10주차 과제

[과제번호6]\_4조\_20115122,20135151

상속예제

1.

class Person:

def \_\_init\_\_(self, name, phoneNumber):

self.name = name

self.phoneNumber = phoneNumber

print("My name is %s. My phoneNumber is %s" %(name, phoneNumber))

class Student(Person):

def \_\_init\_\_(self, name, phoneNumber, subject, studentID):

self.name = name

self.phoneNumber = phoneNumber

self.subject = subject

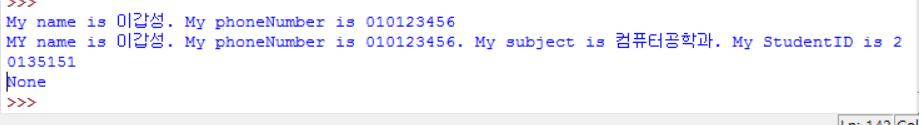
self.studentID = studentID

print("MY name is %s. My phoneNumber is %s. My subject is %s. My StudentID is %s" %(name, phoneNumber, subject, studentID))

def play():

per = Person("이갑성", "010123456")

stu = Student("이갑성", "010123456", "컴퓨터공학과", "20135151")

print(play())

2.

class Tiger:

def Jump(self):

print("호랑이처럼 멀리 점프하기")

class Lion:

def Bite(self):

print("사자처럼 한입에 꿀꺽하기")

class Liger(Tiger, Lion):

def Play(self):

print("라이거만의 사육사와 재미있게 놀기")

def play():

ti = Tiger()

ti.Jump()

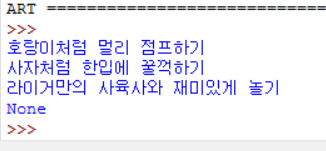
li = Lion()

li.Bite()

lig = Liger()

lig.Play()

print(play())



3.

class SuperClass:

x = 10

def PrintX(self):

print(self.x)

class SubClass(SuperClass):

y = 20

def PrintY(self):

print(self.y)

s = SubClass()

s.a = 30

print(s.PrintX())

print(s.PrintY())

print(s.a)



<명함문제>

class BusinessCard:

def set\_info(self, name, email, addr):

self.name = name

self.email = email

self.addr = addr

def print\_info(self):

print("------------------------------")

print("Name: ", self.name)

print("E-mail: ", self.email)

print("Address: ", self.addr)

print("------------------------------")

member1 = BusinessCard()

member1.set\_info("이갑성", "kabsung@naver.com", "seoul")

print(member1.name)

print(member1.email)

print(member1.addr)

member2 = BusinessCard()

member2.set\_info("홍길동", "gildong@naver.com", "jirisan")

print(member1.print\_info())

print(member2.print\_info())



과제1

class Fruit:

def \_\_init\_\_(self, fruit, color, taste, yesno): #생성자 생성

self.fruit = fruit #fruit에 받은 값을 self.fruit에 넣어줍니다.

self.color = color #color에 받은 값을 self.color에 넣어줍니다.

self.taste = taste #taste에 받은 값을 self.taste에 넣어줍니다.

self.yesno = yesno #yesno에 받은 값을 self.yesno에 넣어줍니다.

def description(self): #description이라는 메소드 생성

print("I'm a %s %s and I taste %s" %(self.color, self.fruit, self.taste))

def is\_edible(self): #is\_edible이라는 메소드 생성

if self.yesno == False: #self.yesno의 입력에 따라서 결과값 출력 if문

print("Yap! I'm edible")

else:

print("Nope! I'm not edible")

lemon = Fruit("lemon", "yellow", "sour", False) #lemon이라는 Fruit클래스의 인스턴스 객체 생성

print(lemon.description()) #lemon의 description메소드 실행

print(lemon.is\_edible()) #lemon의 is\_edible메소드 실행

