**객체지향 part 모의고사**

1번문제 : 객체지향 프로그래밍 언어에 대한 설명으로 거리가 먼 것은?

① 프로시저(일단 함수라고 생각하시길)에 근간을 두고 프로그램을 구현하는 기법이다.

② 현실 세계를 모형화하여 유저와 개발자가 쉽게 이해할 수 있다

③ 소프트웨어의 재사용률이 높아진다.

④ 소프트웨어의 유지보수성이 높아진다.

2번문제 : 객체지향 원리에 대한 설명으로 옳지 않은 것은?

① 인스턴스는 한 클래스의 모든 속성을 상속받고, 거기에 자신의 속성을 추가하여 만들어진 서브 클래스이다.

② 캡슐화는 데이터와 그것의 처리에 관한 작업이 하나로 모아져 있는 것이므로 객체지향원리 만의 특징이라고 할 수 없다.

③ 상속은 객체지향 원리의 특성이나, 다중상속이 적용된 경우에는 개발 및 유지보수가 힘들 수 있다.

④ 다형성은 유지보수에 부정적인 영향을 줄 수 있다.

3번문제 : 다음 설명에 해당하는 것은?

* 다른 객체에게 자신의 정보를 숨기고 내부에서 실행되는 연산만을 통해 접근한다.
* 유지보수와 소프트웨어 개발 시 오류를 최소화할 수 있다.

① 추상화

② 정보 은닉

③ 상속

④ 다형성

4번문제 :

python에서 class B(A): pass 와 같은 상속관계일 때 옳지 않은 설명은?

① A를 상위 클래스라고 한다

② B를 하위 클래스라고 한다.

③ A가 B에 상속될 때 삭제되는 A의 메소드는 없다.

④ B와 A의 위치를 위의 코드에서 바꾸어도 문제가 없다.

5번문제

* **클래스 1**
* **클래스 1을 부모로 가지는 클래스 2**
* **클래스 2를 부모로 가지는 클래스 3**

만약 클래스 3의 인스턴스를 생성하고, 지우면 constructor와 destructor의 호출 순서는?

① 둘 다 부모 클래스 먼저

② 생성자는 부모먼저, 소멸자는 자식 먼저

③ 생성자는 자식 먼저, 소멸자는 부모 먼저

④ 둘 다 자식 먼저

6번문제 : 연관된 데이터와 함수를 함께 묶어 외부와 경계를 만들고 필요한 인터페이스만 밖으로 드러내는 과정은?

① 메시지

② 캡슐화

③ 상속

④ 다형성

7번문제 : 객체지향 개념 중 하나 이상의 유사한 객체들을 묶어 공통된 특징을 표현된 데이터 추상화를 의미하는 것은?

① 메시지

② 캡슐화

③ 상속

④ 다형성

8번 문제 : 객체지향 프로그래밍 언어에서 캡슐화에 대한 설명으로 옳은 것을 모두 고르시오

ㄱ : 캡슐화를 하면 객체 간의 결합도가 높아진다.

ㄴ : 캡슐화된 객체들은 재사용이 용이하다.

ㄷ : 프로그램 변경에 의한 오류의 파급효과가 적다.

9번 문제 : 객체의 성질을 분해하고 공통된 추출하여 클래스를 선정하는 것이다. 즉 불필요한 부분을 생략하고 객체의 속성 중 가장 중요한 것에만 중점을 두어 모델화한다.

① 상속

② 추상화

③ 다형성

④ 캡슐화

10번 문제 : 빈칸에 공통적으로 들어갈 말은?

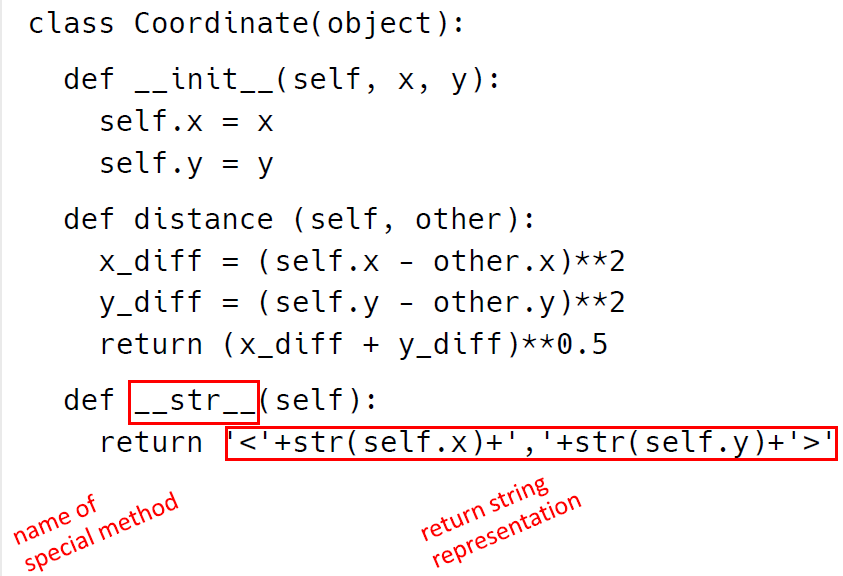
( )은

11번 문제 : 다음 문장에서 설명하는 개념은 무엇인가?

원의 면적을 구하는 getArea() 함수를 가지고 있는 객체, 직사갹형의 면적을 구하는 getArea() 함수를 가지고 있는 객체, 삼각형의 면적을 구하는 getArea() 함수를 가지고 있는 객체는 getArea() 라는 함수를 공통적으로 가지고 있다. 여기서 getArea() 함수는 실제로 면적을 계산할 때는 객체의 모양에 따라 계산방법이 각각 다르다.

12번문제 :

1. Operrator overloading (연산자 오버로딩)에 대해서 서술하시오.
2. 다음의 클래스에 대한 add연산을 정의하시오

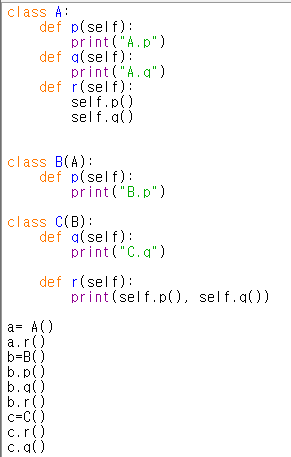
a=coordinate(4,3)

b= coordinate(7,5)

c=a+b

(여기서 c는 coordindate의 인스턴스이다.)

13번 문제 : 다음의 python code가 주어졌다고 하자(overriding)



다음의 출력결과는?

14번 문제

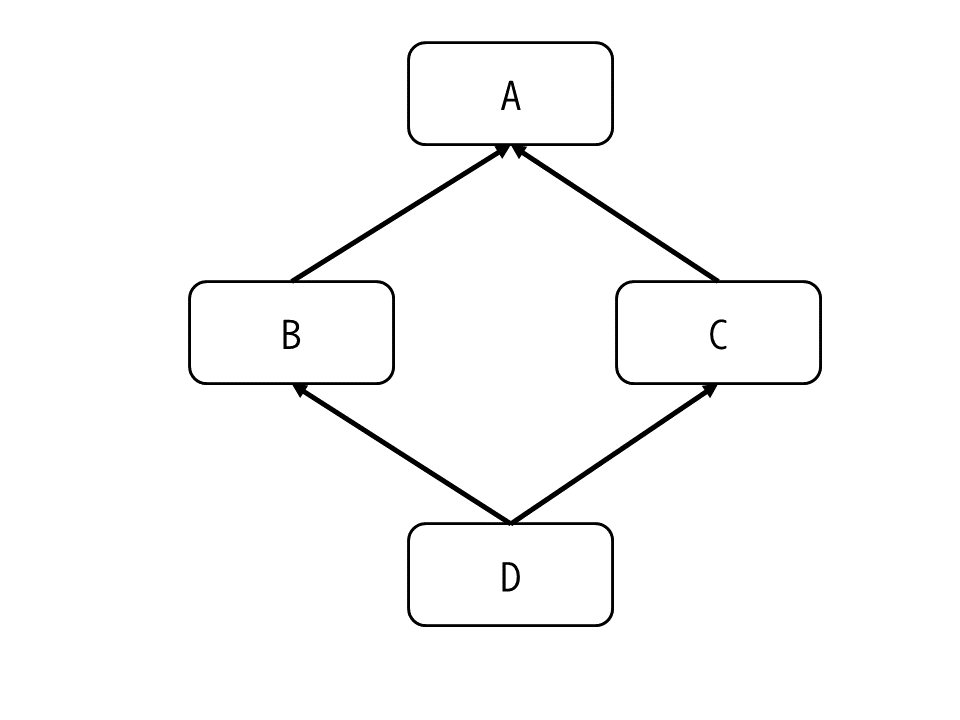


이 코드는 에러가 발생한다.

1. 이와 같은 원리로 추상클래스의 객체는 생성할 수 없는데 이 이유는?
2. 또한 이 말은 추상 클래스 타입의 변수를 선언하지 못한다는 것인가? 설명해보세요
3. 추상 클래스는 상속할 수 있는가?

15번 문제

클래스 A,B,C,D가 있다고 가정하자. 이 클래스들의 상속관계는 다음과 같다.

여기서 발생할 수 있는 문제점에 대해서 서술하시오.

16번 문제

Person 클래스를 상속받아서 Student 클래스를 만드는 방법으로 올바른 것을 고르세요.

1. class Student:Person
2. class Student(Person)
3. class Student(Person):
4. class Person(Student):
5. class Student=Person:

17번 문제

다음 중 상위 클래스의 생성자를 호출하는 방법으로 올바른 것을 고르세요.

1. \_\_init\_\_()
2. super.\_\_init\_\_()
3. super()
4. super().\_\_init\_\_()
5. super().\_\_init\_\_

18번문제

3. 다음 중 메서드 오버라이딩에 대한 설명으로 잘못된 것을 모두 고르세요.

1. 메서드 오버라이딩은 함수 이름을 다르게 만들어도 된다.
2. 메서드 오버라이딩은 기반 클래스의 메서드는 무시하고 새로운 메서드를 만든다는 뜻이다.
3. 메서드 오버라이딩은 어떤 기능이 같은 이름으로 사용되어야 할 때 활용한다.
4. 오버라이딩 된 메서드에서는 기반 클래스의 메서드를 호출할 수 없다.
5. 메서드 오버라이딩은 원래 기능을 유지하면서 새로운 기능을 덧붙일 때 사용한다.

19번문제

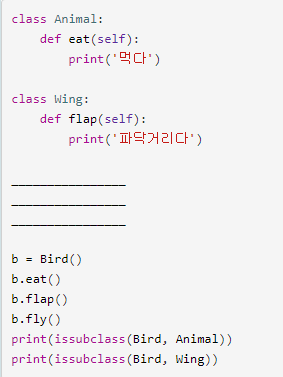
4. 다음 중 추상 클래스를 만드는 방법으로 올바른 것을 모두 고르세요.

1. class PersonBase(metaclass=ABCMeta):
2. from abc import \*  
   class PersonBase(metaclass=ABCMeta):
3. from abc import \*  
   class PersonBase(ABCMeta):
4. import abc  
   class PersonBase(abc.ABCMeta):
5. import abc  
   class PersonBase(metaclass=abc.ABCMeta):

20번 문제

(a)

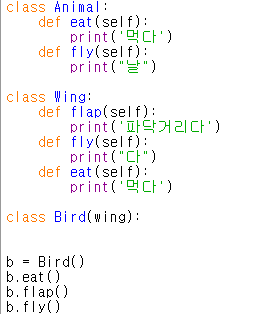
다음 소스 코드에서 동물 클래스 Animal과 날개 클래스 Wing을 상속받아 새 클래스 Bird를 작성하여 '먹다', '파닥거리다', '날다', True, True가 각 줄에 출력되게 만드세요.



출력 결과

먹다  
파닥거리다  
날다  
True  
True

b) 다음은 앞서 (a)에서 본 클래스를 변형하였다.



여기에서 bird 클래스는 wing의 모든 메소드와 Animal의 fly 메소드만을 상속받는다.

선택적으로 상속받는 것은 가능한가? 가능하다면 구현해보아라.

21번 문제 : 다음은 calc라는 클래스에 대한 설명이다.

* 생성할 때 두 정수를 parameter로 받는다.
* gcd라는 메소드는 인스턴스에 저장되어있는 값의 최대공약수를 찾는다.
* Sum이라는 메소드는 두 정수를 더한다.
* Sub라는 메소드는 두 정수를 뺀다.
* 만약 sum이나 sub의 결과가 음수라면 오류를 발생시킨다.

다음을 만족하는 calc를 구현하시오.

22번 문제

다음 클래스의 greeting 메서드를 호출하기 위한 방법으로 올바른 것을 고르세요.

class Person:

    def greeting(self):

        print('Hello')

1. Person.greeting()
2. greeting()
3. maria = Person  
   maria.greeting()
4. maria = Person()  
   maria.greeting()
5. Person(greeting)

23번 문제

다음과 같이 Person 클래스가 있습니다. 클래스에서 다른 메서드를 만들었을 때 인스턴스 속성 name에 접근하기 위한 방법으로 올바른 것을 고르세요.

class Person:

    def \_\_init\_\_(self, name):

        self.name = name

1. name
2. self
3. Person.name
4. self[name]
5. self.name

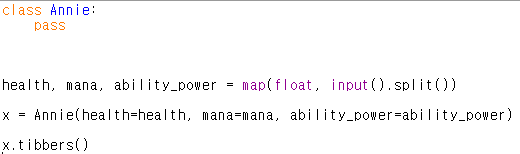
24번 문제

클래스의 메서드 def \_\_init\_\_(self):에서 속성을 만들려고 합니다. 다음 중 비공개 속성을 고르세요.

1. self.name
2. self.\_name
3. self.\_\_name
4. self.\_\_name\_\_
5. self.name\_\_

25번 문제

(a) 표준 입력으로 게임 캐릭터 능력치(체력, 마나, AP)가 입력됩니다. 다음 소스 코드에서 애니(Annie) 클래스를 작성하여 티버(tibbers) 스킬의 피해량이 출력되게 만드세요. 티버의 피해량은 AP \* 0.65 + 400이며 AP(Ability Power, 주문력)는 마법 능력치를 뜻합니다.



Input1 : 511.68 334.0 298

Output 1 : 티버: 피해량 593.7

Input2 : 1803.68 1184.0 645

Output2 : 티버: 피해량 819.25

26번 문제

다음 중 클래스 바깥에서 클래스 속성 x에 접근하는 방법으로 올바른 것을 고르세요.

class Person:

    x = {}

1. Person.x
2. Person(x)
3. x
4. self.x
5. Person['x']

27번 문제

2. 다음 중 정적 메서드로 올바른 것을 고르세요.

1. def print\_count(self):  
       print(self.count)
2. @staticmethod  
   def sub(self, a, b):  
       print(a - b)
3. @staticmethod  
   def div(a, b):  
       print(a / b)
4. @staticmethod  
   def add(cls, a, b)  
       print(a + b)
5. def print\_count(cls):  
       print(cls.count)

28번 문제

3. 다음 중 클래스 메서드에 대한 설명으로 잘못된 것을 고르세요.

1. 클래스 메서드는 **클래스.메서드()** 형식으로 호출한다.
2. 클래스 메서드는 위에 @classmethod를 붙여서 만든다.
3. 클래스 메서드의 첫 번째 매개변수는 self이며 현재 인스턴스가 들어온다.
4. 클래스 메서드는 인스턴스 없이 호출할 수 있다.
5. 클래스 메서드는 위에 @staticmethod를 붙여서 만든다.

29번 문제

표준 입력으로 시:분:초 형식의 시간이 입력됩니다. 다음 소스 코드에서 Time 클래스를 완성하여 시, 분, 초가 출력되게 만드세요. from\_string은 문자열로 인스턴스를 만드는 메서드이며 is\_time\_valid는 문자열이 올바른 시간인지 검사하는 메서드입니다. 시간은 24시까지, 분은 59분까지, 초는 60초까지 있어야 합니다. 정답에 코드를 작성할 때는 class Time:에 맞춰서 들여쓰기를 해주세요.

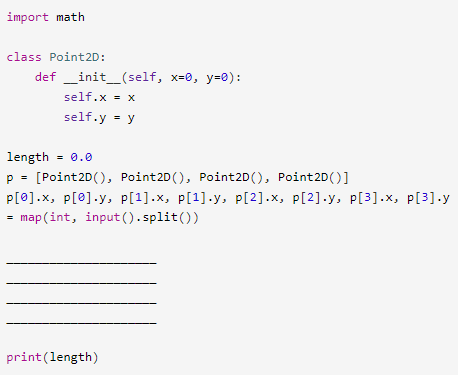


Input1 : 23:35:59 -> 23 35 59

Input 2 : 12:62:43 -> 잘못된 시간 형식입니다.

30번 문제

표준 입력으로 x, y 좌표 4개가 입력되어 Point2D 클래스의 인스턴스 리스트에 저장됩니다. 여기서 점 4개는 첫 번째 점부터 마지막 점까지 순서대로 이어져 있습니다. 다음 소스 코드를 완성하여 첫 번째 점부터 마지막 점까지 연결된 선의 길이가 출력되게 만드세요.



Input 1: 10 10 20 20 30 30 40 40

Output 1: 42.42640687119285

Input 2: 100 100 200 200 300 300 400 400

Output 1: 424.26406871192853