

-Part1-

제3장 변수란 무엇인가?

(교재 58페이지 ~ 74페이지)

학습목차

3.1 변수 선언하기

-교재 59페이지 -

3.2 변수 선언 시 주의할 점

-교재 66페이지 -

3.1 변수 선언하기

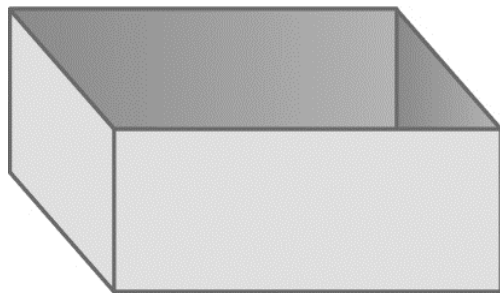
-교재 59페이지 -

3.1 변수 선언하기 (1/7)

▶ 변수란?

- ✓ 데이터를 저장하는 임시 저장 공간
- ✓ 메모리 공간

임시 저장 공간



변수



카트

3.1 변수 선언하기 (2/7)

▶ 변수 a, b 만들어 보기

```
/* 3-1.c */
#include<stdio.h>
int main(void)
{
    int a;    // 변수 a (메모리공간 a)
    int b;    // 변수 b (메모리공간 b)

    return 0;
}
```

3.1 변수 선언하기 (3/7)

▶ 변수 선언 방법 - 변수의 종류

- ✓ 정수형 변수: char형, short형, int형, long형
- ✓ 실수형 변수: float형, double형, long double형

```
/* 3-2.c */
#include<stdio.h>
int main(void)
{
    int a;    // 정수형 변수 선언
    float b;  // 실수형 변수 선언

    return 0;
}
```

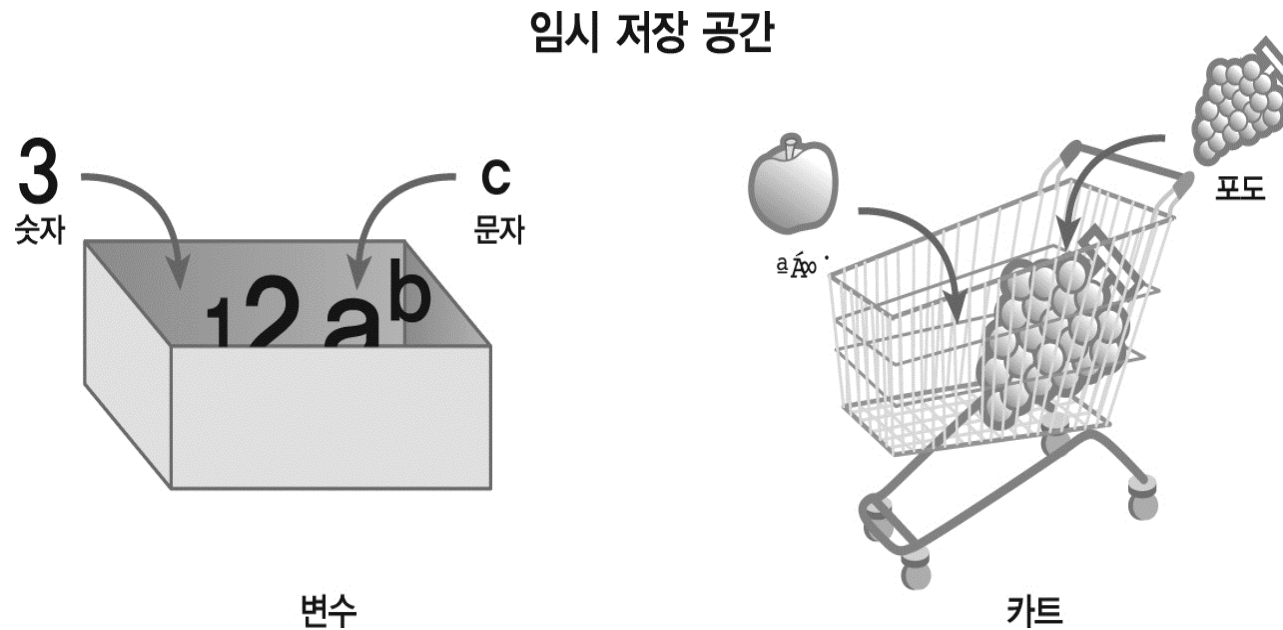
```
/* 3-3.c */
#include<stdio.h>
int main(void)
{
    int a, b;    // 정수형 변수 선언
    float c, d;  // 실수형 변수 선언

    return 0;
}
```

3.1 변수 선언하기 (4/7)

▶ 데이터란?

- ✓ 변수에 저장되는 값



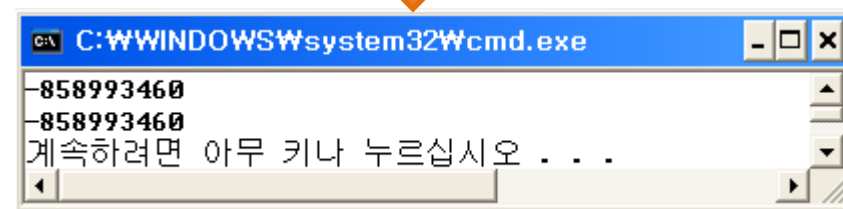
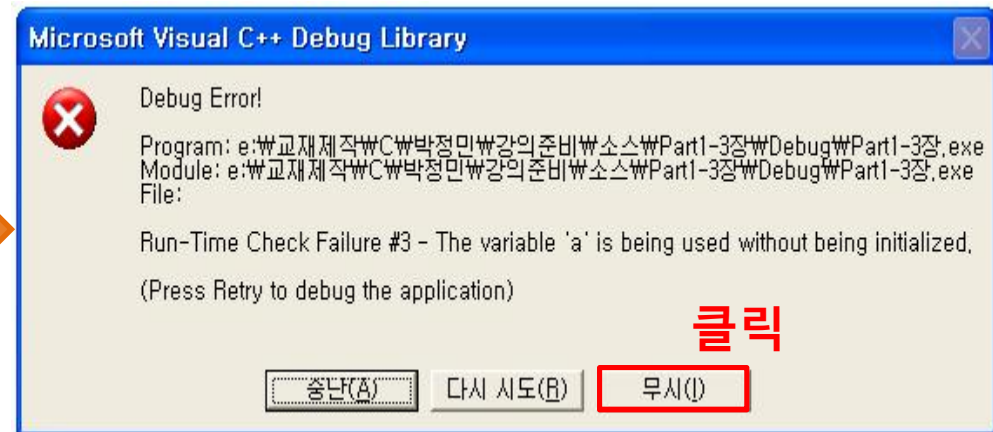
3.1 변수 선언하기 (5/7)

▶ 예제 3-4.c 실습

```
/* 3-4.c */
#include<stdio.h>
int main(void)
{
    int a;
    int b;

    printf("%d \n", a);
    printf("%d \n", b);

    return 0;
}
```



3.1 변수 선언하기 (6/7)

▶ 예제 3-5.c 실습

```
/* 3-5.c */
#include<stdio.h>
int main(void)
{
    int a=0;
    int b=1;

    printf("a의 값은 %d 입니다. \n", a);
    printf("b의 값은 %d 입니다. \n", b);

    return 0;
}
```

3.1 변수 선언하기 (7/7)

▶ 변수에 저장한 데이터는 변경될 수 있다 (변수는 임시공간)

```
/* 3-6.c */
#include<stdio.h>
int main(void)
{
    int a=0;
    int b=1;

    printf("a의 값은 %d 입니다. \n", a);
    printf("b의 값은 %d 입니다. \n", b);

    a = a+10;
    b = b+10;

    printf("변경된 a의 값은 %d 입니다. \n", a);
    printf("변경된 b의 값은 %d 입니다. \n", b);
    return 0;
}
```

3.2 변수 선언 시 주의할 점

-교재 66페이지 -

3.2 변수 선언 시 주의할 점 (1/4)

▶ 변수 선언은 제일 앞쪽에 한다

```
/* 3-7.c */
#include<stdio.h>
int main(void)
{
    int a;    // 정상
    int b;    // 정상

    a=1;
    b=2;

    int c;    // 에러 발생
    ...
}
```

3.2 변수 선언 시 주의할 점 (2/4)

▶ 변수의 이름은 의미 있게 짓는다

```
/* 3-8.c */
#include<stdio.h>
int main(void)
{
    int appleBox = 30;
    int grapeBox = 20;
    int total;

    total = appleBox + grapeBox;
    printf("총 %d 박스가 있습니다 \n", total);

    return 0;
}
```

3.2 변수 선언 시 주의할 점 (3/4)

▶ 변수의 이름을 지을 때 주의할 점

① 특수 기호, 공백 문자, 맨 처음 숫자를 사용하면 안된다

변수 이름이 올바른 경우	변수 이름이 잘못된 경우	잘못된 이유
<code>int Apple;</code>	<code>int ?apple;</code>	특수 문자 ? 사용
<code>int total;</code>	<code>int to tal;</code>	to 와 tal 사이에 공백 문자 사용
<code>int result2;</code>	<code>int 2result;</code>	맨 처음에 숫자 사용

② C언어에서 사용되는 키워드를 변수 이름에 사용하면 안된다

변수 이름이 잘못된 경우	잘못된 이유
<code>int int;</code>	키워드 int 사용
<code>int long;</code>	키워드 long 사용
<code>int short;</code>	키워드 short 사용

3.2 변수 선언 시 주의할 점 (4/4)

▶ 변수의 이름을 지을 때 주의할 점

③ C언어는 대소문자를 구분한다

대문자 변수 이름	소문자 변수 이름	설명
<code>int Apple;</code>	<code>int apple;</code>	같은 변수가 아닙니다.
<code>int TOTAL;</code>	<code>int total;</code>	같은 변수가 아닙니다.
<code>int result;</code>	<code>int result;</code>	같은 변수가 아닙니다.

④ ANSI(American National Standards Institute)에 표준화된 키워드들

auto	beak	case	char	const	continue
default	do	double	else	enum	extern
float	for	goto	if	int	long
register	return	short	signed	sizeof	static
struct	switch	typedef	union	unsigned	void
volatile	while

3.3 변수의 시작 주소와 &연산자

-교재 69페이지 -

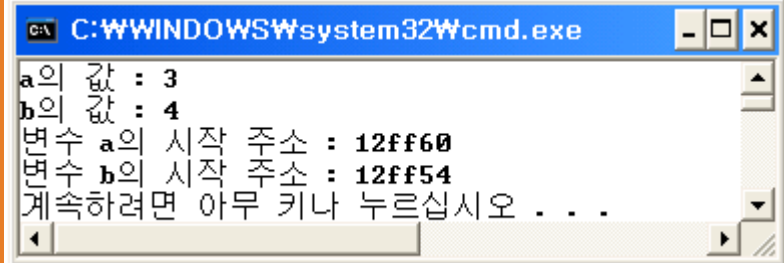
3.3 변수의 시작 주소와 &연산자

▶ 변수의 시작주소

✓ 변수 앞에 &를 붙이면 변수의 시작 주소를 알 수 있다

```
/* 3-9.c */
#include <stdio.h>
int main(void)
{
    int a=3;
    int b=4;
    printf("a의 값: %d \n", a);
    printf("b의 값: %d \n", b);

    printf("변수 a의 시작 주소: %x \n", &a);
    printf("변수 b의 시작 주소: %x \n", &b);
    return 0;
}
```



```
C:\WINDOWS\system32\cmd.exe
a의 값 : 3
b의 값 : 4
변수 a의 시작 주소 : 12ff60
변수 b의 시작 주소 : 12ff54
계속하려면 아무 키나 누르십시오 . . .
```

공부한 내용 떠올리기

- ▶ 변수란 무엇인가?
- ▶ 데이터란 무엇인가?
- ▶ 변수에 저장된 데이터는 변경 가능
- ▶ 변수를 선언하는 방법
- ▶ 변수 선언 시 주의할 점과 변수의 시작 주소

간디와 신발 (출처: 사랑과 지혜의 탈무드)

