**12차 연습문항**

## **파이썬 추가문제: 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.doc  ex) DR-11111\_홍길동\_파이썬\_연습문항\_1차시\_환경및기본.doc  ex) DR-11111\_홍길동\_파이썬\_추가문항\_1차시\_환경및기본.doc  ❖ 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.** **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 |

|  |
| --- |

### **2.** **아래 코드를 실행하여 출력 결과를 확인하고, 그 결과가 나온 이유를 서술하시오(결과가 나온 이유만 작성하셔도 됩니다).**

| Example Code |
| --- |
| class Vehicle:  def \_\_init\_\_(self, brand, model):  self.brand = brand  self.model = model  print(f"Vehicle initialized: {self.brand} {self.model}")    class Car(Vehicle):  def \_\_init\_\_(self, brand, model, horsepower):  super().\_\_init\_\_(brand, model) # 부모 클래스의 생성자 호출  self.horsepower = horsepower  print(f"Car initialized: {self.brand} {self.model} with {self.horsepower} HP")    # 객체 생성  my\_car = Car("AAA", "BBB", 350) |

| 부모 클래스 (Vehicle): 이 클래스는 모든 차량의 기본 정보인 브랜드와 모델을 초기화합니다. 생성자 메서드 \_\_init\_\_에서 이 두 속성을 설정합니다.    자식 클래스 (Car): 이 클래스는 Vehicle을 상속받아 더 구체적인 차량 유형인 'Car'를 표현합니다. 추가로 horsepower 속성이 포함됩니다. 자식 클래스의 생성자에서 super().\_\_init\_\_(brand, model)을 호출함으로써, 부모 클래스의 생성자를 실행하고 브랜드와 모델을 초기화합니다. 이렇게 하면 Car 인스턴스를 생성할 때 Vehicle의 모든 초기화 로직을 재사용할 수 있습니다.  그리고 부모 클래스의 생성자 안에 있는 출력 구문과 자식 클래스 생성자의 출력 구문 두개 다 객체를 생성하게 되며 실행되게 된다. |
| --- |

### **3.** **입력 데이터를 기반으로 사각형 또는 원의 면적을 계산하는 프로그램을 작성하십시오. 아래 요구사항을 참고하시오.**

**요구사항:**

* 사각형과 원을 별도의 클래스로 구현합니다. 사각형은 "1"에 대응되고 원은 "0"에 대응됩니다.
* 사각형 클래스는 왼쪽 상단 (x1, y1)과 오른쪽 하단(x2, y2) 좌표를 입력받습니다.
* 원 클래스는 중심 좌표(x, y)와 반지름을 입력받습니다.
* 좌표 값은 정수형으로 입력받습니다.
* 사용자는 모양의 종류를 문자열로 입력하며, 이후에 필요한 좌표 값이나 반지름을 입력합니다.
* 총 세 번의 모양 데이터를 입력받아 각 모양의 면적을 계산하고 출력합니다.

| Example Code |
| --- |
| import math    class Rectangle:  def \_\_init\_\_(self, x1, y1, x2, y2):  # Write your code here    def calcArea(self):  # Write your code here    class Circle:  def \_\_init\_\_(self, x, y, r):  # Write your code here    def calcArea(self):  # Write your code here      shapeList = []    shapeList = []    for i in range(3):  s = input("도형 모양을 입력하세요 (1:사각형 0:원): ")  if s == "1":  x1 = int(input("왼쪽 상단의 x좌표를 입력: "))  y1 = int(input("왼쪽 상단의 y좌표를 입력: "))  x2 = int(input("오른쪽 하단의 x좌표를 입력: "))  y2 = int(input("오른쪽 하단의 y좌표를 입력: "))  shapeList.append(Rectangle(x1, y1, x2, y2))  elif s == "0":  x = int(input("원의 중심 x 좌표를 입력: "))  y = int(input("원의 중심 y 좌표를 입력: "))  r = int(input("원의 반지름을 입력: "))  shapeList.append(Circle(x, y, r))  for s in shapeList:  print(f"면적: {s.calcArea()}") |

|  |
| --- |

### **4.** **문제14를 수정해서 사용자로부터 모양을 문자열로 입력 받은 후, 객체를 먼저 생성하고, 객체의 멤버 함수를 이용해서 각 모양에 맞는 정보들을 입력 받도록 한다.**

| Example Code |
| --- |
| import math  class Rectangle:  def \_\_init\_\_(self):  # Write your code here    def calcArea(self):  # Write your code here    def getCoordsInfo(self):  self.x1 = int(input("왼쪽 상단의 x 좌표값을 정수로 입력하세요: "))  self.y1 = int(input("왼쪽 상단의 y 좌표값을 정수로 입력하세요: "))  self.x2 = int(input("오른쪽 하단의 x 좌표값을 정수로 입력하세요: "))  self.y2 = int(input("오른쪽 하단의 y 좌표값을 정수로 입력하세요: "))    class Circle:  def \_\_init\_\_(self):  # Write your code here    def calcArea(self):  # Write your code here    def getCoordsInfo(self):  # Write your code here  # Rectangle 클래스의 getCoordsInfo 함수를 참고하여 작성하시요.    shapeList = []  for i in range(3):  s = input("도형 모양을 입력하세요 (1:사각형 0:원): ")  if s == "1":  rect = Rectangle()  rect.getCoordsInfo()  shapeList.append(rect)  else:  circle = Circle()  circle.getCoordsInfo()  shapeList.append(circle)  for s in shapeList:  print(f"면적: {s.calcArea()}") |

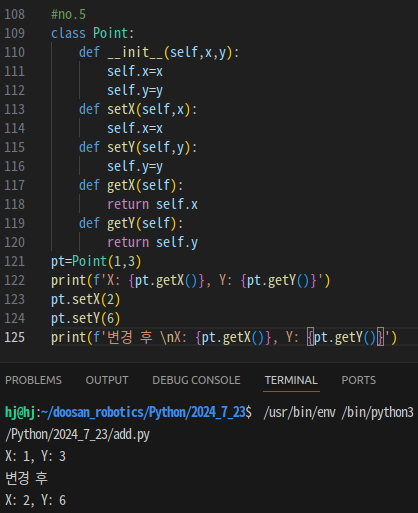
|  |
| --- |

### **5.** **x, y 좌표를 저장할 수 있는 Point 클래스를 구현해 보시오.**

(1) x, y 좌표를 인자로 전달받아 멤버 변수에 저장하는\_\_init\_\_ 함수를 구현한다.

(2) Point 클래스의 멤버 변수인 x, y를 변경할 수 있는 setX(x). setY(y) 멤버 함수를 구현한다.

(3) Point 클래스의 멤버 변수인 x, y를 각각 반환하는 getX() getY( )를 구현한다.



### **6.** **아래 자료(딕셔너리)는 연도별 고등학교의 학급당 학생 수이다(출처: 2021 교육 기본 통계 결과 발표 자료) 아래의 요구사항을 프로그램으로 작성하라.**

d = {2010:33.7, 2011:33.1, 2012:32.5, 2013:31.9, 2014:30.9, 2015:30.0, 2016:29.3, 2017:28.2, 2018:26.2, 2019:24.5, 2020:23.4, 2021:23} #년도:학급당 학생수

**요구사항**

1. 전해에 비해 가장 급격하게 학생 수가 줄어든 해는 언제인가?

2. 학급당 학생 수가 30명 미만으로 떨어진 해는 언제인가?

3. 2010년부터 2021년 사이에 평균적으로 학급당 학생 수는 어느 정도 감소했는가?

| Example Code |
| --- |
| class Problem:  def \_\_init\_\_(self, studentsNumDictionary):  self.studentsNumDictionary = studentsNumDictionary  def subproblem1(self):  max = 0  year = 0    # Write your code here    print(f"(1) {year}에 {max:.1f}명 감소한 것이 최근 가장 급격하게 줄어든 경우입니다.")  def subproblem2(self):    # Write your code here    print(f"(2) 학생 수가 30명 미만으로 떨어진 첫 해는 {i}년입니다.")    def subproblem3(self):  # 3 2010년부터 2021년까지 평균적으로 감소한 학생 수  sum = 0  # Write your code here  print(f"(3) 2010년부터 2021년까지 평균적으로 감소한 학생 수는 {sum / 11:.2f}명이다")    d = {2010:33.7, 2011:33.1, 2012:32.5, 2013:31.9, 2014:30.9, 2015:30.0, 2016:29.3, 2017:28.2, 2018:26.2, 2019:24.5, 2020:23.4, 2021:23}    p = Problem(d)  p.subproblem1()  p.subproblem2()  p.subproblem3() |

|  |
| --- |

### **7.** **클래스 Car에서 다음 조건의 매직 메소드를 만들어 아래와 같은 결과가 나오도록 프로그램을 작성하라.**

- 초기화 메소드 \_\_init\_\_

- 문자열화 메소드 \_\_str\_\_

- 동등 비교 메소드 \_\_eq\_\_: 객체의 모든 값이 동일하면 True, 아니면 False를 반환

| Example Code |
| --- |
| class Car:  def \_\_init\_\_(self, company, year, color):  # Write your code here    def \_\_str\_\_(self):  # Write your code here    def \_\_eq\_\_(self, other):  # Write your code here    # 예시 코드  mycar = Car('ABC', 2020, '검정')  yourcar = Car('DEF', 2021, '백색')  print(mycar)  print(yourcar)  print(mycar == yourcar) |
| output  자동차 회사: ABC, 년식: 2020, 색상: 검정  자동차 회사: DEF, 년식: 2021, 색상: 백색  False |

|  |
| --- |