|  |  |  |
| --- | --- | --- |
| **LAB211 Assignment** | **Type:** | **Short Assignment** |
| **Code:** | **J1.S.P0081** |
| **LOC:** | **90** |
| **Slot(s):** | **2** |

**Title**

Bees

**Background**

NA

**Program Specifications**

Write classes to represent 3 different types of Bee - Worker, Queen and Drone.

Each Bee has a floating-point health property, which is not writable externally and upon creation is set to a value of 100 (percent).

Each Bee has a Damage() method that takes a single integer parameter that should be a value between 0 and 100. When this method is called, the health of the bee is to be reduced by that percentage of their current health.

When a Worker has a health below 70% it cannot fly and therefore is pronounced Dead. When a Queen has a health below 20%, or a Drone below 50%, it is pronounced dead. This 'Dead' property should be readable from each Bee. When a Bee is dead, no further health deductions should be recorded by the bee, although the Damage() method should still be invokable without error.

Your application should create a single list containing 10 instances of each type of Bee and store in a list or array. It must support methods to allow Damage() to be called for each Bee, and to return the health status of each bee, including whether it is alive or not.

Your application interface must contains 2 functions (user press “1” or “2” to activate this function :

1 – Create bee list – Clear current bee list and create new random bees, then display in the console windows

2 – Attack bees - Attack current bee list, a different random value between 0 and 80 should be selected for each bee and applied with a call to Damage(). After attacked, the user interface should refresh to show the health status of the bees in console windows

### *Expectation of User interface:*

NA

**Guidelines**

NA