COMP212   Programming 3
2020M - Lab Assignment 3
Implement WPF app to manipulate data within Database

Goals: In this assignment, you will learn how to use the **WPF** controls DataGrid, ComboBox, Button. You will also learn how to make your **WPF** app create and interact with a database by exploiting two classes **DbContext** and **DbSet\<TEntity>** of Entity Framework (specifically Entity Framework 6 (EF6). You can think EF6 to be a restricted version of EF Core).

Rubric: (Total marks = 10, Weight = 10%)

Cases
Three DataGrid, two ComboBoxes, six Buttons are present in the GUI layout.
When you select an item in Fruit ComboBox, the selected item shows up in a row in DataGrid-1 (left DataGrid).
When you select an item in Planet ComboBox, the selected item shows up in a row in DataGrid-2 (middle DataGrid).
When you select an item in the DataGrid-1, the selection (if any) at DataGrid-2 should get unselected.
When you select an item in the DataGrid-2, the selection (if any) at DataGrid-1 should get unselected.
An element in DataGrid (i.e. a grid location) should not be editable.
On clicking button "LINQ Project QS" shows relevant rows in DataGrid-3 (right DataGrid).
On clicking button "LINQ Filter QS" shows relevant rows in DataGrid-3.
On clicking button "LINQ Order Ascending QS" shows relevant rows in DataGrid-3.
On clicking button "LINQ Inner Join QS" shows relevant rows in DataGrid-3.
Button "Clear" clears all three DataGrids, clears the selections in every ComboBox
Button "Delete selected row" deletes the selected row in DataGrid-1 (left DataGrid) or the selected row in DataGrid-2 (middle DataGrid).
Button "Delete selected row" should not delete any selected row in DataGrid-3 (right DataGrid).
When DataGrids are empty (i.e. they have no rows of items), ComboBoxes should be in reset condition (i.e. they should have no item selected)
When DataGrids are empty, clicking any button should have no effect on the GUI.

Submission of Assignment:

1. Slides/Code related to lectures on WPF, LINQ may be useful.

2. Make sure that there is one project folder Proj1 under the Visual Studio 2019 solution folder Lab3.

3. Make sure your code **compiles** (and runs) in **Visual Studio 2019** installed in a **Windows 10** computer before submission.

4. In Visual Studio, in Solution Explorer, right-click on the solution Lab3 and choose Clean Solution.

5. Compress the Visual Studio solution folder Lab3 and all its contents into a single zip file. This will be your submission file. You MUST name your submission file according to the following rule:

   `YourFirstname_YourLastname_COMP212_Labnumber.zip`

   Example: John_Smith_COMP212_Lab3.zip

6. **Next, submit above zip file on eCentennial** through the **assignment** link.

Develop a WPF application that interacts with database through Entity Framework 6.

1. Below are several pictures of how your GUI should look. **The rubric in Page 1 of this document tells how your application should work. Read the figure captions carefully as well.** They also provide hint on how the GUI should work.
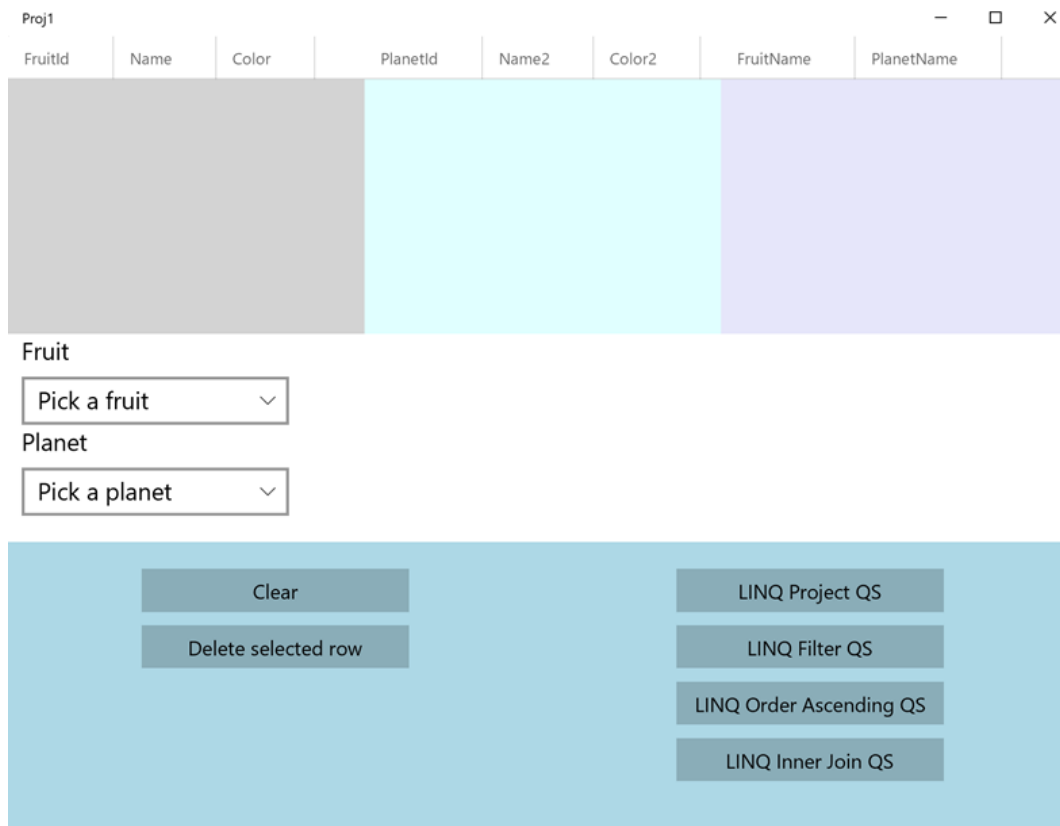
Figure 1. The GUI when launched.

2. There are Three DataGrids (in top row), two ComboBoxes (in middle row), six Buttons (in bottom row) present in the GUI layout. **Refer to the rubric for more information**.

Figure 2. Items in Fruit ComboBox.

FruitId	Name	Color	PlanetId	Name2	Color2	FruitName	PlanetName

Fruit

Pick a fruit ⌄
Earth red
Jupiter blue

Clear		LINQ Project QS
Delete selected row		LINQ Filter QS
		LINQ Order Ascending QS
		LINQ Inner Join QS

Figure 3. Items in Planet ComboBox.

FruitId	Name	Color	PlanetId	Name2	Color2	FruitName	PlanetName
8	kiwi	red	6	Earth	red		
9	grape	blue	7	Jupiter	blue		
10	dates	red					
11	pear	blue					
12	kiwi	red					

Fruit

| kiwi red | ∨ |

Planet

| Jupiter blue | ∨ |

Clear		LINQ Project QS
Delete selected row		LINQ Filter QS
		LINQ Order Ascending QS
		LINQ Inner Join QS

Figure 4. A sample set of items in DataGrid-1 (left datagrid) and DataGrid-2 (middle datagrid). **Note:** The **FruitId** **MUST NOT** be set by you. It should be automatically created for you when a Fruit object is saved in the database as a row. Same argument holds for the **PlanetId** of a Planet object.

FruitId	Name	Color		PlanetId	Name2	Color2		FruitName	
8	kiwi	red		6	Earth	red		kiwi	
9	grape	blue		7	Jupiter	blue		grape	
10	dates	red						dates	
11	pear	blue						pear	
12	kiwi	red						kiwi	

Fruit

kiwi red	∨

Planet

Jupiter blue	∨

Clear		LINQ Project QS
Delete selected row		LINQ Filter QS
		LINQ Order Ascending QS
		LINQ Inner Join QS

Figure 5. The relevant items in DataGrid-3 (right datagrid) on clicking button "LINQ Project QS".

FruitId	Name	Color		PlanetId	Name2	Color2		FruitName	
8	kiwi	red		6	Earth	red		kiwi	
9	grape	blue		7	Jupiter	blue		dates	
10	dates	red						kiwi	
11	pear	blue							
12	kiwi	red							

Fruit

kiwi red   ⌄

Planet

Jupiter blue   ⌄

Clear	LINQ Project QS
Delete selected row	LINQ Filter QS
	LINQ Order Ascending QS
	LINQ Inner Join QS

Figure 6. The relevant items in DataGrid-3 (right datagrid) on clicking button "LINQ Filter QS". Here the "filter" is the color **red**.

FruitId	Name	Color		PlanetId	Name2	Color2	AscendingName	
8	kiwi	red		6	Earth	red	dates	
9	grape	blue		7	Jupiter	blue	grape	
10	dates	red					kiwi	
11	pear	blue					kiwi	
12	kiwi	red					pear	

Fruit

kiwi red	⌄

Planet

Jupiter blue	⌄

Clear	LINQ Project QS
Delete selected row	LINQ Filter QS
	LINQ Order Ascending QS
	LINQ Inner Join QS

Figure 7. The relevant items in DataGrid-3 (right datagrid) on clicking button "LINQ Order Ascending QS".

FruitId	Name	Color		PlanetId	Name2	Color2		FruitName	PlanetName
8	kiwi	red		6	Earth	red		kiwi	Earth
9	grape	blue		7	Jupiter	blue		grape	Jupiter
10	dates	red						dates	Earth
11	pear	blue						pear	Jupiter
12	kiwi	red						kiwi	Earth

Fruit

kiwi red ⌄

Planet

Jupiter blue ⌄

Clear		LINQ Project QS
Delete selected row		LINQ Filter QS
		LINQ Order Ascending QS
		LINQ Inner Join QS

Figure 8. The relevant items in DataGrid-3 (right datagrid) on clicking button "LINQ Inner Join QS".