

# 배열

2025 프로그래밍 기초

# 목표

- 배열을 이해하고 사용
- 배열에 데이터를 저장하고 조건 비교
- 야구게임 제작

(연습문제)

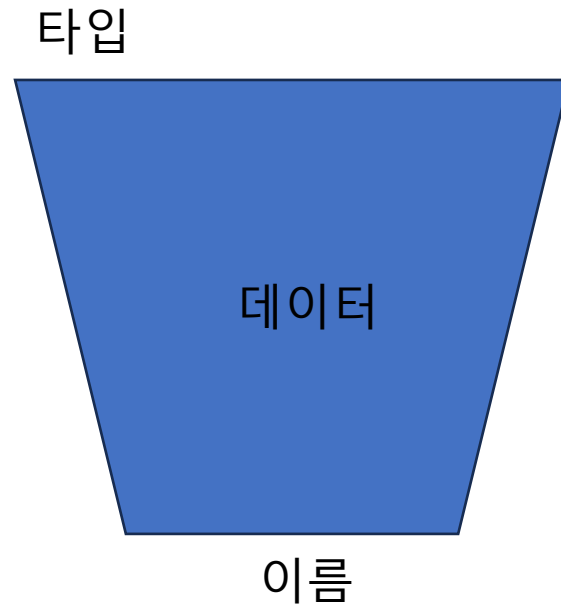
게임 시작(1)을 선택하면 랜덤숫자 3개를 생성하여 출력하고, 종료(2)를 선택하면 Exit을 출력하고 프로그램을 종료하는 프로그램을 완성하시오.



```
PS C:\Users\Juhong\OneDrive - 주식회사 랑데뷰\바탕
1: Game Start
2: Exit
1
Random : 6, 3, 5
1: Game Start
2: Exit
1
Random : 3, 8, 4
1: Game Start
2: Exit
2
Exit
PS C:\Users\Juhong\OneDrive - 주식회사 랑데뷰\바탕
```

# 변수

- 값을 저장해놓는 기억공간(RAM, 메모리)



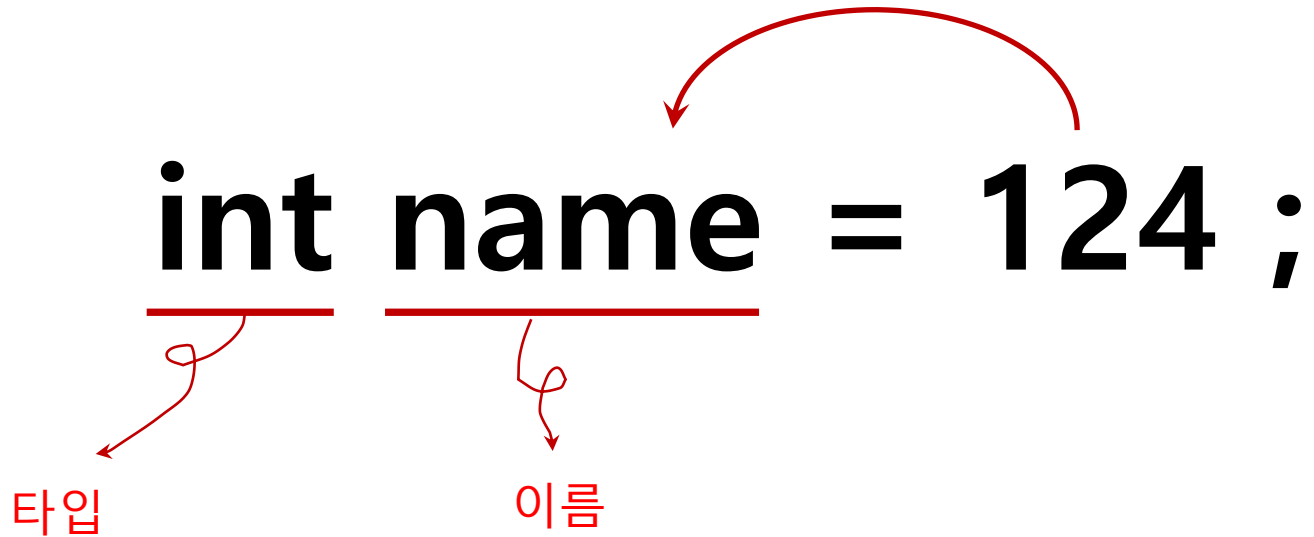
# 변수 타입

구분	자료형	크기	데이터 범위	부호
문자형(정수형)	char	1바이트	-128 ~ 127	양수 음수
정수형	short	2바이트	-32768 ~ 32767	
	int	4바이트	약 +-21억(-2147483648 ~ ..7)	
	long	4바이트	약 +-21억(-2147483648 ~ ..7)	
문자형(정수형)	unsigned char	1바이트	0 ~ 255	양수
정수형	unsigned short	2바이트	0 ~ 65535	
	unsigned int	4바이트	0 ~ 약 43억	
	unsigned long	4바이트	0 ~ 약 43억	
실수형	float	4바이트	$3.4 \times 10^{-38} \sim 3.4 \times 10^{38}$	양수 음수
	double	8바이트	$1.7 \times 10^{-308} \sim 1.7 \times 10^{308}$	

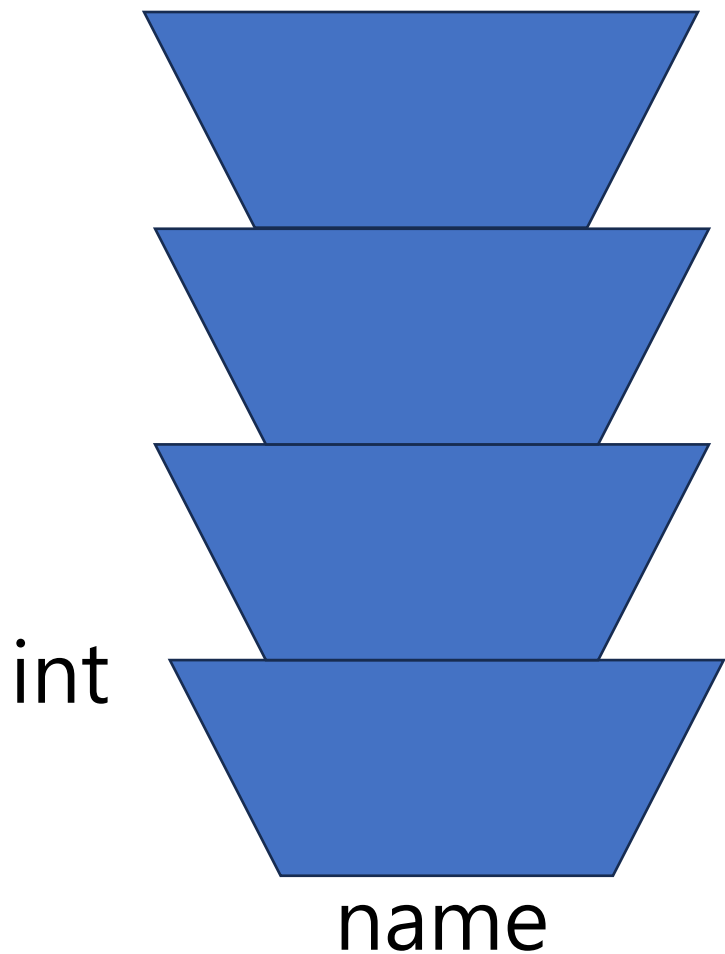
# 정수형 변수를 선언하고 데이터를 저장

int name = 124 ;

타입                  이름

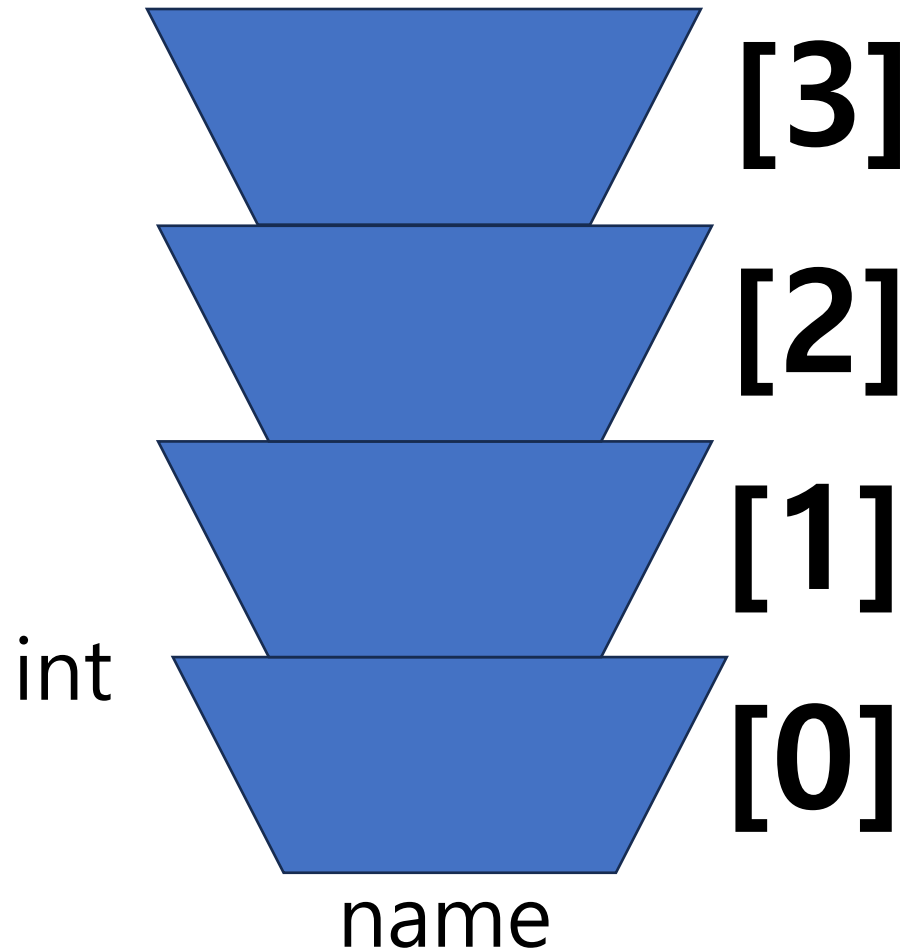
The diagram shows the code 'int name = 124 ;'. The word 'int' is underlined with a red line, and a red arrow points from this underline to the Korean word '타입' (type) below it. The word 'name' is also underlined with a red line, and a red arrow points from this underline to the Korean word '이름' (name) below it. A red curved arrow points from the space between 'name' and '=' to the space between 'int' and 'name', indicating the association between the type and the variable name.

배열



```
int name[4] ;
```

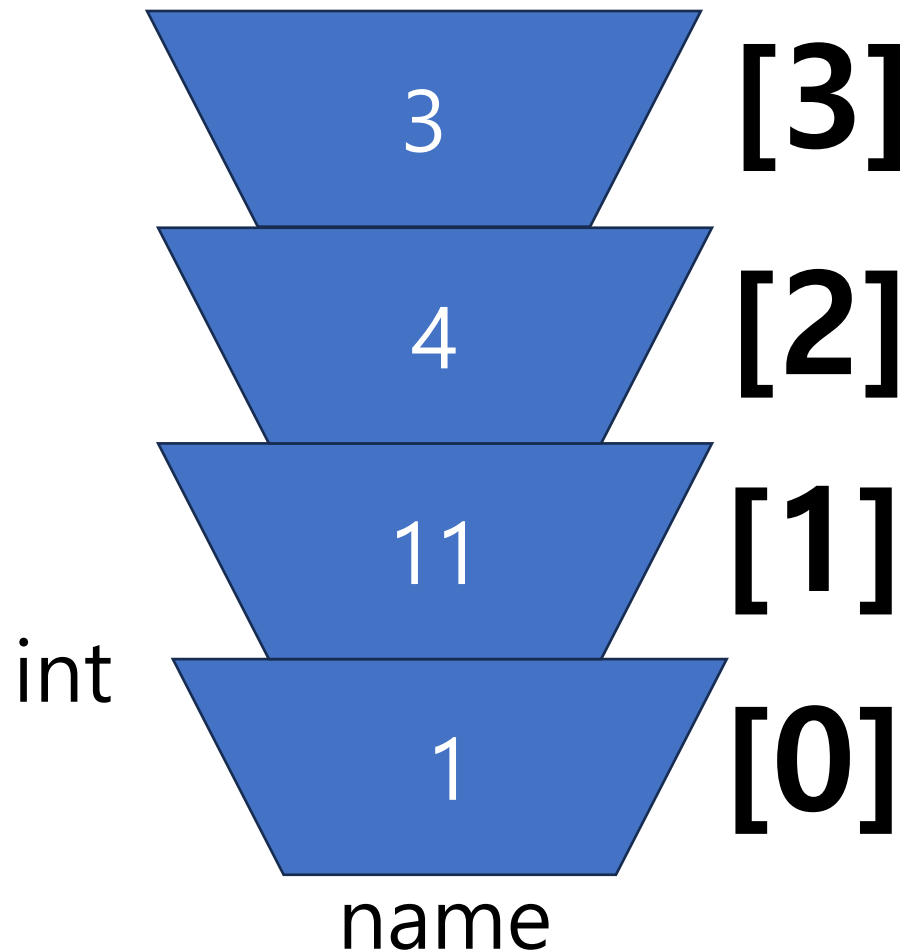
배열



```
int name[4] ;
```



배열



```
int name[4] ;
```

```
name[0] = 1 ;  
name[1] = 11 ;  
name[2] = 4 ;  
name[3] = 3 ;
```

# 배열

## int(정수)형 배열을 만들고 임의의 숫자를 생성

```
test.cpp
파일 편집 보기
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main(void)
{
    srand( time(NULL) );

    int number[5];

    number[0] = rand() % 10 ;
    number[1] = rand() % 10 ;
    number[2] = rand() % 10 ;
    number[3] = rand() % 10 ;
    number[4] = rand() % 10 ;

    printf("number[0] = %d\n", number[0]);
    printf("number[1] = %d\n", number[1]);
    printf("number[2] = %d\n", number[2]);
    printf("number[3] = %d\n", number[3]);
    printf("number[4] = %d\n", number[4]);

    return 0 ;
}
```

컴파일 → **> C:\mingw64\bin\gcc -o test test.cpp**

실행 → **> ./test**

### 실행 결과

```
PS C:\Users\Juhong\OneDrive
number[0] = 6
number[1] = 7
number[2] = 8
number[3] = 8
number[4] = 8
PS C:\Users\Juhong\OneDrive
```



배열을 이용하여  
1000개의 랜덤한 숫자를 만들고  
아래 같이 출력하는 코드를 작성하시오

```
test.cpp
파일 편집 보기

#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main(void)
{
    srand( time(NULL) );

    int number[1000];

    for( int i=0 ; i<1000 ; i++ )
    {
        [REDACTED]
    }

    return 0 ;
}
```

줄 21, 열 1 | 252자 | 일반 텍스트 | 120% | Windows (CRLF) | UTF-8

```
Windows PowerShell
number[971] = 2
number[972] = 4
number[973] = 3
number[974] = 3
number[975] = 8
number[976] = 7
number[977] = 9
number[978] = 5
number[979] = 6
number[980] = 5
number[981] = 9
number[982] = 5
number[983] = 2
number[984] = 7
number[985] = 8
number[986] = 7
number[987] = 4
number[988] = 9
number[989] = 7
number[990] = 0
number[991] = 2
number[992] = 8
number[993] = 8
number[994] = 5
number[995] = 3
number[996] = 3
number[997] = 3
number[998] = 9
number[999] = 7
PS C:\Users\Juhong\OneDrive - 주식회사 랑데뷰\바탕 화면\work> |
```

# 숫자야구 게임 만들기



- 1) 세 자리 수를 정합니다.
  - 2) 짝이 말한 숫자와 내가 정한 숫자를 비교해봅니다.
  - 3) 다음 규칙에 따라 짝에게 말합니다. 짝과 번갈아가며 숫자를 맞혀봅니다.
    - \* 스트라이크(S): 자리와 숫자가 모두 일치하는 경우
    - \* 볼(B): 숫자가 같되 자리가 틀리는 경우
    - \* 아웃(O): 그 어떤 숫자도 내가 정한 숫자와 다른 경우
- 예) 내가 정한 숫자가 287이라면 다음과 같습니다.

3	6	9	아웃	S	○○	B	○○○	아웃	●
6	7	2	2볼	S	○○	B	●●○	아웃	○
2	8	0	2스트라이크	S	●●	B	○○○	아웃	○
2	7	8	1스트라이크 2볼	S	●○	B	●●○	아웃	○
2	8	7	홈런! (승리)	S	○○	B	○○○	아웃	○
								홈런	●

- 4) 먼저 숫자를 맞힌 사람이 이깁니다.

# 숫자야구게임

int(정수)형 배열을 이용하여 임의의 숫자 3개를 생성

```
test.cpp
파일 편집 보기
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main(void)
{
    srand( time(NULL) );

    int number[3];

    for( int i=0 ; i<3 ; i++ )
    {
        number[i] = rand() % 10 ;
    }

    printf("baseball : %d, %d, %d\n", number[0], number[1], number[2] );

    return 0 ;
}
```

컴파일 → `> C:\mingw64\bin\gcc -o test test.cpp`

실행 → `> ./test`

실행 결과

```
PS C:\Users\Juhong\One
baseball : 7, 7, 4
PS C:\Users\Juhong\One
```

야구 게임에서 똑같은 숫자가 반복되면 안됨!!

# 숫자야구게임

**서로 다른** 임의의 숫자 3개를 생성 해보자!

# 반복문 : do~while

- 언제사용?) 동일한 코드(명령)를 특정 조건일때 **반복**하고 싶다.

```
while( 조건 )  
{  
    //반복할 코드(명령)  
    ....  
}
```

# 반복문 : do~while

- 언제사용?) 동일한 코드(명령)를 특정 조건일때 **반복**하고 싶다.

```
do
{
    //반복할 코드(명령)
    ....
} while( 조건 )
```



# 반복문 : do~while

- 언제사용?) 동일한 코드(명령)를 특정 조건일때 **반복**하고 싶다.

**do**

{

//반복할 코드(명령)

....

} **while**( **조건** )

조건식

A < B  
A <= B  
A > B  
A >= B  
A == B  
A != B

조건식 조합

- OR → 조건식1 || 조건식2
- AND → 조건식1 && 조건식2

# 숫자야구게임

**서로 다른** 임의의 숫자 3개를 생성 해보자!

```
test.cpp
파일 편집 보기

#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main(void) ₩
{
    int a, b, c;
    srand(time(NULL));

    do {
        a = rand() % 10 + 1;
        b = rand() % 10 + 1;
        c = rand() % 10 + 1;
    } while (a == b || b == c || a == c);

    printf("baseball : %d %d %d₩n", a, b, c);

    return 0;
}
```

줄 23, 열 1 323자 일반 텍스트 120% Windows (CRLF) UTF-8

컴파일 → **> C:\mingw64\bin\gcc -o test test.cpp**

실행 → **> ./test**

실행 결과

```
PS C:\Users\Juhong\OneDrive - 주4
baseball : 7, 7, 4
PS C:\Users\Juhong\OneDrive - 주4
PS C:\Users\Juhong\OneDrive - 주4
baseball : 1 6 4
PS C:\Users\Juhong\OneDrive - 주4
baseball : 8 5 4
PS C:\Users\Juhong\OneDrive - 주4
baseball : 8 5 4
PS C:\Users\Juhong\OneDrive - 주4
baseball : 4 2 7
PS C:\Users\Juhong\OneDrive - 주4
baseball : 1 4 3
PS C:\Users\Juhong\OneDrive - 주4
baseball : 4 9 1
PS C:\Users\Juhong\OneDrive - 주4
```

# 숫자야구게임

**서로 다른** 임의의 숫자 3개를 생성 해보자!

```
test.cpp
파일 편집 보기

int main(void) ₩
{
    int ball[3];
    srand(time(NULL));

    do {
        ball[0] = rand() % 10 + 1;
        ball[1] = rand() % 10 + 1;
        ball[2] = rand() % 10 + 1;
    } while (ball[0] == ball[1] || ball[0] == ball[2] || ball[1] == ball[2]);

    printf("baseball : %d %d %d₩n", ball[0], ball[1], ball[2]);

    return 0;
}
```

줄 22, 열 1 | 395자 | 일반 텍스트 | 120% | Windows (CRLF) | UTF-8

컴파일 → **> C:\mingw64\bin\gcc -o test test.cpp**

실행 → **> ./test**

실행 결과

```
PS C:\Users\Juhong\OneDrive - 주4  
baseball : 7, 7, 4  
PS C:\Users\Juhong\OneDrive - 주4  
PS C:\Users\Juhong\OneDrive - 주4  
baseball : 1 6 4  
PS C:\Users\Juhong\OneDrive - 주4  
baseball : 8 5 4  
PS C:\Users\Juhong\OneDrive - 주4  
baseball : 8 5 4  
PS C:\Users\Juhong\OneDrive - 주4  
baseball : 4 2 7  
PS C:\Users\Juhong\OneDrive - 주4  
baseball : 1 4 3  
PS C:\Users\Juhong\OneDrive - 주4  
baseball : 4 9 1  
PS C:\Users\Juhong\OneDrive - 주4
```

# 숫자야구게임

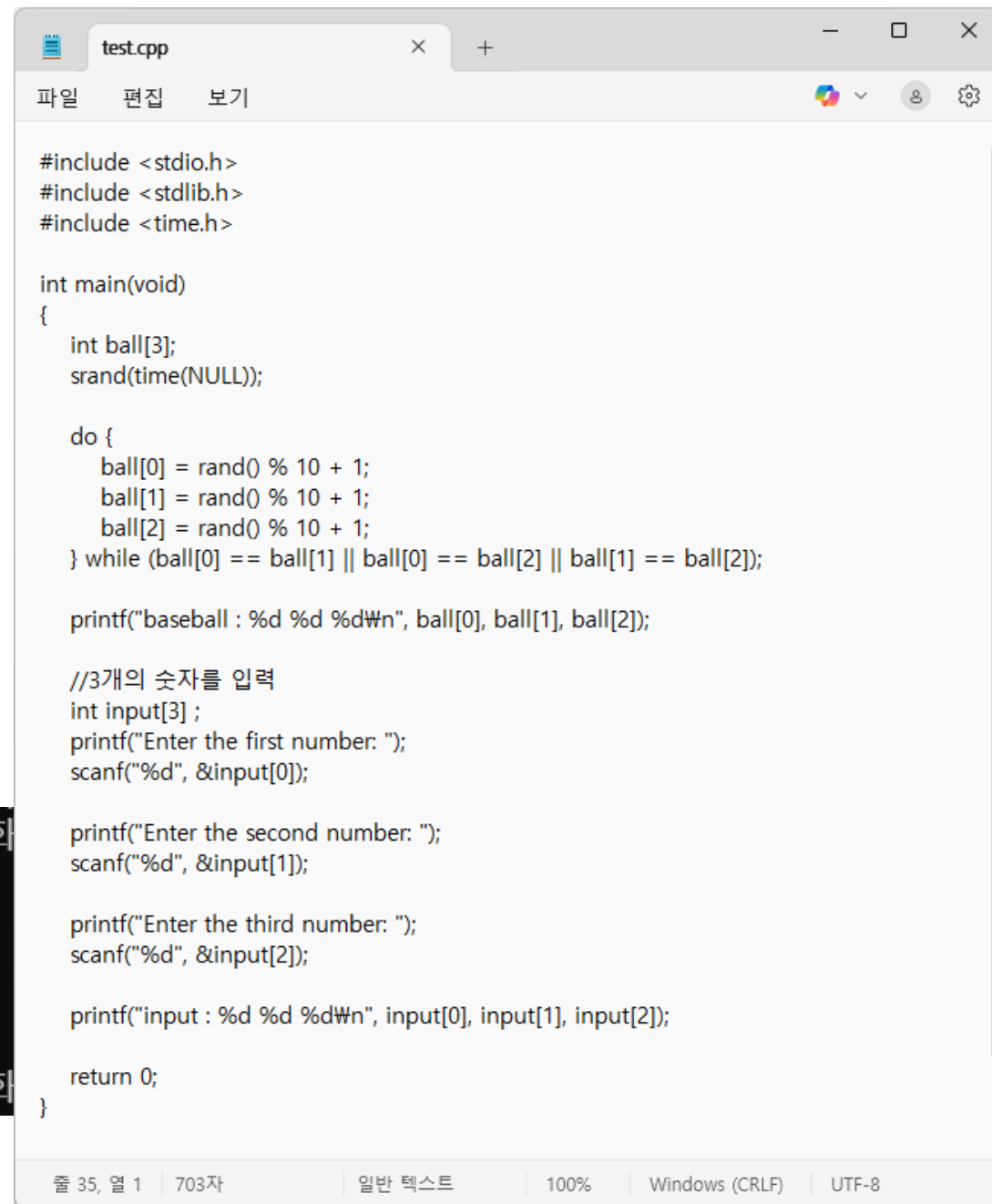
## 3개의 숫자를 입력

컴파일 → `> C:\mingw64\bin\gcc -o test test.cpp`

실행 → `> ./test`

### 실행 결과

```
PS C:\Users\Juhong\OneDrive - 주식회사 랑데뷰\바탕 화면> gcc test.cpp
baseball : 5 7 9
Enter the first number: 3
Enter the second number: 6
Enter the third number: 7
input : 3 6 7
PS C:\Users\Juhong\OneDrive - 주식회사 랑데뷰\바탕 화면>
```



```
test.cpp
파일 편집 보기
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main(void)
{
    int ball[3];
    srand(time(NULL));

    do {
        ball[0] = rand() % 10 + 1;
        ball[1] = rand() % 10 + 1;
        ball[2] = rand() % 10 + 1;
    } while (ball[0] == ball[1] || ball[0] == ball[2] || ball[1] == ball[2]);

    printf("baseball : %d %d %d\n", ball[0], ball[1], ball[2]);

    //3개의 숫자를 입력
    int input[3];
    printf("Enter the first number: ");
    scanf("%d", &input[0]);

    printf("Enter the second number: ");
    scanf("%d", &input[1]);

    printf("Enter the third number: ");
    scanf("%d", &input[2]);

    printf("input : %d %d %d\n", input[0], input[1], input[2]);

    return 0;
}
```

줄 35, 열 1 | 703자 | 일반 텍스트 | 100% | Windows (CRLF) | UTF-8

# 숫자야구게임

3개의 숫자를 비교하여 모두 동일하면 Home Run~ 출력



```
PS C:\Users\Juhong\OneDrive - 주  
baseball : 8 1 5  
Enter the first number: 8  
Enter the second number: 1  
Enter the third number: 5  
input : 8 1 5  
Home Run~  
PS C:\Users\Juhong\OneDrive - 주
```