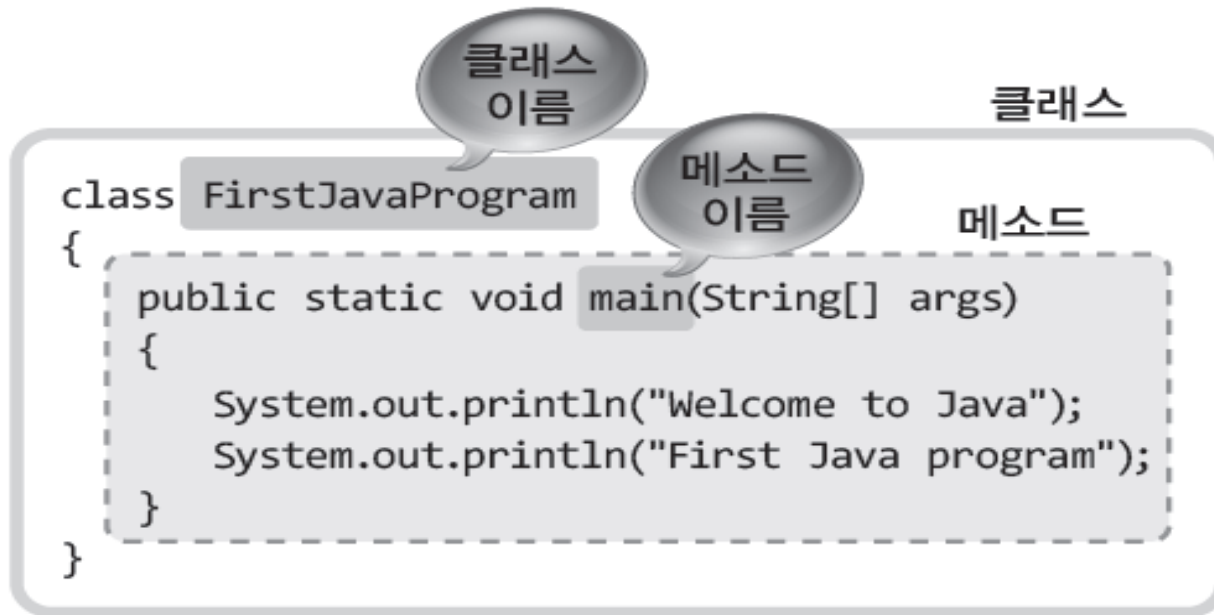


Method

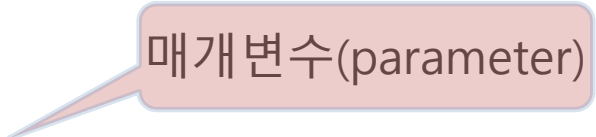
# Method의 기본 구조



- 자바 프로그램의 시작은 main 이라는 이름의 메소드를 실행하는 데서 부터 시작.
- Method의 중괄호 내에 존재하는 문장들이 위에서 아래로 순차적으로 실행.

# Method의 정의

```
1. class MethodDefAdd {  
2.     public static void main(String[] args) {  
3.         System.out.println("프로그램의 시작");  
4.         hiEveryone(12);  
5.         hiEveryone(13);  
6.         system.out.println("프로그램의 끝");  
7.     }  
8.  
9.     public static void hiEveryone(int age) {  
10.        System.out.println("좋은 아침입니다.");  
11.        System.out.println("제 나이는 "+ age+" 세입니다.");  
12.    }  
13. }
```



매개변수(parameter)

# Method 호출문

- 해당 메소드의 실행 완료 후 다음 문장 실행.
- 매개변수(parameter).
- 메소드의 실행방법과 데이터 전달 방법.
- 메소드는 한번 정의되면 여러 번 실행 가능.
- 메소드 정의 위치는 프로그램에 영향을 미치지 않음.

# Exam. – parameter

```
1. class MethodParam {  
2.     public static void main(String[] args) {  
3.         double myHeight = 175.9;  
4.         hiEveryone(12, 12.5);  
5.         hiEveryone(13, myHeight);  
6.         byEveryone();  
7.     }  
8.     public static void hiEveryone(int age, double height) {  
9.         System.out.println("좋은 아침입니다.");  
10.        System.out.println("제 나이는 "+ age+" 세입니다.");  
11.    }  
12.    public static void byEveryone() {  
13.        System.out.println("다음에 뵙겠습니다.");  
14.    }  
15. }
```

# Exam. – return

```
1. class MethodReturn {
2.     public static void main(String[] args) {
3.         int result = adder(4, 5);
4.         System.out.println("4 + 5 = " + result);
5.         System.out.println("3.5의 제곱 : " + square(3.5));
6.     }
7.     public static int adder(int num1, int num2) {
8.         int addResult = num1 + num2;
9.         return addResult;
10.    }
11.    public static double square(double num) {
12.        return num * num;
13.    }
14. }
```

# return

## 값의 반환 / 메소드의 종료

```
1. class OnlyExitReturn {
2.     public static void main(String[] args) {
3.         divide(4, 2);
4.         divide(6, 2);
5.         divide(9, 0);
6.     }
7.     public static void divide(int num1, int num2) {
8.         if(num2 == 0) {
9.             System.out.println("0으로는 값을 나눌 수 없습니다.");
10.            return;
11.        }
12.        System.out.println("나눗셈 결과 : " + (num1 / num2));
13.    }
14. }
```

# 변수의 범위 (1/2)

```
1. class ScopeVariable {  
2.     public static void main(String[] args) {  
3.         boolean scope = true;  
4.         if(scope) {  
5.             int num = 1;  
6.             num++;  
7.             System.out.println(num);  
8.         } else {  
9.             int num = 2;  
10.            System.out.println(num);  
11.        }  
12.        simple();  
13.    }  
14.    public static void simple() {  
15.        int num = 3;  
16.        System.out.println(num);  
17.    }  
18. }
```



## 변수의 범위(2/2)

- ```
for(int num = 0; num < 5; num++)  
{  
    /* 추가적인 변수 num 선언 불가 영역 */  
}
```
- ```
public static void myFunc(int num)  
{  
    /* 추가적인 변수 num 선언 불가 영역 */  
}
```
- 지역변수는 선언된 지역을 벗어나 버리면 메모리 공간에서 소멸됨.