Caleb Drake
CM #1065

[Question 1 (4 pt)]
Explain, in your own words, the process used to mock the database and test the getRoomOccupant function.

Defined the test method
Created a mock stub of type IDatabase
Defined *room occupants* of type String
Recorded desired results
Set-up target as a Hotel with a 10-night stay and database set as our mock DB
Composed Assertions Tests as normal

[Question 2 (4 pts)]
Specify how you can use the LastCall class to throw an exception.

LastCall.Throw(Exception exception)

[Question 3 (4 pts)]
In the above example, we used a \stub" since the mocked object returned a value. Do I need to use a stub if the mocked object did not return a value? Can I replace the stub with a DynamicMock?

No
Yes

[Question 4 (4 pts)]
Explain, in your own words, the process used to mock the database and test the AvailableRooms property.

Defined the test method
Created a mock stub of type IDatabase
Defined Rooms of type List<32>
Assigned Rooms to the mock DB's Room List
Set-up target as a Hotel with a 10-night stay and database set as our mock DB
Composed Assertions Tests comparing the number of rooms fetched from the Database and count of the *Rooms* list.

[Question 5 (4 pts)] Explain, in your own words, the process used to ensure that the service locator removes a car from its available cars when a car is booked.

Define the test method and create a ServiceLocator and car objects
Adde cars to serviceLocator
set the bounds within global ServiceLocator.Instance
Composed Assertions Tests as normal