Bees Exercise

Part 1- Classes

Decided 4 classes would be created

- Bees- Abstract parent class
- Worker- child class
- Queen – child class
- Drone – child class

Bee Class

Bee class would be an abstract class because the class itself would not be instantiated.

The classes share the same properties so to prevent repeat code; 3 types of bee class would inherit the properties and methods from the bee abstract class.

Type

Type property represents the string type of the bee ("worker", "queen", "drone")

Health

The health property is a type float and contains a private backing field called "_health", which is set to 100. The health property has a private setter so that it can't be externally writable.

Alive and MinHealth

The property Alive is used to check whether the bee is alive or not by checking if the "Health" is greater than or equalled to "MinHealth" property which represents the limited health of the bee. If the health property is less than "MinHealth" then the bee would be considered as dead.

Damage

A bee takes damage whether it is still alive and if the damage amount is between 0 and 100. If one of the conditions is not fulfilled, then the bee will take 0 damage.

Worker, Queen and Drone Class

Each of these classes inherits the properties and methods of the Bee class and sets the constructor according to its class.

Part 2- The Application

The UI was developed using WPF.

Main Window Class

PopulateBeeBox()

A Listbox was used to show all 10 instances of each Bee type. This was done by using a for loop to create each of the Bee types 10 times and then storing them within a list which is then shown onto the Listbox(See PopulateBeeBox() function in MainWindow.xaml.cs).

Damage_Click()

Damage button is shown on the UI and when clicked would decrease the Bee's health by a random amount between 0 and 100 which then updates the Alive and Health Text.

Unit Test

Unit Test was also created to check if the values and outputs of the Health of the bees were correct. However, the unit test could only be done by changing the health property setter to public because the private keyword would not allow the test to run.