

## 제 06 장 클래스(실습)

---

### □ 개념 확인 학습

---

1. 클래스란 무엇인가요?
2. 객체란 무엇인가요?
3. 인스턴스란 무엇인가요?
4. 클래스 멤버에는 어떤 것들이 있나요?
5. 필드란 무엇인가요?
6. 생성자란 무엇인가요?
7. 생성자 오버로딩이란 무엇인가요?
8. 메소드란 무엇인가요?
9. 메소드 오버로딩이란 무엇인가요?
10. `this`은 언제 사용하나요?
11. `this()`는 언제 사용하나요?
12. `void`는 언제 사용하나요?
13. `static`은 언제 사용하나요?

14. 싱글톤이란 무엇인가요?
15. final이란 무엇인가요?
16. 상수는 무엇인가요?
17. 상수는 어떻게 만드나요?
18. 접근 제한자가 명시적으로 쓰여 있지 않은 경우 접근 범위는 어떻게 되나요?

---

## □ 적용 확인 학습 & 응용 프로그래밍

---

1. 다음 프로그램은 정상적인가? 가비지 컬렉터 의해 사라지게 되는 객체가 존재하는가?

```
class MyClass { }

class Test
{
    MyClass doSomething()
    {
        MyClass b = new MyClass(); //6번째 줄
        return b;
    }
    public static void main (String args[])
    {
        Test t = new Test();
        MyClass newObj = t.doSomething();
        newObj = new MyClass(); //13번째 줄
    }
}
```

2. 다음 프로그램에서 잘못된 부분은 무엇인가? 올바르게 수정하시오.

```
class Rectangle
{
    int width, height;
    int area() { return width*height; }
```

```

}
public class Test {
    public static void main(String[] args)
    {
        Rectangle myRect;
        myRect.width = 10;
        myRect.height = 20;
        System.out.println("면적은 " + myRect.area());
    }
}

```

3. 다음과 같이 클래스가 정의되어 있다고 가정하자. 이 클래스의 객체를 생성하고 필드를 10과 1.2345로 초기화하며 각 필드의 값을 출력하는 코드를 작성하라.

```

public class NumberBox {
    public int ivalue;
    public float fvalue;
}

```

4. 사각형을 나타내는 클래스 `Rectangle`을 만들어보자. 사각형은 가로(w)와 세로(h)를 가지며, 사각형의 넓이를 반환하는 `area()`, 사각형의 둘레를 반환하는 `perimeter()` 메소드를 가진다. `Rectangle` 클래스를 작성하고 객체를 생성하여 테스트하라.

5. 학생을 나타내는 클래스 `Student`를 만들어보자. 학생은 이름(private name)과 학번(private rollno)을 가진다. 이름학번(홍길동\_202020)를 출력하는 메소드 `toString()`를 가진다. 이름을 전달하는 생성자도 필요하다. 학번을 전달해 세팅하는 메소드도 필요하다. `Test` 클래스를 작성하고 이름 없는 객체 생성(학번은 객체 생성 후 세팅), 이름을 전달한 객체를 생성하여 메소드 `toString()` 호출의 결과를 출력하라.

6. 날짜를 나타내는 클래스 `Date`를 만들어보자. `Date`는 연도, 월, 일 등의 속성을 가지며, 날짜를 “2012.7.12”와 같이 출력하는 메소드 `print1()`, 날짜를 “July 12, 2012”와 같이 출력하는 `print2()` 등의 메소드를 가진다. `Date` 클래스를 작성하고 객체를 생성하여서 테스트하라.

7. 영화 `Movie` 클래스를 정의하여 보자. `Movie` 클래스는 영화 제목, 평점, 감독, 발표된 연도 등의 필드를 가진다. 영화의 모든 정보를 화면에 출력하는 `print()`라는 메소드를 구현하라. `Movie` 클래스를 작성하고 객체를 생성하여서 테스트하라.

8. 강아지를 나타내는 클래스 `Dog`을 만들어보자.

- 강아지 클래스는 나이(age), 색깔(color), 간식(food), Dog객체 수(total)의 필드를 가지며,
- 몇 개의 메소드를 가진다.
  - : barking() //barking()이 호출될 때 마다 간식은 1개씩 감소하고, '멍멍' 리턴.  
//간식의 개수가 0이면 더 이상 감소하지 않고, '시무룩' 리턴
  - : hungry() //간식이 1개 이하이면 true를 리턴.
  - : sleeping() //매개변수로 간식을 주면 false, 간식을 주지 않으면 true 리턴  
//전달된 간식의 수는 food에 누적 저장된다.
  - : toString() //현재 Dog 객체 상태를 문자열로 리턴
- Dog 객체가 생성될 때 마다 현재 생성된 Dog 객체의 수를 저장해야 한다.
  - : getTotal() //main()에서 Dog 객체가 몇 개 생성되었는지 출력

9. Dog 클래스를 작성하고 객체를 생성하여 테스트하라.

10. Dog 객체 배열을 만들어 다섯 개를 저장 한 후 사용자가 묻는 색깔의 강아지가 가지고 있는 간식 수를 출력하는 기능을 DogTest에 추가 하시오.

11. 다음의 내용을 클래스(CellPhone)로 작성하시오.

- 핸드폰에는 폰 번호와 현재 전원의 상태를 저장하고 있다.
- 폰이 처음 생성될 때는 폰 번호와 현재의 전원 상태를 on상태(true)로 한다.
- CellPhone 객체가 생성될 때 마다 현재 생성된 CellPhone의 전체 수량을 저장(count)해야 한다.
- 기능으로는 전원 버튼을 누를 때 마다 현재의 전원상태를 토글(on(true)/off(false)) 한다.
- 현재 객체 상태를 문자열로 리턴한다.
- telNum, power, getCount(), toString(), powerToggle(), getCount()

12. 위 CellPhone 클래스를 테스트하는 클래스(CellPhoneTest)를 작성하시오.

13. CellPhone을 배열을 만들어 다섯 개를 저장 한 후 사용자가 묻는 번호의 전원상태를 알려주는 내용을 CellPhoneTest에 추가 하시오.

14. class Calculation에서 2개의 정수의 합을 구하는 sum()과 3개의 정수의 합을 구하는 sum()을 메소드 오버로딩을 이용하여 정의하라. class CalculationTest를 작성하여 sum() 함수를 테스트 하시오.