**Python程式設計作業**

**範圍： 類別、模組的應用**

**銘傳大學電腦與通訊工程系**

|  |  |
| --- | --- |
| 班 級 | 電通四乙 |
| 姓 名 | 李易軒 |
| 學 號 | 05050084 |
| 作業成果 | 應繳作業共 7 題，前3題每題20分，後4題每題10分，滿分為100分  我共完成 7 題，應得 100 分 |
| 授課教師 | 陳慶逸 |

* 請確實填寫自己寫完成題數，並且計算得分。填寫不實者(如上傳與作業明顯無關的答案，或是計算題數有誤者)，本次作業先扣50分。
* 確實填妥封面的內容，完成後請上傳pdf檔。

EX 1: 試寫出一個命名為strSplitClass的Python類別，該類別裡有一個StrSplit方法(methods)可將輸入的字串進行切割，得到以字(word)組成的串列。

例如:

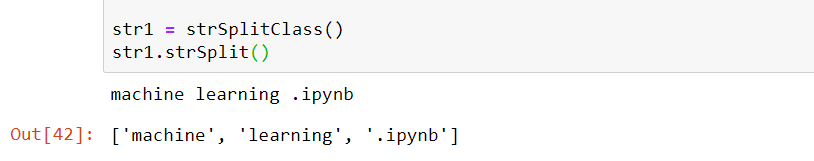
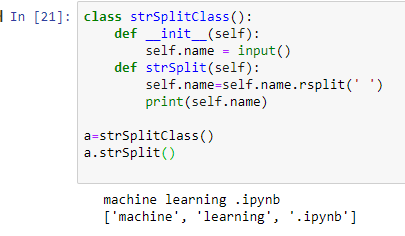


圖5-16　輸入與輸出結果

|  |
| --- |
| **class strSplitClass():**  **def \_\_init\_\_(self):**  **self.name = input()**  **def strSplit(self):**  **self.name=self.name.rsplit(' ')**  **print(self.name)**  **a=strSplitClass()**  **a.strSplit()** |



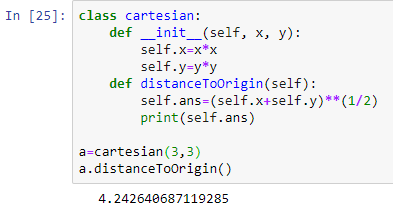
EX 2: 試寫出一個命名為cartesian的Python類別，該類別的輸入參數為 x和y兩個數值(為平面座標中的一個點(x,y))，而類別裡的方法(distanceToOrigin)可以計算出(x,y)這個點離原點的距離。

例如:



圖5-167　輸入與輸出結果

|  |
| --- |
| **class cartesian:**  **def \_\_init\_\_(self, x, y):**  **self.x=x\*x**  **self.y=y\*y**  **def distanceToOrigin(self):**  **self.ans=(self.x+self.y)\*\*(1/2)**  **print(self.ans)**    **a=cartesian(3,3)**  **a.distanceToOrigin()** |



EX 4: 試寫出一個命名為Shape的Python類別，該類別由計算矩形面積(area)、矩形周長(perimeter)和尺度大小改變(scaleSize)等三個方法(methods)所構成。

例如:

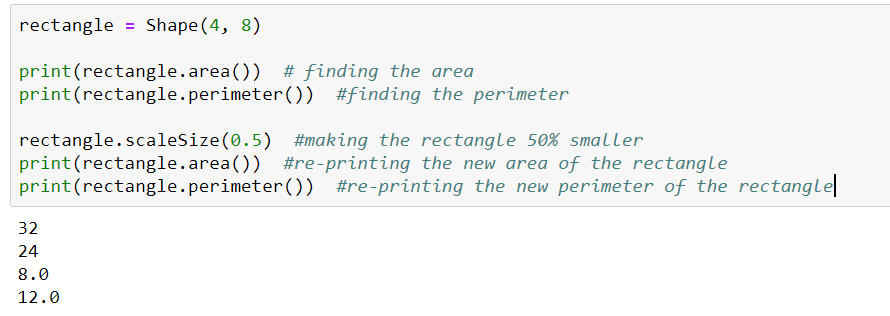
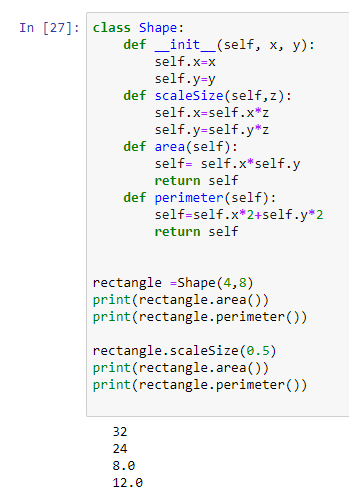


圖5-19　輸入與輸出結果

|  |
| --- |
| **class Shape:**  **def \_\_init\_\_(self, x, y):**  **self.x=x**  **self.y=y**  **def scaleSize(self,z):**  **self.x=self.x\*z**  **self.y=self.y\*z**  **def area(self):**  **self= self.x\*self.y**  **return self**  **def perimeter(self):**  **self=self.x\*2+self.y\*2**  **return self**    **rectangle =Shape(4,8)**  **print(rectangle.area())**  **print(rectangle.perimeter())**  **rectangle.scaleSize(0.5)**  **print(rectangle.area())**  **print(rectangle.perimeter())** |



EX 5: 試寫出一個命名為IOString的Python類別，該類別由兩個兩個方法(methods)所構成，其中getString負責接收使用者所輸入的字串，而另一個方法printString則是將字串改成大寫後列印出來。

例如:

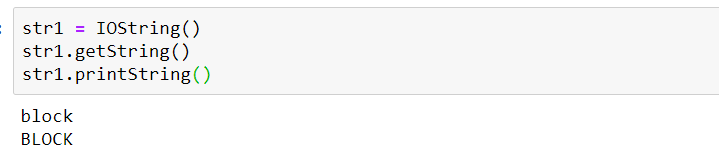
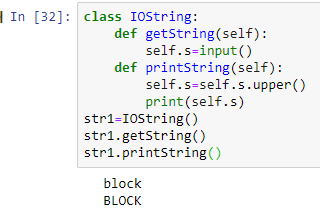


圖5-20　輸入與輸出結果

|  |
| --- |
| **class IOString:**  **def getString(self):**  **self.s=input()**  **def printString(self):**  **self.s=self.s.upper()**  **print(self.s)**  **str1=IOString()**  **str1.getString()**  **str1.printString()** |



EX 6: 試寫出一個命名為Employee的Python類別，該類別由兩個兩個方法(methods)所構成，其中displayEmployee可以列印受僱員的名字和薪水，displayCount則是輸出受僱者的總數。

例如:

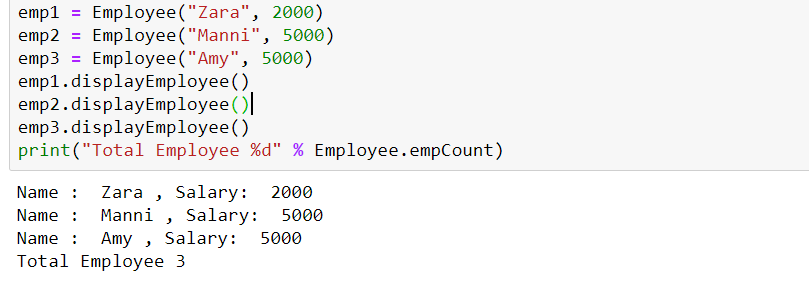


圖5-21　輸入與輸出結果

|  |
| --- |
| **class Employee():**  **empCount=0**  **def \_\_init\_\_(self, name, money):**  **self.name=name**  **self.money=money**  **Employee.empCount+=1**  **def displayEmployee(self):**  **print("Name",self.name,end="")**  **print(",Salary:",self.money)**  **def displayCount(self):**  **self=""**    **emp1=Employee("Zara",2000)**  **emp2=Employee("Manni",5000)**  **emp3=Employee("Amy",5000)**  **emp1.displayEmployee()**  **emp2.displayEmployee()**  **emp3.displayEmployee()**  **print("Total Employee %d"%Employee.empCount)** |



EX 7: 對於一個運動學公式: S = Vot + 1/2 gt2，其中Vo代表初速，t為時間，g = 9.8。試寫出一個命名為kinematic的Python類別，該類別的輸入參數為初速Vo，而類別裡的方法(disFormula)在給予時間t這個參數後，可以計算出移動距離S。

例如:

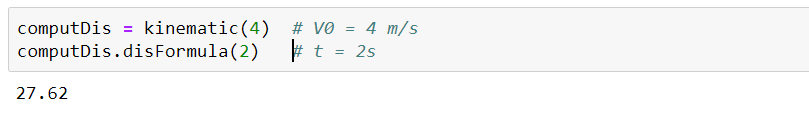
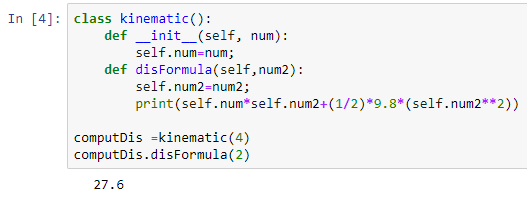


圖5-22　輸入與輸出結果

|  |
| --- |
| **class kinematic():**  **def \_\_init\_\_(self, num):**  **self.num=num;**  **def disFormula(self,num2):**  **self.num2=num2;**  **print(self.num\*self.num2+(1/2)\*9.8\*(self.num2\*\*2))**    **computDis =kinematic(4)**  **computDis.disFormula(2)** |



EX 8: 試建立一個命名為calculation的Python模組，該模組提供加、減、乘、除等功能的運算。

例如:

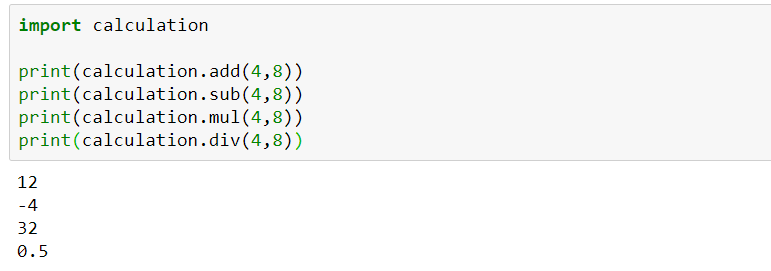


圖5-23　輸入與輸出結果

Calculation.py

|  |
| --- |
| **import calculation**  **print(calculation.add(4,8))**  **print(calculation.sub(4,8))**  **print(calculation.mul(4,8))**  **print(calculation.div(4,8))** |

