

컴퓨터 프로그래밍1

실습 10주차

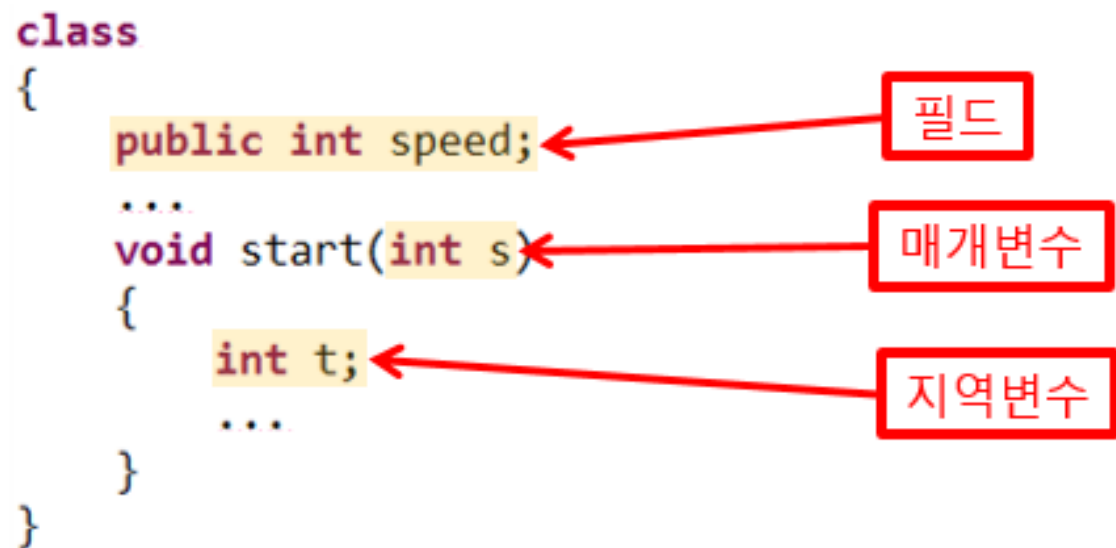
- 클래스와 메소드 사용법 숙달 -

필드와 메소드 (1/4)

필드

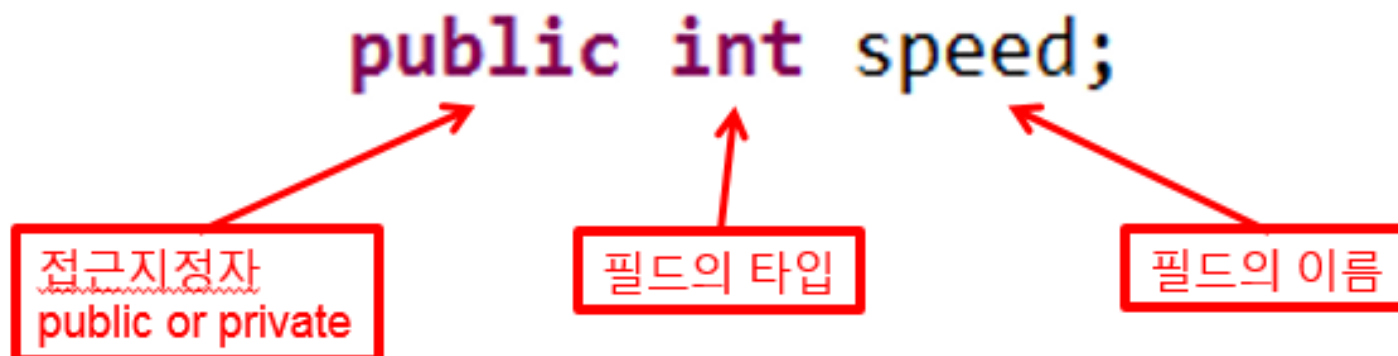
■ 변수의 종류

```
class
{
    public int speed;
    ...
    void start(int s)
    {
        int t;
        ...
    }
}
```



■ 필드 선언 형식

public int speed;



필드와 메소드 (2/4)

접근자와 설정자

- 객체 지향 방법의 핵심은 구현의 세부 사항을 감추는 것
- 필드에 직접 접근하기 위한 접근자와 설정자를 사용

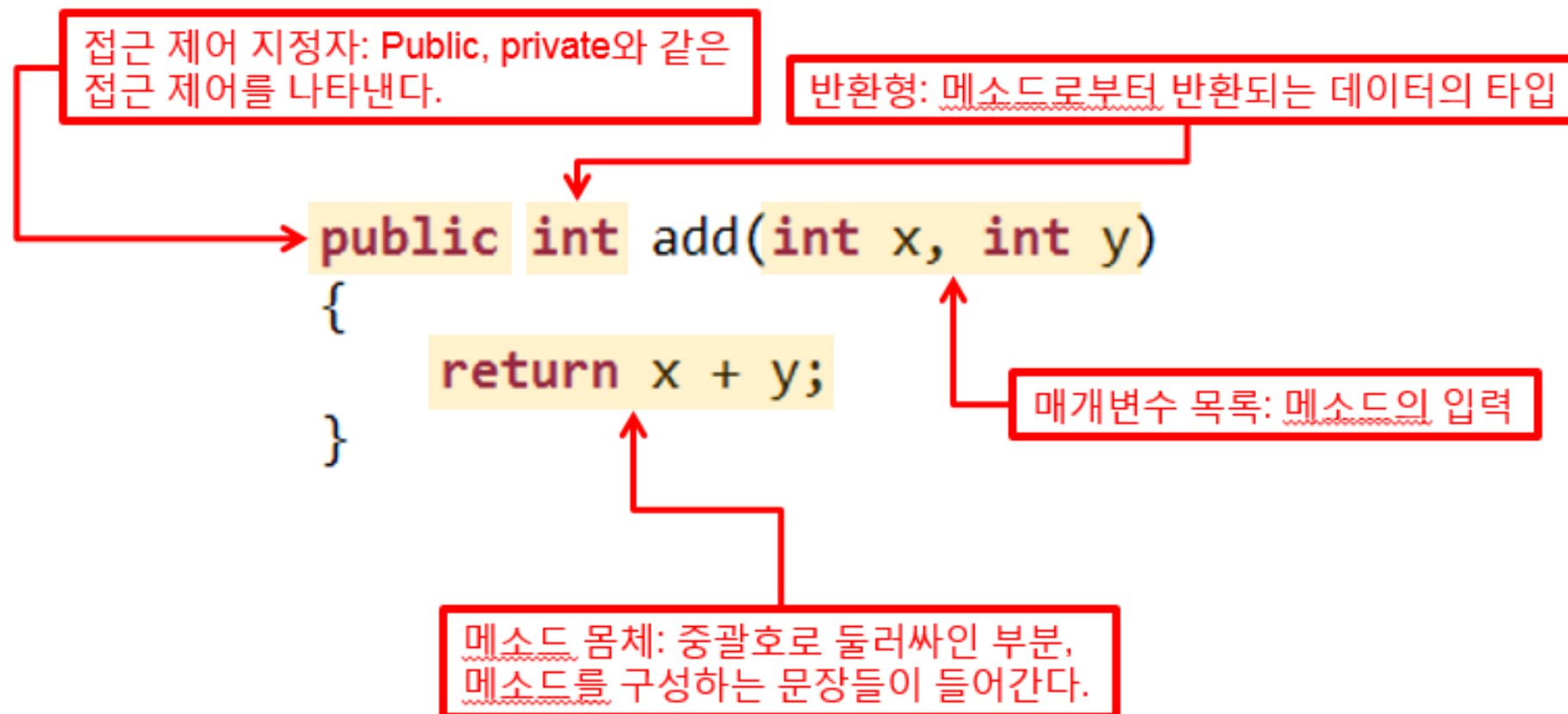
```
class Car {  
    private String color; // 색상  
    private int speed; // 속도  
    private int gear; // 기어  
  
    public String getColor() {  
        return color;  
    }  
    public void setColor(String c) {  
        color = c;  
    }  
    public int getSpeed() {  
        return speed;  
    }  
    public void setSpeed(int s) {  
        speed = s;  
    }  
    public int getGear() {  
        return gear;  
    }  
    public void setGear(int g) {  
        gear = g;  
    }  
}
```

필드가 모두 private로 선언되었기 때문에
클래스 내부에서만 사용이 가능하다.

필드와 메소드 (3/4)

메소드 형식

- 메소드의 선언은 다음과 같은 형식을 가진다



필드와 메소드 (4/4)

메소드의 반환값

- 메소드는 하나의 반환값을 가질 수 있으며, 메소드 호출로 값을 받는 경우 반환값의 타입과 일치 해야 한다.

```
Car myCar = new Car();  
int value = myCar.getSpeed();
```

```
class Car {  
    private int speed = 0; // 속도  
  
    public int getSpeed(){ return speed; }  
    public void setSpeed(int s){ speed = s; }  
}
```

실습 (1/2)

■ 사각형을 나타내는 클래스 Rectangle 만들기 (172p. 1번 문제)

- 필드 : W(넓이), H(높이)
- 메소드 : 접근자, 설정자, 넓이를 반환하는 메소드 area(), 둘레를 반환하는 메소드 perimeter()
- 사각형 클래스를 테스트 할 Rectangle 클래스 (main 함수) 생성

실습 (2/2)

문자열의 자음과 모음 개수를 출력하는 메소드 작성 (172p. 6번 문제)

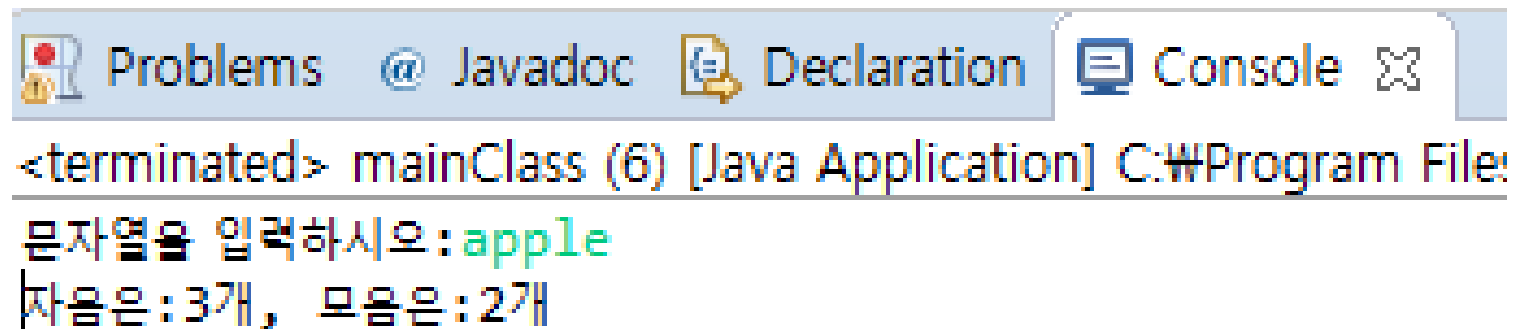
- 사용자에게 문자열을 입력 받고, 입력 받은 문자열에서 자음과 모음의 개수를 출력하는 메소드 작성 (반복문, 조건문 이용)
- 문자열의 n 번째 글자를 가져오는 메소드 : `charAt(int n)`

```
String text = "abcde";  
System.out.println("text의 두번째 글자 : "+ text.charAt(1));
```

text의 두번째 글자 : b

- 만든 메소드를 main 함수에서 호출하여 동작 확인

출력 결과



```
<terminated> mainClass (6) [Java Application] C:\#Program File:  
문자열을 입력하시오:apple  
자음은:3개, 모음은:2개
```

과제 (1/2)

과제 I. 문자열을 역순으로 출력하는 프로그램 작성

```
public static void main(String[] args) {
    Scanner Input = new Scanner(System.in);
    String str;

    System.out.print("문자열을 입력하시오: ");
    str = Input.nextLine();

    System.out.println(reverse(str));
}
```

```
public static String reverse(String s) {  
  
}  
}
```

오픈 라이브러리를 사용해도 좋지만
본인이 직접 reverse 메소드를 만들면 가점 부여

과제 (2/2)

과제2. 계정 클래스 생성

- 필요 메소드1

: 사용자로부터 아이디와 비밀번호를 입력 받아 계정을 생성하는 메소드

- 필요 메소드2

: 사용자가 입력한 아이디, 비밀번호를 설정된 아이디, 비밀번호와 비교하여 일치하는지 검사하는 메소드

- main 함수에서 계정 생성 -> 로그인 순으로 동작하는 메인 클래스를 생성하여 구현한 계정 클래스를 테스트

출력 결과

<계정생성>

아이디를 입력하시오: test123

비밀번호를 입력하시오: 1234

<로그인>

ID: test123

PW: 12345

로그인에 실패하였습니다.

Q&A



수고하셨습니다