

6장 여러 데이터 한번에 저장하기 : 배열과 문자열

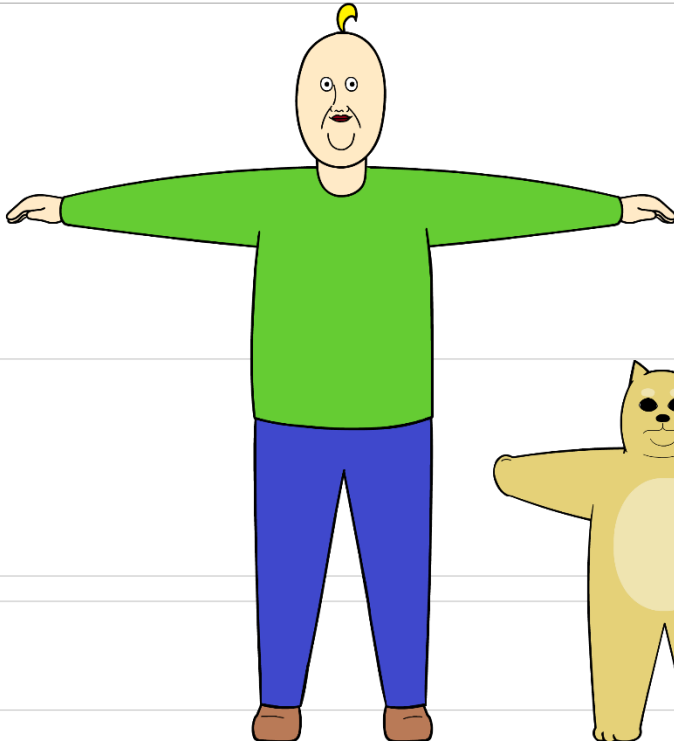
1. 이 장에서 만드는 프로그램
2. 배열이란
3. 배열 초기화하기
4. 문자열 다루기
5. 문자열 깊게 다루기
6. 프로젝트: 자라나라 머리카락



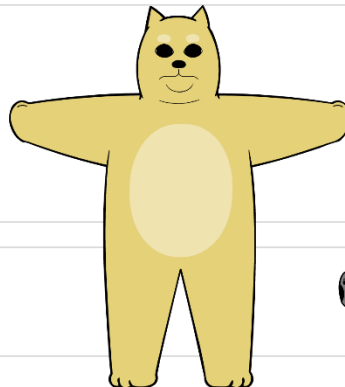
1. 이 장에서 만드는 프로그램

- 친구들의 키를 입력받고, 평균, 가장 큰 학생, 가장 작은 학생 구해보자

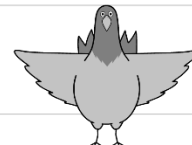
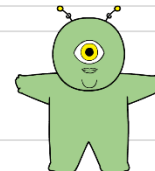
유수민



김현식



땅땅이 김덕배
새대갈



숨당이





2. 배열이란

(1) 배열이 필요한 이유

- 3칸짜리 지하철, 1호차 30명, 2호차 40명, 3호차 50명 탑승
- 정수형 변수 subway_1, subway_2, subway_3로 선언 탑승 인원 저장
- printf() 문으로 각 변수에 담긴 탑승 인원 출력하기

6.2.1 배열이필요한이유.c

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

```
int main(void) {  
    int subway_1 = 30;  
    int subway_2 = 40;  
    int subway_3 = 50;  
    printf("지하철 1호차에 %d명이 타고 있습니다.\n", subway_1);  
    printf("지하철 2호차에 %d명이 타고 있습니다.\n", subway_2);  
    printf("지하철 3호차에 %d명이 타고 있습니다.\n", subway_3);  
    return 0;  
}
```

실행결과

지하철 1호차에 30명이 타고 있습니다.
지하철 2호차에 40명이 타고 있습니다.
지하철 3호차에 50명이 타고 있습니다.



2. 배열이란

(1) 배열이 필요한 이유

- 지하철이 100호차까지 있다면?
→ 변수를 100번 선언?
- 이럴 때 배열 사용
- **배열(array)**은 동일한 자료형의 집합. 하나의 이름으로 여러 개의 데이터를 저장하는 연속된 공간



2. 배열이란

(2) 배열 선언하기

- 배열 선언: 자료형과 배열명을 적고, 대괄호([])안에 배열 크기 넣기

형식 자료형 배열명[배열크기];

- 배열명: 배열을 저장하는 메모리 공간에 붙인 이름
- 배열 크기: 연속된 공간이 몇 칸인지



2. 배열이란

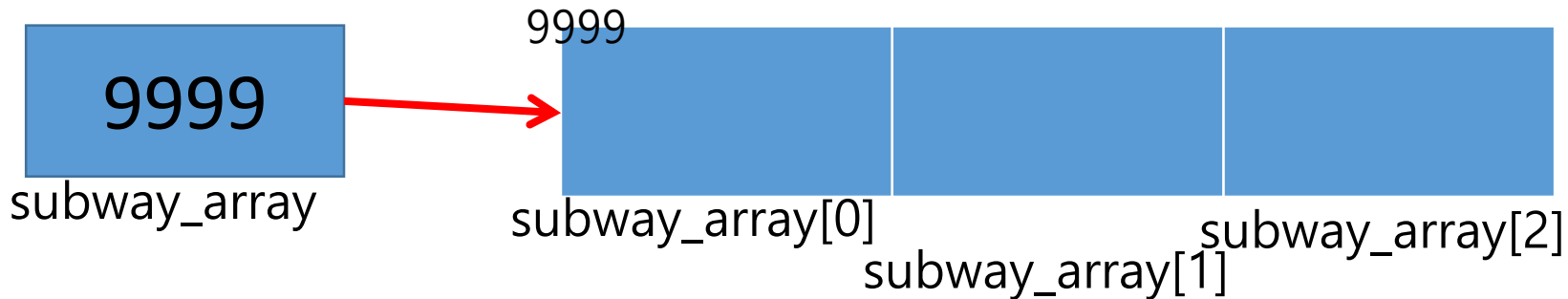
(2) 배열 선언하기

- 지하철 예제를 배열로 바꾸기
- 배열명 subway_array, 자료형 int, 배열 크기 3
- subway_array 배열은 정수형 변수 3개를 선언한 것과 같음

```
int subway_array[3];
```

자료형 배열명 배열 크기

- subway_array 배열은 공간이 3개
- 요소: 배열에서 변수 하나에 해당하는 공간
- 인덱스(index): 배열의 요소를 구분하는 번호





2. 배열이란

(2) 배열 선언하기

- 배열 초기화: 배열에 값 넣기
- 요소가 여러 개이므로 값도 여러 개
- 대입 연산자(=)로 저장

6.2.2 배열선언하기.c

```
int main(void) {  
    int subway_array[3];  
    subway_array[0] = 30; // [30][ ][ ]  
    subway_array[1] = 40; // [30][40][ ]  
    subway_array[2] = 50; // [30][40][50]  
    return 0;  
}
```

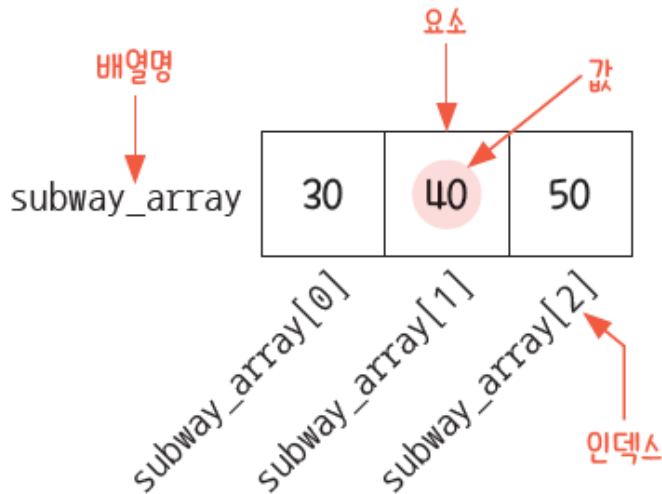



2. 배열이란

(2) 배열 선언하기

- subway_array 배열: 공간 3개에 각각 30, 40, 50이라는 값 저장

그림 6-4 배열에 값을 저장한 모습





2. 배열이란

(2) 배열 선언하기

- 배열을 선언할 때 배열 크기에는 항상 상수만 넣을 수 있다.

그림 6-5 배열 크기는 항상 상수로 선언

```
int size = 10;
```

```
int array[size]; (×)
```

```
int array[10]; (○)
```



3. 배열 초기화하기

(1) 한 번에 초기화하기

- 모든 값을 중괄호({})로 묶고 각 값을 쉼표(,)로 구분

형식

자료형 배열명[배열크기] = { 값1, 값2, ... };

- 더미 값(dummy value) 또는 쓰레기 값(garbage value)
 - 배열을 선언만 하고 초기화하지 않으면 어떤 값인지 아무도 모르는 쓰레기 값이 할당된 메모리 공간에 들어간다.
 - 쓰레기 값은 실행한 컴퓨터마다 다르게 나올 수 있다.



3. 배열 초기화하기

(2) 일부 값 초기화하기

- 크기가 10인 배열을 만들 때 일부만 초기화해 보기

6.3.2 일부값초기화하기.c

```
int main(void) {  
    int arr[10] = { 1, 2 }; // 일부 값 초기화  
    for (int i = 0; i < 10; i++) {  
        printf("%d\n", arr[i]);  
    }  
    return 0;  
}
```

| 실행결과 |
|------|
| 1 |
| 2 |
| 0 |
| (중략) |
| 0 |




3. 배열 초기화하기

(2) 일부 값 초기화하기

그림 6-6 배열 크기를 지정하지 않아도 자동으로 값의 개수를 세어 배열 생성

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



| | |
|---|---|
| 1 | 2 |
|---|---|



3. 배열 초기화하기

(3) 실수형 배열 초기화하기

- 배열 크기가 5인 실수형 배열을 만들고 값 3개 넣기

6.3.3 실수형배열초기화하기.c

```
int main(void) {  
    float arr_f[5] = { 1.0f, 2.0f, 3.0f };  
    for (int i = 0; i < 5; i++) {  
        printf("%.2f\n", arr_f[i]);  
    }  
    return 0;  
}
```

| 실행결과 | — | □ | × |
|------|---|---|---|
| 1.00 | | | |
| 2.00 | | | |
| 3.00 | | | |
| 0.00 | | | |
| 0.00 | | | |



4. 문자열 다루기

(1) 배열에 문자열 저장하기

- 값을 감쌀 때
 - 문자 : 작은따옴표로
 - 문자열 큰따옴표로
- 배열에 문자열을 저장할 때는 항상 문자열의 끝을 알리는 널 문자가 들어갈 공간을 마련해야 한다.

6.4.1 배열에 문자열 저장하기.c

```
int main(void) {  
    char str[7] = "coding"; // [c][o][d][i][n][g][\0]  
    printf("%s\n", str);  
    return 0;  
}
```

실행결과

coding

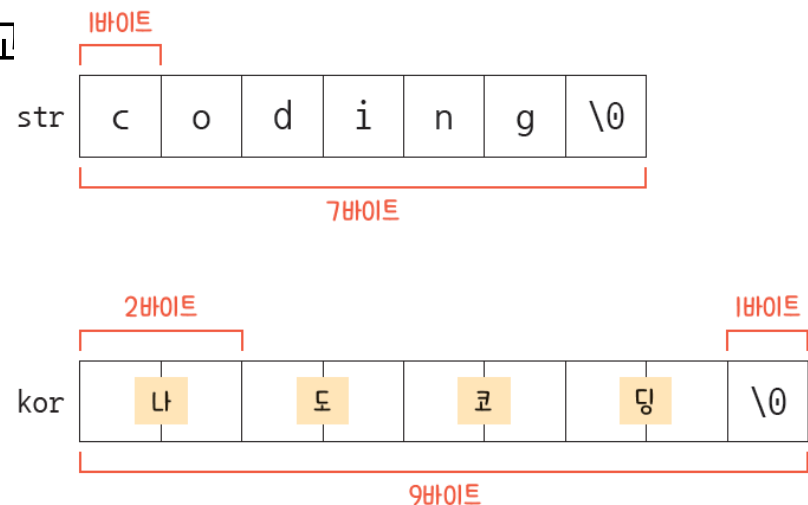


4. 문자열 다루기

(2) 배열에 한글 저장하기

- 알파벳과 한글이 차지하는 저장 공간의 크기가 다르다.
 - 알파벳과 숫자 : 한 글자에 1바이트
 - 한글 : 한 글자에 2바이트

그림 6-7 str 배열과 kor 배열 비교





5. 문자열 깊게 다루기

(1) 문자열 배열과 널 문자

- 문자를 하나씩 저장하든 문자열로 저장하든 문자열의 마지막에는 널 문자가 들어간다.

6.5.1 문자열배열과널문자.c

```
int main(void) {  
    char c_array[10] = { 'c', 'o', 'd', 'i', 'n', 'g' };  
    for (int i = 0; i < sizeof(c_array); i++) {  
        printf("%d\n", c_array[i]); // %c를 %d로 수정  
    }  
    return 0;  
}
```

실행결과

```
99  
111  
100  
105  
110  
103  
0  
0  
0  
0
```



5. 문자열 깊게 다루기

(2) 문자열 입력받기

- 배열 크기를 넉넉히 선언하는 편이 좋다.

6.5.2 문자열입력받기.c

```
int main(void) {  
    char name[256];  
    printf("이름이 뭐예요? ");  
    scanf_s("%s", name, sizeof(name));  
    printf("%s\n", name);  
    return 0;  
}
```

실행결과

이름이 뭐예요? 홍길동
홍길동



5. 문자열 깊게 다루기

(3) 아스키코드

- 미국표준협회(ANSI, American National Standards Interchange)에서 제시한 표준 코드 체계
- 전체 아스키코드 표 <https://ko.wikipedia.org/wiki/ASCII>



6. 프로젝트: 동물친구들의 키

다섯 동물(유수민, 김현식, 땡땡이, 새대갈, 숨당이)들의 키를 입력받아 평균 및 최단신, 최단신 구하기

- ⊖ for 문에 idx가 0부터 4까지 1씩 증가하는 조건 작성
 - ⊖ 동물친구들의 키 입력받기
 - ⊖ 누적하기
- ⊖ 누적된 값을 이용해 평균구하기
- ⊛ 최장키값 = 0, 최단키값 = 999로 초기화
- ④ for문 idx가 0부터 4까지 1씩 증가하는 조건 작성
 - ⊖ arrHeight[idx]가 최장키값보다 크면 최장키값으로
 - ⊖ arrHeight[idx]가 최단키값보다 작으면 최단키값으로



6. 프로젝트: 동물친구들의 키

실행화면

유수민의 키는 ? 180

김현식의 키는 ? 185

땅땅이의 키는 ? 125

새대갈의 키는 ? 123

숨당이의 키는 ? 105

scanf를 이용한 입력

- 평균키 : 143.60

- 최장신 친구 김현식의 키는 185

- 최단신 친구 숨당이의 키는 105