

If-else 문

```
if ( height >= 180 )  
    cout << “합격입니다.\n”;  
  
else  
    cout << “불합격입니다.\n”;
```



```
if( height >= 175 )  
    if( score >= 90 )  
        cout << “박보검입니다.\n”;  
    else  
        cout << “이광수입니다n”;
```

Switch

```
int number;
cout << "정수를 입력하시오:";
cin >> number;
switch(number)
{
    case 0:
        cout << "없음\n";
        break ;
    case 1:
        cout << "하나\n";
        break ;
    case 2:
        cout << "둘\n";
        break ;
    default:
        cout << "많음\n";
        break;
}
```

정수를 입력하시오: 1
하나

Switch

Q. '0'을 입력 했을때 다음 프로그램의 출력 결과는?

```
int number;
cout << "정수를 입력하시오:";
cin >> number;
switch(number)
{
    case 0:
        cout << "없음\n";
    case 1:
        cout << "하나\n";
    case 2:
        cout << "둘\n";
    default:
        cout << "많음\n";
        break;
}
```

없음
하나
둘
많음

While

```
// while 문을 이용한 구구단 출력 프로그램
#include <iostream>
using namespace std;
int main()
{
    int n;
    int i = 1;

    cout << "구구단 중에서 출력하고 싶은 단을 입력하시오: ";
    cin >> n;
    while (i <= 9){
        cout << n << "*" << i << "=" << n * i << endl;
        i++;
    }
    return 0;
}
```

```
구구단 중에서 출력하고 싶은 단을 입력하시오: 9
9*1 = 9
9*2 = 18
9*3 = 27
....
9*9 = 81
```

do...while문

- 반복 조건을 루프의 끝에서 검사

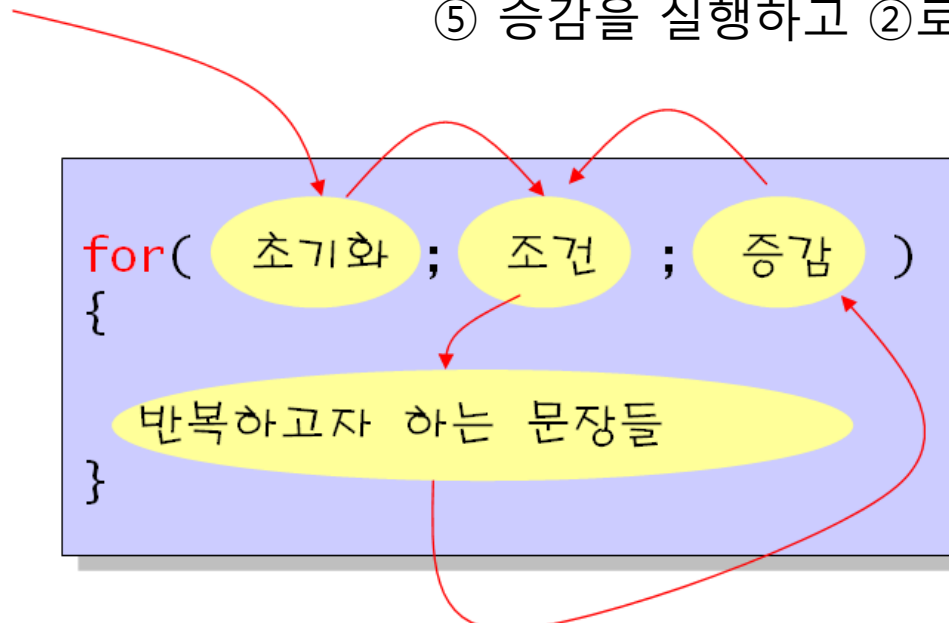
```
int main()
{
    int i = 10;
    do {
        cout << "i의 값: " << i << endl;
        i++;
    } while (i < 3);
}
```

i의 값: 10

for 문의 구조

```
for ( 초기화; 조건식; 증감식 )  
    문장;
```

- ① 초기화를 실행한다.
- ② 반복 조건을 나타내는 조건식을 계산한다.
- ③ 수식의 값이 거짓이면 for 문의 실행이 종료된다.
- ④ 수식의 값이 참이면 문장이 실행된다.
- ⑤ 증감을 실행하고 ②로 돌아간다.



for 예제

sum.cpp

```
1. #include <iostream>
2. using namespace std;
3.
4. int main()
5. {
6.     int sum = 0;
7.
8.     for (int i = 1; i<= 10; i++)
9.         sum += i;
10.
11.     cout << "1부터 10까지의 정수의 합 = " << sum << endl;
12.     return 0;
13. }
```

for 문안에서도 변수를 선언할 수 있다.

실행 결과

1부터 10까지의 정수의 합 = 55

break 문과 continue 문

- break 문은 반복 루프를 빠져 나오는데 사용된다.
- continue 문은 현재 수행하고 있는 반복 과정의 나머지를 건너뛰고 다음 반복 과정을 강제로 시작

break 문과 continue 문 예제

```
// 소문자를 대문자로 변경한다.
```

```
#include <iostream>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    char letter;
```

```
    while(1)
```

```
    {
```

```
        cout << "소문자를 입력하시오: ";
```

```
        cin >> letter;    // 공백 문자 제외
```

```
        if( letter == 'Q' )
```

```
            break;
```

```
        if( letter < 'a' || letter > 'z' )
```

```
            continue;
```

```
        letter -= 32;
```

```
        // 소문자 -> 대문자
```

```
        cout << "변환된 대문자는 " << letter << "입니다.\n";
```

```
    }
```

```
    return 0;
```

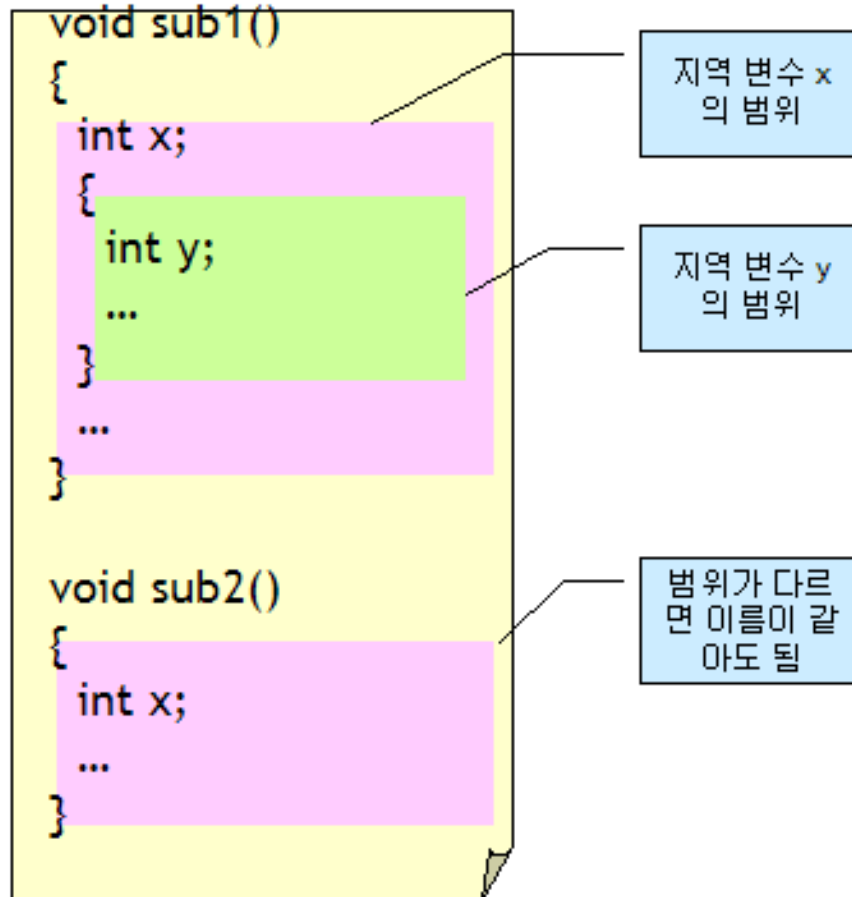
```
}
```

break는 반복문을 탈출한다.

continue는 다음 반복을 시작한다.

지역 변수

- 지역 변수(local variable): 블록 안에서 선언되는 변수



전역 변수

- 전역 변수(global variable): 함수의 외부에 선언되는 변수

```
#include <iostream>
using namespace std;
```

```
int x = 1;    // 전역 변수
```

```
void sub()
```

```
{
    x++;
```

```
}
```

```
int main(void)
```

```
{
```

```
    x++;
```

```
    ...
```

```
}
```

전역 변수는 어디서나 접근이 가능하다.

#실습 저장 유형 지정자 static

```
#include <iostream>
using namespace std;
```

```
void sub(void)
{
```

```
    int i = 0;
```

```
    static int s = 0;
```

```
        i++;
```

```
        s++;
```

```
        cout << "i: " << i << " s: " << s << endl;
```

```
}
```

```
int main()
```

```
{
```

```
    sub();
```

```
    sub();
```

```
    sub();
```

```
    return 0;
```

```
}
```

Q. 변수 i의 값의 변화는? 1 1 1

Q. 변수 s의 값의 변화는? 1 2 3

static을 붙이면 지역 변수가
정적 변수로 된다.