ConsoleApplication1 (全域範圍) main()

```cpp
#include <iostream>
#include "s1091436_zoo_stu_hw5.h"

int main()
{
    std::cout << "Hello World!\n";
    Zoo ZooSim1(5);

    ZooSim1.AddAnimal(new horse("Pony", 100));
    ZooSim1.AddAnimal(new pig("Piggy", 100));
    ZooSim1.AddAnimal(new human("Pony", 100));
    ZooSim1 += new pig("Piggy", 100);
    ZooSim1.Listing();
    std::cout << "==================================\n";
    //...example...
    std::cout << ZooSim1 << std::endl;
    ZooSim1.Run(3);
    std::cout << "==================================\n";
    std::cout << "---Test getMorefood()--- \n";
    std::cout << "Before:  "<< ZooSim1 << std::endl;
    ZooSim1.getMoreFood(100);
    std::cout << "After:  " << ZooSim1 << std::endl;
    ZooSim1.getMoreFood(500);
```

選取 Microsoft Visual Studio 偵錯主控台

```
Hello World!
Pony (hungry: 0, energy : 100)
Piggy (hungry: 0, energy : 100)
Pony (hungry: 0, energy : 100)
Piggy (hungry: 0, energy : 100)
==================================
[INFO] Food: 500, Water: 100, Foud$: 100$

Pony (hungry: 35, energy : 65)
Piggy (hungry: 20, energy : 65)
Pony (hungry: 35, energy : 55)
Piggy (hungry: 20, energy : 65)
[LOG] ask all animals to rest.
[LOG] horse take a rest, eat 0 food & drink 10 water, gain 10 energy
[LOG] pig take a rest, eat 10 food & drink 0 water, gain 0 energy
[LOG] human take a rest, eat 10 food & drink 20 water, gain 20 energy
[LOG] pig take a rest, eat 10 food & drink 0 water, gain 0 energy
[LOG] Feeding food...
[LOG] horse eat 50 food
[LOG] pig eat 50 food
[LOG] human eat 50 food
[LOG] pig eat 50 food
[LOG] total consume 200 food.
[LOG] Feeding water...
[LOG] horse drink 10 water
[LOG] pig drink 10 water
[LOG] human drink 10 water
[LOG] pig drink 10 water
[LOG] total consume 40 water.
Pony (hungry: 35, energy : 150)
```

基本功能都能執行，且並無問題產生。

```cpp
int Zoo::Run(int quantity)
{
    day++;
    cout << "today is " << day << " day\n";
    int amount = 0;
    int money;
    //you SHOULD update this function
    for (int i = 0; i < quantity; i++)
    {
        for (std::vector<Animal*>::iterator iter = house.begin(); iter != house.end(); ++iter)
        {
            money= (*iter)->exercise(1);
            amount += money;
            (*iter)->print();
        }
        AskRest();
        FeedFood(1);
        FeedWater(1);
    }
    if (foodCount == 0)
    {
        getMoreFood(100);
    }
    waterCount += 100;
    deposit += amount;
    return amount;
}
```

我的策略是讓每隻動物運動完後都會喝水一次和吃食物一次，食物是每天才會去檢查是否充足並再添購，水則是每天補充100，並我添加了天數，讓執行時可以更好觀察。