# **12차 연습문항**

## **파이썬 심화: 객체지향**

| 수강생 공지 사항  ❖ 제출은 프로그래머스를 통해 해주시기 바랍니다([링크](https://campus.programmers.co.kr/app/courses/24550/curriculum)).  ❖ 파일명은 아래와 같은 형식으로 제출해주세요  ➢ 교육생번호\_이름\_교과목\_문항\_N차시\_강의명.pdf  ex) DR-11111\_홍길동\_파이썬\_연습문항\_1차시\_환경및기본.pdf  ex) DR-11111\_홍길동\_파이썬\_추가문항\_1차시\_환경및기본.pdf  ❖ 답은 "write your answer"에 적어주세요. 다만 코딩 문제의 경우 output까지 답에 포함시켜 주세요.  ex)  ❖ 답은 캡쳐를 하셔도 되고 텍스트로 넣으셔도 됩니다.  ❖ 마감 기한은 문제가 나간 주 **일요일 23:59까지**입니다. |
| --- |

### 

### **1.** **다음에서 객체지향의 특징이라고 볼 수 없는 것을 고르시오.**

(1) 추상화 (2) 재사용성

(3) 캡슐화 (4) 상속

(5) 다형성 (6) 분산화

| 6번 |
| --- |

### **2.** **보기 중 아래 설명에 해당하는 특징을 찾아 적으시오.**

보기: 추상화, 다형성, 상속, 캡슐화

(1) 사물 또는 개념을 컴퓨터에서 처리할 수 있을 수준으로 축약하고 핵심을 추출

(2) 필요한 정보와 인터페이스만을 공개하고, 나머지 자세한 구현 내용들을 숨김

(3) 자식 클래스는 부모 클래스의 모든 속성과 함수들을 물려받음

(4) 상속 관계의 여러 클래스들에서 동일한 명칭으로 구현된 함수들이 각각 다르게 작동

| 1. 추상화 (2) 캡슐화 (3) 상속 (4) 다형성 |
| --- |

### **3.** **다음 중 파이썬 클래스에서 멤버변수 또는 멤버함수의 접근 제한자에 대한 설명 으로 틀린 것은 무엇인가?**

A) 멤버변수나 멤버함수 앞에 밑줄 한 개(\_)를 붙이면 protected 접근 제한자가 된다.

B) 멤버변수나 멤버함수 앞에 밑줄 두 개(\_\_)를 붙이면 private 접근 제한자가 된다.

C) 멤버변수나 멤버함수 앞에 밑줄이 없으면 public 접근 제한자가 된다.

D) 파이썬에서는 엄격한 접근 제한자를 지원하지 않으며, 밑줄 표기는 관용적인 규칙 일뿐이다.

| A |
| --- |

### **4.** **파이썬에서는 다른 프로그래밍 언어와 달리 엄격한 접근 제한자(private, protected, public)를 지원하지 않는다. 대신, 관례적으로 변수나 함수의 접근 수준을 나타내기 위해 밑줄을 사용한다. 이를 이용해 다음 보기에서 요구하는 멤버함수의 코드를 작성하라.**

(1) foo() 함수의 private 선언

(2) foo() 함수의 protected 선언

(3) foo() 함수의 public 선언

| 1. def \_\_foo(self) 2. def \_foo(self) 3. def foo(self) |
| --- |

### **5.** **다음 중 객체가 생성될 때에 대한 설명으로 잘못된 을 고르시오.**

A) 객체가 생성될 때는 클래스의 생성자(\_\_init\_\_ 메서드)가 호출된다.

B) 객체가 생성될 때는 객체의 메서드가 메모리에 로드된다.

C) 객체가 생성될 때는 클래스의 모든 멤버변수가 초기화된다.

D) 객체가 생성될 때는 파이썬 내부적으로 객체의 메모리 공간을 할당한다.

| B |
| --- |

### **6.** **다음 중 객체의 사용에 대한 설명으로 틀린 것을 고르시오.**

A) 객체는 데이터와 해당 데이터를 처리하는 메서드를 함께 묶어서 관리할 수 있다.

B) 객체는 다른 객체로부터 상속을 받아 기능을 확장할 수 있다.

C) 객체는 생성자 없이도 멤버함수를 가질 수 있다.

D) 리스트는 객체를 요소로 포함할 수 없다.

| D |
| --- |

### **7.** **생성자에서 사각형의 좌측 상단 좌표 x,y와 너비와 높이 w, h를 초기화 하는 사각형(Rectangle) 클래스의 생성자를 구현하시오.**

| Example Code |
| --- |
| class Rectangle:  def \_\_init\_\_(self, x, y, width, height):  # Write your code here |

|  |
| --- |

### **8.** **자식 클래스에서 부모클래스의 함수를 호출하는 코드를 작성하는 방법은 무엇인가요?**

| super().func()을 사용해서 부모의 함수를 호출해서 오버라이딩이 가능하다.  오버라이딩을 하지 않을 거면 super().func()을 사용하지 않고 test.foo()를 실행해도 정상적으로 실행되는 것을 확인할 수 있다. |
| --- |