

物件導向設計 homework 3

Q3

1. 程式簡介

程式名稱:	D0687151_homework3_Q3
程式說明:	<p>一個外星人遊戲。請改寫題目提供範例，使其符合 object oriented，並且刪除” type”，將不同種類外星人寫成 class 繼承 Alien，且需符合以下標準。</p> <ol style="list-style-type: none">1. 隱藏 Alien class 的 instance variables。2. Alien 衍生 class 需有自己的 getDamage 方法。3. 計算在 AlienPack 中，所有外星人的攻擊傷害。
程式功能:	<ol style="list-style-type: none">1. 建立三個不同種類的外星人2. 將外星人物件加入 AlienPack 中3. 輸出所有外星人的姓名、血量和傷害4. 輸出 AlienPack 中所有外星人的攻擊傷害

2. 程式規格

D0687151_homework3_Q3 程式規格	
一個 class Alien	為父類別，初始最原始外星人
一個 class AlienPack	宣告 alien 陣列，放入不同種的外星人
一個 class Marshmallow	外星人種類之一，繼承 Alien
一個 class Ogre	外星人種類之一，繼承 Alien
一個 class Snake	外星人種類之一，繼承 Alien
一個 class main	主程式

Class Alien 的程式規格	
(一) 一個 private String	紀錄外星人姓名
(二) 一個 private int	紀錄外星人血量
(三) 兩個 Alien 建構子	一個空建構子和一個初始姓名和血量建構子
(四) 兩個 accessor method	取得外星人的名字和血量
(五) 兩個 mutator method	修改外星人的名字和血量
(六) 一個 toString()	輸出外星人的名字和血量
(七) 一個 getDamage	回傳 Alien 傷害(0 點傷害)

Class AlienPack 的程式規格	
(一) 一個 private aliens 陣列	存放外星人物件
(二) 一個 AlienPack(int num)建構子	建立一個大小為 num 的 Alien 陣列
(三) 一個 addAlien(Alien , int)	將外星人加到陣列裡
(四) 一個 getAliens()	回傳陣列內容
(五) 一個 calculateDamage()	回傳陣列內所有外星人的總傷害

Class Marshmallow 的程式規格	
(一) 兩個 Marshmallow 建構子	一個空建構子和一個初始姓名和血量建構子(請用 super)
(二) 一個 getDamage()	回傳 Marshmallow 傷害(1 點傷害)

Class Ogre 的程式規格	
(一) 兩個 Ogre 建構子	一個空建構子和一個初始姓名和血量建構子(請用 super)
(二) 一個 getDamage()	回傳 Ogre 傷害(6 點傷害)

Class Snake 的程式規格	
(一) 兩個 Snake 建構子	一個空建構子和一個初始姓名和血量建構子(請用 super)
(二) 一個 getDamage()	回傳 Snake 傷害(10 點傷害)

3. 流程圖

創造一個 Alien 類別
 裡面包含
 Accessor 和 mutator 設定及取得函式
 getName 取得姓名
 getHealth 取得生命值
 和同樣 set 函式
 還有 toString 函式
 和 getDamage 函式



Marshmallow 繼承 Alien 類別

Orge 繼承 Alien 類別

Snake 繼承 Alien 類別

三個類別都建立自己的 getDamage 函式



AlienPack 繼承 Alien 類別

建立 AlienPack 建構子 //建立 Alien 陣列

建立 addAlien 函式 //加入物件到陣列

建立 calculateDamage 函式

計算陣列裡的外星人總傷害



Main:

建立不同的外星人各一個

建立 AlienPack 物件把這三個物件都放進去

輸出資訊 還有 回傳陣列裡外星人總傷害



END

4. 部分程式碼解說

Alien class:

```
private String name;
```

```
private int health;
```

1. 外星人姓名與生命值

```
public int getDamage() {
```

```
    return 0;
```

```
}
```

2. 回傳傷害 0

Marshmallow class:

```
public int getDamage() {
```

```
    return 1;
```

```
}
```

1. 回傳傷害 1

Orge class:

```
public int getDamage() {
```

```
    return 6;
```

```
}
```

1. 回傳傷害 6

Snack class:

```
public int getDamage() {
```

```
    return 10;
```

```
}
```

1. 回傳傷害 10

AlienPack class:

```
public int calculateDamage() {
```

```
    int sum = 0;
```

```
    for(int i = 0; i < aliens.length; i++) {
```

```
        sum += aliens[i].getDamage();
```

```
    }
```

```
    return sum;
```

```
}
```

1. 回傳物件內的外星人總傷害

5.心得:這個作業主要練習 **overloading** 還有建立父類別陣列來存放不同的子孫類別(練習動態繫節)

Q6

1. 程式簡介

程式名稱：	學號_homework3_Q6
程式說明：	<p>一個紀錄卡車駕駛人程式。請創造一個 Vehicle class，包含 manufacturer、cylinders 和 owner(Person class)的 instance variables。</p> <p>再寫一個 Truck class 去繼承 Vehicle，且多加 load 和 towing 的 instance variables，且確保上述 class 有合理的 accessor、mutator、toString 和 equals。</p>
程式功能：	<ol style="list-style-type: none"> 1. 建立三個 Truck 物件(利用建構子設值) 2. 輸出三個 Truck 物件內容 3. 兩兩比較 Truck 物件內容是否一樣

2. 程式規格

學號_homework3_Q6	
一個 class Person	題目提供改寫範例，為車主的類別
一個 class Vehicle	父類別，初始車子
一個 class Truck	繼承 Vehicle
一個 class main	主程式

Class Person 的程式規格	
(一) 一個 private String	紀錄車子的擁有者名字
(二) 三個 Person 建構子	依照輸入的參數對應不同的建構子建立 Person 物件，有空建構子、string 參數和 Person 物件的參數傳入建構子。
(三) 一個 accessor method	取得名字
(四) 一個 mutator method	設定名字
(五) 一個 toString()	印出名字
(六) 一個 equals()	比較兩 Person 是否相同

Class Vehicle 的程式規格	
(一) 一個 private String	紀錄車子的生產商(manufacturer)
(二) 一個 private int	紀錄車子的汽缸(cylinders)

(三) 一個 protected Person	紀錄車子的擁有者
(四) 兩個 Vehicle 建構子	一個為空建構子，另一個傳入參數為 Person 物件、manufacturer 和 cylinders 的建構子
(五) 兩個 accessor method	取得生產商和汽缸
(六) 兩個 mutator method	設定生產商和汽缸
(七) 一個 toString()	印出生產商名、汽缸、車主名
(八) 一個 equals()	比較兩 Vehicle 是否相同

Class Truck 的程式規格	
(一) 一個 private double	紀錄車子的承載量(load)
(二) 一個 private int	紀錄車子的牽引能力(towing)
(三) 兩個 Truck 建構子	一個為空建構子，另一個傳入參數為 Person 物件、manufacturer、cylinders、load 和 towing 的建構子。(請用 super)
(四) 兩個 accessor method	取得承載量和牽引能力
(五) 兩個 mutator method	設定承載量和牽引能力
(六) 一個 toString()	印出生產商名、汽缸、承載量、牽引能力和車主名
(七) 一個 equals()	比較兩 Truck 是否相同

3.流程圖

創造 Person 類別
 裡面包含 accessor method 和 mutator method
 getName 取得姓名
 和同樣 set 函式
 toString 和 equals 函式



創造 Vehicle 類別
同樣 accessor method 和 mutator method
setManufacturer 設定生產商
setCylinders 設定汽缸
同樣 get 函式
還有 toString 跟 equals 函式



Truck 類別繼承 Vehicle
多了 setLoad 設定承載量
setTowing 設定牽引能力



Main:
建立 person 物件陣列放三個名字
建立 Truck 物件陣列分別把 person 物件放入創造三個 Truck 物件
輸出所有資訊
比較車子是否相同



END

4. 部分程式碼解說

Vehicle 類別：

```
//三個instance  
private String manufacturer;  
private int cylinders;  
protected Person person ;
```

Truck 類別：

```
//兩個instance  
private double load;  
private int towing;  
//toString method  
public String toString() {  
  
    return super.toString() //利用 Vehicle 類別的toString  
        + "Load: " + String.format("%-10.1f", this.load)  
        + "Towing: " + String.format("%-5d", this.towing);  
}
```

5. 心得：

練習繼承，還有把不同類別的物件當作 instance variable，實作沒有遇到任何困難。