

I 배열

• 배열의 기본

배열의 필요성

- 5개의 정수를 입력 받으려면 5개의 정수형 변수를 선언해야 함

```
int a, b, c, d, e;
int sum = 0;
double average;
scanf("%d %d %d %d %d", &a, &b, &c, &d, &e);
sum = a + b + c + d + e;
average = (double) sum / 5;
```

5개의 int형 변수를 선언한다.

- 만약 100개의 정수를 입력 받으려면???

배열(Array)

배열은 데이터들의 묶음

- 마치 학교의 수많은 학생을 학년과 반, 번호로 구분하듯 한다.

같은 자료형의 데이터를 연속적으로 메모리에 할당, 하나의 이름으로 묶는다



서로 다른 이름의 변수를
사용하는 경우



배열을 사용하는 경우

배열의 선언(1/3)

- 데이터형과 배열명, 그리고 배열의 크기가 필요함

형식	데이터형 배열명 [크기];		
예제	<div>int</div> <div>float</div> <div>short</div> <div>char</div> <div>double</div>	<div>num</div> <div>ave</div> <div>arr</div> <div>str</div> <div>x</div>	<div>[5];</div> <div>[10];</div> <div>[100];</div> <div>[80];</div> <div>[MAX];</div>
	↓ 데이터형	↓ 배열명	↓ 크기

`int arr[5];` — 크기가 5인 int 배열을 선언한다.

배열의 선언(2/3)

변수의 메모리 할당



배열의 크기는 반드시 0보다 큰 정수형 상수로 지정해야 함

```
int arr1[ ]; ○ ————— 배열의 크기가 없으므로 컴파일 에러
int arr2[0]; ○ ————— 배열의 크기가 0이므로 컴파일 에러

int size = 5;
int arr3[size]; ○ ————— 배열의 크기가 변수이므로 컴파일 에러
int arr4[size + 10]; ○ ————— 배열의 크기가 변수를 포함한 수식이므로 컴파일 에러
```

배열의 선언(3/3)

- 배열의 크기는 리터럴 상수, 매크로 상수로 지정할 수 있음

```
#define MAX 100  
int arr5[MAX];
```

매크로 상수로 배열의 크기를 지정할 수 있다.

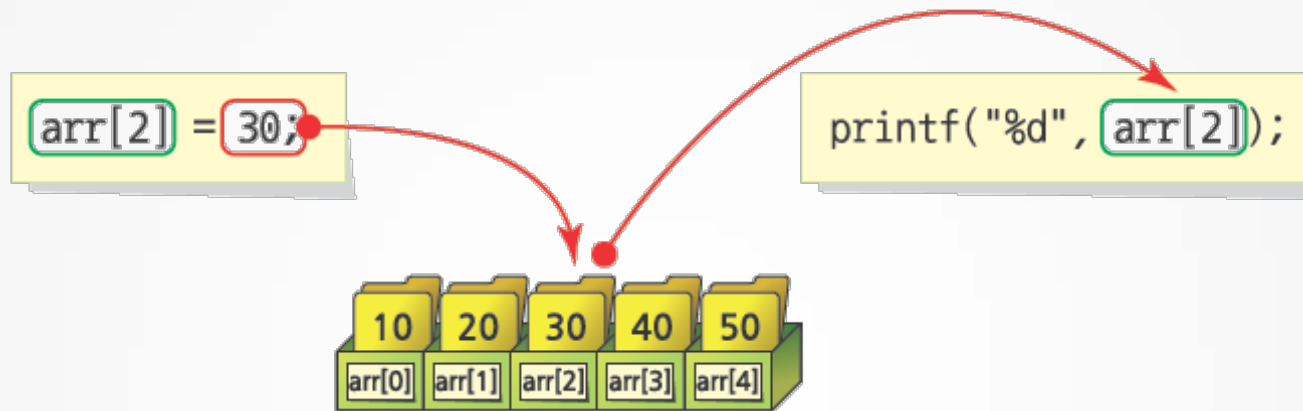
- 배열 전체의 바이트 크기를 배열 원소 하나의 바이트 크기로 나누면 배열 원소의 개수(배열의 크기)를 구할 수 있음

```
int arr[5];  
int count;  
count = sizeof(arr) / sizeof(int);  
printf("배열 원소의 개수 : %d\n", count);
```

배열 원소의 개수(배열의 크기)를 구한다.

배열의 사용(1/2)

- 배열의 각 원소에 접근하려면 첨자 또는 인덱스(index)를 이용함

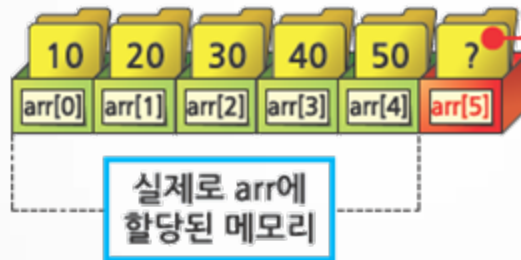


배열의 사용(2/2)

- 배열 인덱스의 유효 범위는 0 ~ (배열의 크기 - 1) 사이의 값
- 잘못된 인덱스를 사용하면 실행 에러가 발생할 수 있음

잘못된 인덱스 사용

`arr[5] = 60;`



arr에 할당된 메모리가 아니므로 메모리에 무엇이 있을지 알 수 없습니다.

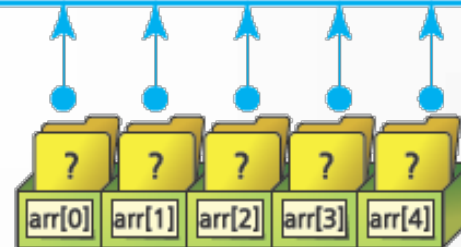
배열의 초기화

- 배열을 초기화하지 않으면 쓰레기 값을 가짐

초기화되지 않은 경우

```
int arr [5];
```

초기화되지 않은 배열은
쓰레기 값을 가집니다.



배열의 초기화

형식 데이터형 배열명 [크기]={초기값1, 초기값2, ...};

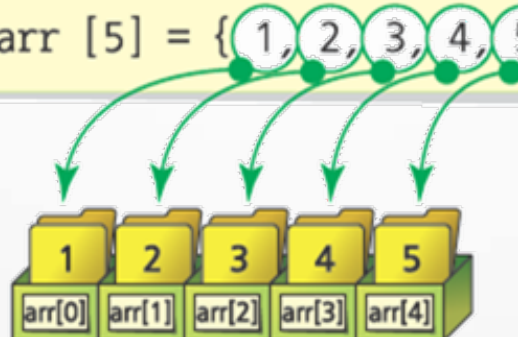
예제

```
int    w [ 5] = { 10, 20, 30, 40, 50 };
short  x [ 3] = { 1,  2,  3 };
float  y [10] = { 0.1, 0.2, 0.3 };
int    z [  ] = { 100, 200, 300, 400 };
```

배열의 초기값

배열의 초기화

```
int arr [5] = { 1, 2, 3, 4, 5 };
```



배열의 초기화

- 배열을 초기화할 때 초기값을 원소의 개수보다 부족하게 지정하면 나머지 원소를 0으로 초기화함

```
int arr[5] = { 1, 2, 3 }; ○———— arr[3]과 arr[4]는 0으로 초기화된다.  
int arr[5] = { 1, 2, 3, 4, 5, 6 }; ○———— 초기값이 더 많으므로 컴파일 에러
```

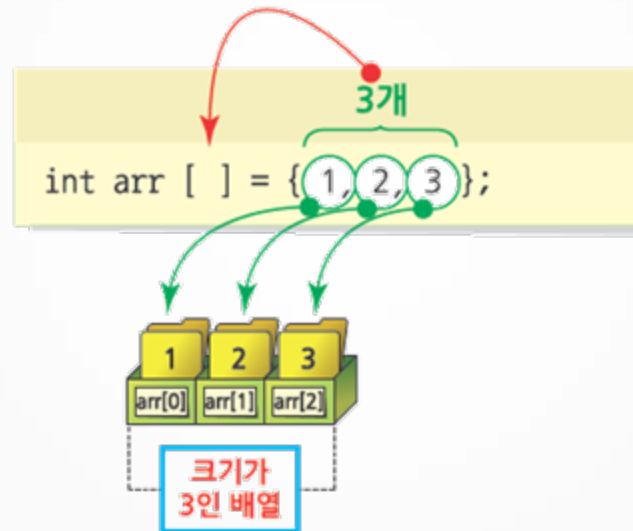
- 배열을 초기화하기 위해서 { }를 사용할 때는 반드시 초기값을 하나 이상 지정해야 함

```
int arr[5] = { }; ○———— 초기값을 지정하지 않았으므로 컴파일 에러  
int arr[5] = { 0 }; ○———— 배열 전체를 0으로 초기화한다.
```

배열의 초기화

- 배열의 초기값을 지정할 때는 배열의 크기를 생략할 수 있음

`int arr[] = { 1, 2, 3 }; ○` — 초기값이 3개이므로 크기가 3인 배열로 선언된다.
`int arr[]; ○` — 초기값 없이 배열의 크기를 생략했으므로 컴파일 에러



실습 1. 배열 활용하기

- 정수 값 10개를 배열에 저장한 다음, 그 중 양수의 개수와 음수의 개수, 그리고 0의 개수를 출력하는 프로그램을 작성하시오.
- 파일명: myArray.c