**Question 1**

How do you encapsulate the LiveData stored in a ViewModel so that external objects can read data without being able to update it?

* Inside the ViewModel object, change the data type of the data to private LiveData. Use a backing property to expose read-only data of the type MutableLiveData.
* Inside the ViewModel object, change the data type of the data to private MutableLiveData. Use a backing property to expose read-only data of the type LiveData.
* Inside the UI controller, change the data type of the data to private MutableLiveData. Use a backing property to expose read-only data of the type LiveData.
* Inside the ViewModel object, change the data type of the data to LiveData. Use a backing property to expose read-only data of the type LiveData.

**Question 2**

LiveData updates a UI controller (such as a fragment) if the UI controller is in which of the following states?

* Resumed
* In the background
* Paused
* Stopped

**Question 3**

In the LiveData observer pattern, what's the observable item (what is observed)?

* The observer method
* The data in a LiveData object
* The UI controller
* The ViewModel object