



# JAVA 프로그래밍

## Chapter 06 메소드

# 목차

- ❖ 메소드의 정의 및 용법
- ❖ 메소드의 종류
- ❖ 결과형 리턴값

# 메소드의 정의

## ❖ 메소드

- 어떤 작업을 수행하기 위한 명령문의 집합.
- 어떤 값(패러미터)을 받아서 처리하고 그 결과(리턴값)를 되돌려 준다.
- 반복하여 사용하는 내용에 대해 사용하면 중복 코드를 막을 수 있다.
- 모듈화로 인해 코드의 가독성을 높인다.

## 메소드의 정의

- ❖ 접근 제어자 : 해당 메서드에 접근할 수 있는 범위를 명시.
- ❖ 리턴타입 : 메서드가 모든 작업을 마치고 반환하는 데이터의 타입을 명시.
- ❖ 메서드 명 : 메서드를 호출하기 위한 이름.
- ❖ 매개변수목록 : 메서드 호출 시에 전달되는 인수의 값을 저장할 변수들을 명시.

```
접근제어자 [지정예약어] 리턴타입 메소드명(매개변수목록)[throws 예외클래스들]
{
    실행문;
    실행문;
}

public int add(int x, int y){
    int result = x + y;
    return result;
}
```

# 매개변수 전달 기법

## ❖ Call by Value

- 기본형 매개변수이며 값을 전달한다.
- `int a = 10;`
- `int b = 20;`
- `int num = sum(a, b);`

## ❖ Call by Reference

- 참조형 매개변수를 전달하며 데이터의 참조값(주소)를 전달한다.
- `StringBuffer sb = new StringBuffer("메시지");`
- `printMessage(sb);`

# 리턴값

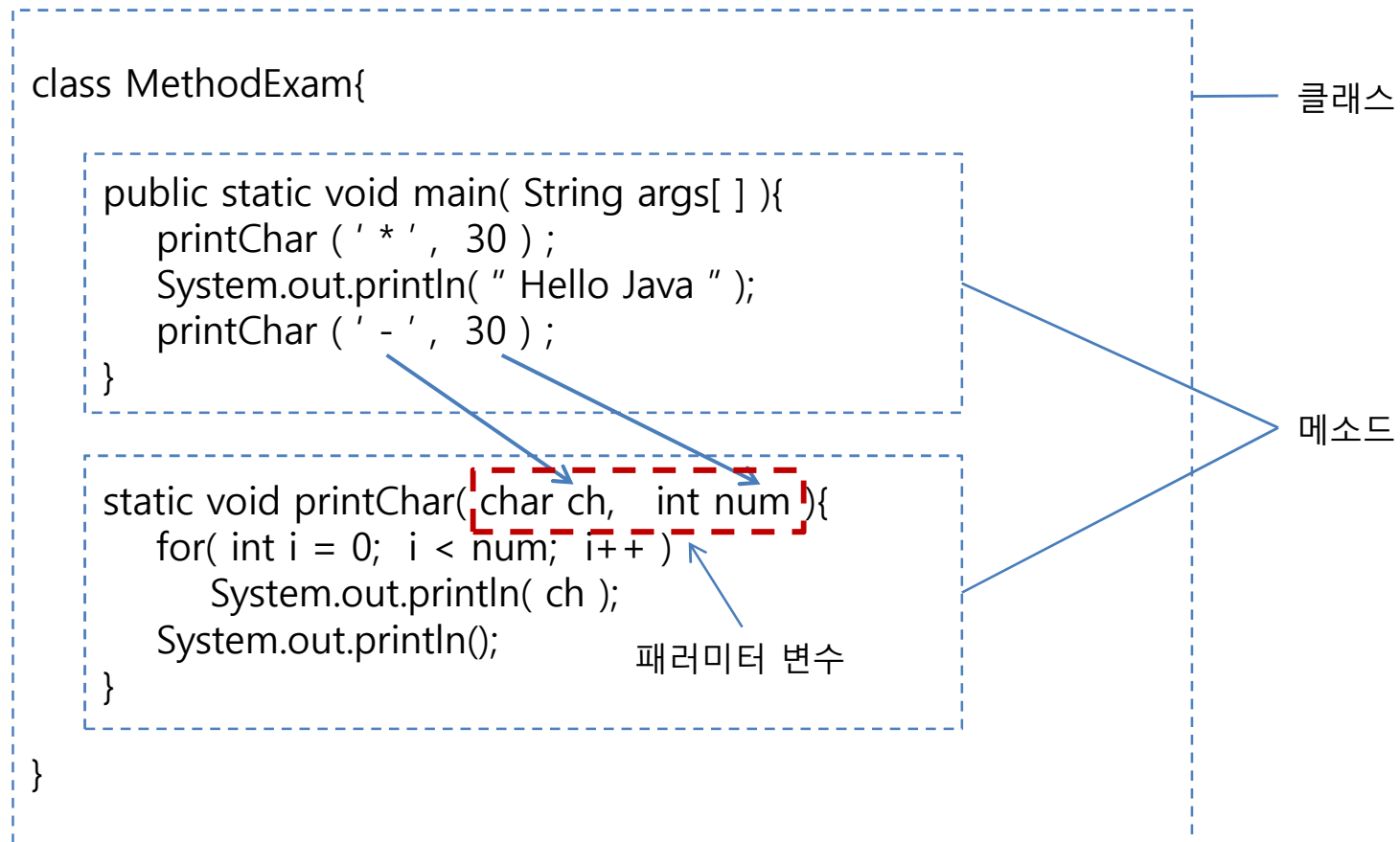
## ❖ 리턴값

- 메소드를 실행한 후 결과로 되돌려 주는 값
- return 예약어를 이용.

## ❖ 종류

- void : 실행 후 돌려 줄 결과가 없을 때.
- 기본 자료형 : 8대 기본자료형
- 레퍼런스 자료형 : 모든 클래스나 인터페이스와 같은 레퍼런스 타입의 자료형

# 메소드 호출과 프로그램의 실행 흐름



## 메소드 호출과 프로그램의 실행 흐름

