

재귀함수 - 반복문 개념

```
void recursive_func()
{
    printf("재귀함수입니다.\n");
    recursive_func();
}

int main()
{
    recursive_func();
    return 0;
}
```

```
int main()
{
    while(1)
    {
        printf("반복문입니다.\n");
    }

    return 0;
}
```

재귀함수 - 종료조건

```
void recursive_func(int n)
{
    if(n==0) return; //종료 조건
    printf("재귀함수입니다. %d\n", n);
    recursive_func(n-1);
}

int main()
{
    recursive_func(10);
    return 0;
}
```

```
int main()
{
    int n = 10;
    while(1)
    {
        if(n==0) break; //종료 조건
        printf("반복문입니다. %d\n", n);
        n--;
    }

    return 0;
}
```

재귀함수 - 재귀 알고리즘을 통한 문제해결

1~5까지의 합

= 5 더하기 1~4까지의 합

= 4 더하기 1~3까지의 합

= 3 더하기 1~2까지의 합

= 2 더하기 1~1까지의 합

= 1~1까지의 합은 1 //끝

```
int sum1to5(int n)
{
    if(n == 1) //끝, 종료조건
    {
        printf("sum 1 to %d = %d\n", n, n);
        return 1; //재귀함수 호출 안 함
    }
    int sum = n + sum1to5(n-1);
    printf("sum 1 to %d = %d + sum 1 to %d\n", n, n, n-1);
    return sum;
}

void main()
{
    int sum = sum1to5(5); //시작
    printf("\n\n sum 1 to 5 = %d\n", sum);
}
```

재귀함수 - 함수호출흐름

```
int sum = sum1to5(10); //시작-main함수

int sum = n + sum1to5(5-1); //재귀-sum1to5 함수, n=5
printf("sum1 to %d = %d + sum1 to %d\n", n, n, n-1);
return sum;

int sum = n + sum1to5(4-1); //재귀-sum1to5 함수, n=4
printf("sum1 to %d = %d + sum1 to %d\n", n, n, n-1);
return sum;

int sum = n + sum1to5(3-1); //재귀-sum1to5 함수, n=3
printf("sum1 to %d = %d + sum1 to %d\n", n, n, n-1);
return sum;

int sum = n + sum1to5(2-1); //재귀-sum1to5 함수, n=2
printf("sum1 to %d = %d + sum1 to %d\n", n, n, n-1);
return sum;

if(n == 1) //재귀종료-sum1to5 함수, n=1
{
    printf("sum1 to %d = %d\n", n, n); //출력
    return 1; //재귀함수 호출 안 함
}
```