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: | 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。   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: | 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toStrong() | 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calcaulateDamage 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: | 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。  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。 |
| 程式功能: | 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、sting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

setManfacturer 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

1. **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);

}

1. **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。