

5. 객체지향 프로그램

(1) 객체란?

동일한 성질의 데이터와 메소드를 한 곳에 모아두고 필요한 곳에서 언제든지 이용할 수 있게 만들어놓은 덩어리로, 이를 이용한 프로그래밍 방식이 객체지향 프로그래밍이다.

(2) 메소드 : 작업을 수행하기 위한 명령문의 집합

<메소드의 기본 구성>

```
접근제한자    [static]    return type    메소드명([매개변수1, 매개변수2..]){
public, private..    (return이 없을 경우 void)
처리할 프로세스들
return return값; (return type이 void일 경우 입력하지 않는다.)
}
```

6. 클래스의 기초 코딩방법

(1) 클래스의 기본 순

: 패키지명 -> 클래스명 -> **데이터** -> 생성자 함수 -> **메소드** -> getter & setter

<ex> - '정사각형의 넓이 구하기'

```
package com.lec.Square // 패키지명
public class Square { // 클래스명
    private int side; // 데이터
    public Square() { //매개변수 없는 생성자 => 생성자 오버로딩
        System.out.println("매개변수 없는 생성자 호출됨");
    }
    public Square(int side) { //매개변수 있는 생성자
        this.side = side;
        System.out.println("매개변수 있는 생성자 호출됨");
    }
    public int area() { // 메소드
        return side * side;
    }
    public int getSide() { // getter & setter
        return side;
    }
    public void setSide(int side) {
        this.side = side;
    }
}
```