 정수형: int, short, long , unsigned
- 문자형: char
- 부동수소점형: float, double

- printf( ) : int x, float y, double z, char k, char str[256];

```
printf("%d %f %lf %c %s", x, y, z, k, str);
```

- scanf( ) : scanf("%d %f %lf %c %s", &x, &y, &z, &k, str);

이번 학습에 이어!

- 다음 강의 내용
  - 제 4장 문자열과 입출력 포맷, 연습문제

Thank you !