

物件導向設計

姓名：

學號：

班級：資訊三甲

日期：2012/ 6/ 2

作業四

1. 程式簡介：

程式名稱：	Chap04_Q11
程式說明：	按照題目的需求儲存 Pizza 資料，以及輸出。
程式功能：	<ol style="list-style-type: none">1. 產生一個新的 Pizza 物件。2. 儲存輸入的 Pizza 大小以及配料量。3. 計算費用4. 將資料完整顯示出來。

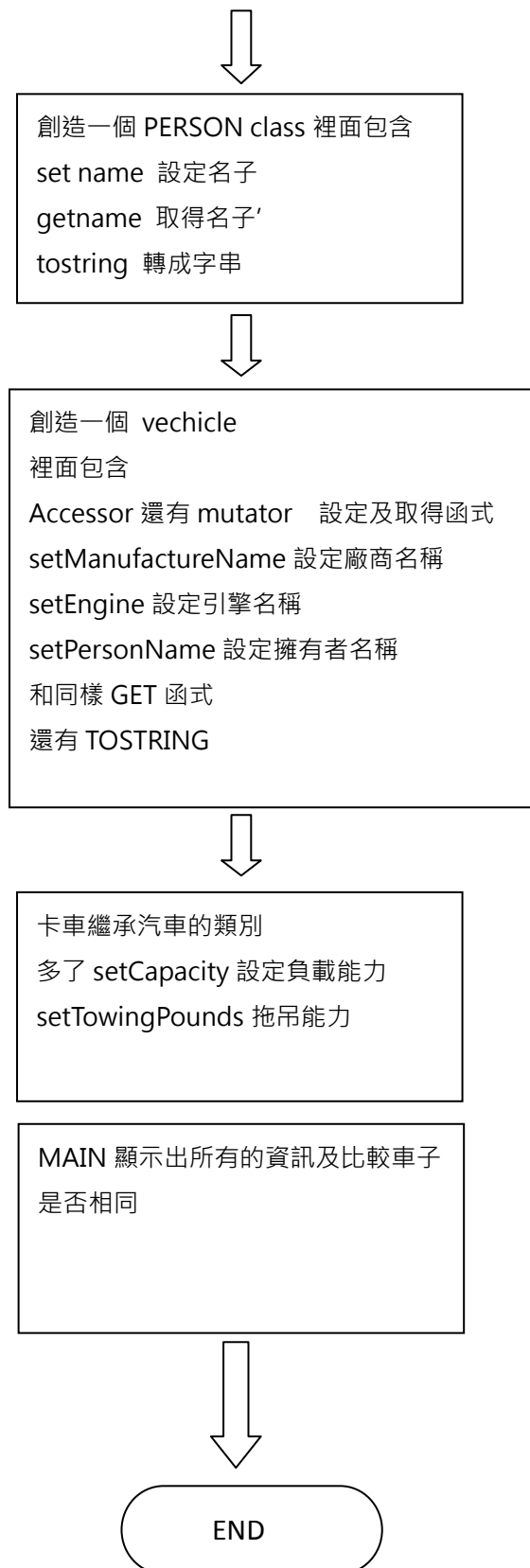
2. 程式規格：

Chap04_11 程式規格：	
3 個 Class Pizza object：	1.儲存 Pizza 詳細資訊(pizza1,pizza2,pizza3)

Class Pizza 的程式規格：	
(一)1 個 private String：	1. 儲存 Pizza 的大小(size)。
(二)3 個 private int：	<ol style="list-style-type: none">1. 儲存 cheese 的數量(num_cheese_toppings)。2. 儲存 pepperoni 的數量(num_pepperoni_toppings)。3. 儲存 ham 的數量(num_ham_toppings)。
(三)1 個 constructor：	1. 設定物件初值(Pizza())。
(四)5 個 mutator method：	<ol style="list-style-type: none">1. 修改 size 內容(setSize(String))。2. 修改 cheese 內容(setNum_cheese_toppings (int))。3. 修改 pepperoni 內容(setNum_pepperoni_toppings (int))。4. 修改 ham 內容(setNum_ham_toppings (int))。5. 修改 size, cheese, pepperoni,ham 內容 setPizza(String size , int num_cheese_toppings , int num_pepperoni_toppings ,int num_ham_toppings)
(五)5 個 accessor method：	<ol style="list-style-type: none">1. 取得 Pizza 的詳細敘述(getDescription())。2. 取得 size 內容

	(getSize ()) 。 3. 取得 cheese 內容 (getNum_cheese_toppings ()) 。 4. 取得 pepperoni 內容 (getNum_pepperoni_toppings ()) 。 5. 取得 ham 內容 (getNum_ham_toppings ()) 。
(六)1 個計算 method :	計算花費(calcCost()) 。

3. 流程圖：



4. 部份程式碼解說：

(1) 設定 instance variables。

```
//三個instance  
private String A_title; //Administrator's title  
private String A_area; //Administrator's area  
private String A_name; //Administrator's name
```

(2) 三個 constructor。(用多載)

//四種建構方法

1. **public** Administrator(); //空建構子
2. **public** Administrator(String E_Name, Date E_hireDate, **double** E_Salary, String Name, String Title, String Area);
3. **public** Administrator(Administrator originalObject);

1. 為空建構子，其內容就是將instance variable設為適合的空值，在此也要呼叫父親類別的建構子來幫忙做建構。
2. E_Name, E_hireDate, E_Salary 這些為繼承下來的variable，我們可以利用父親類別來將這些屬於父親的variable來做建構，我們只要負責Name, Title, Area 的初始建構工作。
3. 其參數為傳一個物件進來，做出類似複製物件的工作，一樣要呼叫到父親的建構子來幫忙做。

(3) 3 個 accessor, 3 個 mutator(用多載)

```
// accessor//  
public String gettitle( );  
public String getarea( );  
public String getname( );  
  
// mutator//  
public void setttitle(String title);  
public void setarea(String area);  
public void setname(String name);
```

三個accessor只是單純的將物件的instance variable 分別輸出來做使用。

三個accessor只是單純的將物件的instance variable 分別設定這些值。

(5) 按照題目的要求利用 **overwrite** 來寫 **equals** 跟 **toString**

```
public String toString( )  
{  
    return (super.toString() + "\nadministered " + getname() + "("  
    +gettitle()+ ")" + " in " + getarea()+"\n");  
}
```

以上為toString的函式，我們利用`super.toString()`，利用父親類別的toString() 幫我們做掉一些變數的輸出工作，我們只需要補齊剩下變數的輸出工作即可。

```
public boolean equals(Object otherObject)
{
    if (otherObject == null)
        return false;
    else if (getClass() != otherObject.getClass())
        return false;
    else
    {
        Administrator otherSalariedEmployee =
            (Administrator)otherObject;

        return (super.equals(otherSalariedEmployee)
            && (A_title.equals(otherSalariedEmployee.A_title))
            && (A_area.equals(otherSalariedEmployee.A_area))
            && (A_name.equals(otherSalariedEmployee.A_name)));
    }
}
```

以上為 equals 的函式，我們先檢查其是否為空，然後再判斷其是否與本身的 object 出字相同的 class，可用 getClass() 來達成，若都通過，我們才檢查其所有的 variable 是否相同，其中一部分得變數，我們亦可用 `super.equals(otherSalariedEmployee)`，來幫忙我們做，我們只需要做剩下的變數判斷即可。

5. 心得：

這個作業主要練習如何自定義 **Class** 並且宣告物件來達到自己想要的結果，按照題目的需求打其實只能打出部分的程式碼出來，對於如何將程式完成的更完整得花費時間思考才行，在寫報告時有問過同學要如何寫才能寫出一份完整的，這份作業的不完整讓我得知自己在程式上的思考需要多加強才行。