CSIS 3540 Assignment 1 – Real Estate Transactions

Introduction

Name of the project / solution: **AS1ProjectTeamNN**, where NN is your project team number. **You will lose marks for not naming it correctly.** For example, if you are Project Team 2, then the project and solution name will be: **AS1ProjectTeam02.**

Note that the .sln file and all relevant directories MUST be named correctly.

Name the Form file **RealEstateTranactionsForm.cs**, and the Program file **RealEstateTransactionsProgram.cs**.

You need to follow the same coding standards as shown in class with regard to file, form, control and variable naming, use of constants, comments, indentation etc.

The Application

The task is to write an application that analyzes real estate (housing) transactions.

You should work as a team on this assignment. One member should be the project lead, and will assign parts of the assignment to members. The project lead will also assemble the final project and submit it via blackboard.

Requirements:

1. You must read an XML file (**SacramentoRealEstateTransactions.xml**) to get the transactions. This is based on actual house sales in the Sacramento, California area over a multi-month period.
2. You must use LINQ queries to order the transactions, and perform basic analyses of the data (average price and count).
3. You must use classes to represent the transaction objects.
4. You must use the DataGridView, ListBox and other controls.
5. Filter controls (ListBox, Checkbox, TextBox, Button) must be properly initialized and respond properly to events.

Application Functioning and Design

When the application starts up, it should prompt the user for a file, as in the following:
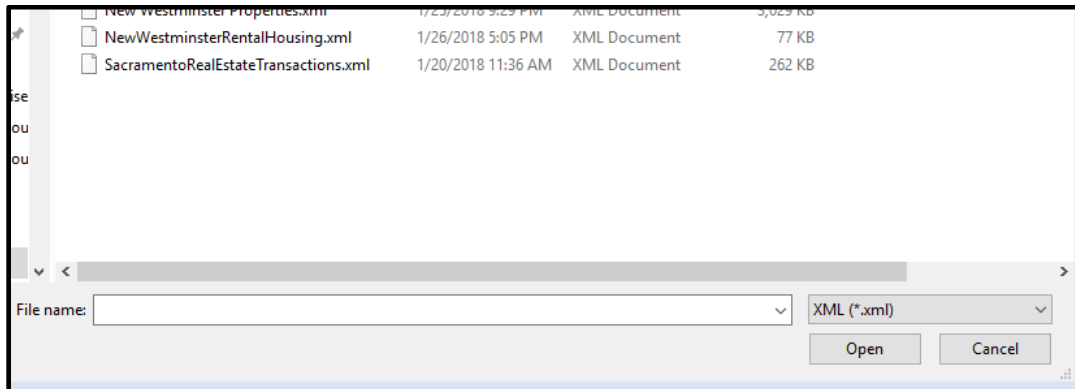
The user should select the **SacramentoRealEstateTransactions.xml** file.

The file is in XML format.

The application should then read the file, and store the information in a list of objects of the type House, which should contain the following properties:

- Address
- City
- HouseType
- SurfaceArea
- Bedrooms
- Bathrooms
- Price

NOTE: If using XML Serializable, your serialized House class MUST name these properties EXACTLY as above and in the EXACT order. View the XML file to examine the data types for properties above.

```xml
<?xml version="1.0" encoding="utf-8"?>
<ArrayOfHouse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <House>
    <Address>3526 HIGH ST</Address>
    <City>SACRAMENTO</City>
    <HouseType>Residential</HouseType>
    <SurfaceArea>836</SurfaceArea>
    <Price>59222</Price>
    <Bedrooms>2</Bedrooms>
    <Bathrooms>1</Bathrooms>
  </House>
```

Once the file is processed, the application should load the form shown below:

The form should be named RealEstateTransactionsForm, and the form text should read as above. All controls should be laid out in the same fashion as above.

There are two DataGridView controls. The top control always contains the full list of transactions, and does not change. There are 985 in total. The columns should be right-justified. The data should be sorted by City, HouseType, then Price (use LINQ).

The bottom control always displays the results of the filters that are set. Note that it has the same structure as the top control, and should be sorted by City, HouseType, and Price (use LINQ).

Under both DataGridView controls, Count and Average Price must be reported (use LINQ).

Filter controls are in the middle of the form. There are four ListBox filters: Cities, Bedrooms, Baths and HouseTypes.

To filter on Price or SurfaceArea, the user must provide max and min values in the TextBoxes. The filter is activated only if the user checks the relevant Checkbox for Price or SurfaceArea.

Initial State

When the program starts, both DataGridView controls should display the exact same information. This is the default initial state.

For each of the ListBox controls which serve as filters, a unique list of all values for the control must be generated then added to the control. The program should generate a list of such values from the transaction data, which will contain many duplicates, then reduce this down to a unique set of values. This is done using LINQ. The unique values only need to be set when the program initially runs. The initial state for the ListBox controls should all items selected (highlighted in blue).

For Price and Surface area, the initial state should have the CheckBoxes and all TextBoxes clear.

The bottom DataGridview control will therefore display data for all Cities, Bedrooms, Bathrooms, and House Types since all ListBox items are selected. Price and SurfaceArea are NOT included in the filter, so, again, all data should be shown.

Filters

When a ListBox item is unselected, the selected transactions in the DataGridViewControl change IMMEDIATELY. Note that the selected transactions result is provided where the selected Cities AND selected Bedrooms AND selected Bathrooms AND selected HouseTypes match the original data records.

LINQ **MUST** be used to generate the result to display.

If the user checks the Search on Price and/or Search on Surface Area boxes, then the filter must further refine the result, only selected records that fall between the max and min for each. If the CheckBoxes are checked, and the max or min TextBox is changed, the result due to the change in the TextBoxes should be shown immediately.

Assignment 1 Real Estate Transactions

All Transactions

	City	Address	Bedrooms	Bathrooms	SurfaceArea	HouseType	Price	
▶	ANTELOPE	8020 WALERGA RD	2	2	836	Condo	115000	
	ANTELOPE	3828 BLACKFOOT WAY	3	2	1088	Residential	126640	
	ANTELOPE	3604 KODIAK WAY	3	2	1206	Residential	142000	
	ANTELOPE	8636 LONGSPUR WAY	3	2	1670	Residential	157296	
	ANTELOPE	5708 RIDGEPOINT DR	2	2	1043	Residential	161250	

Count = 985   Average Price = $234,144.26

Filters   Reset Filters

Cities
RANCHO CORDOVA
RANCHO MURIETA
RIO LINDA
ROCKLIN
ROSEVILLE
SACRAMENTO
SHINGLE SPRINGS
SLOUGHHOUSE

Bedrooms
0
1
2
3
4
5
6
8

Bathrooms
0
1
2
3
4
5

House Types
Condo
Multi-Family
Residential
Unkown

Price   Min [   ]   Surface Area   Min [   ]

        Max [   ]   Max [   ]

☐ Search on Price   ☐ Search on Surface Area

Selected Transactions

	City	Address	Bedrooms	Bathrooms	SurfaceArea	HouseType	Price
▶	SACRAMENTO	7340 HAMDEN PL	2	2	1134	Condo	110700
	SACRAMENTO	4200 COMMERCE WAY Unit 711	2	2	958	Condo	120000
	SACRAMENTO	4800 WESTLAKE PKWY Unit 2109	2	2	1304	Condo	152000
	SACRAMENTO	301 OLIVADI WAY	2	2	1250	Condo	232500
	SACRAMENTO	3301 PARK DR Unit 1914	3	2	1678	Condo	238000
	SACRAMENTO	4236 NATOMAS CENTRAL DR	3	2	1672	Condo	265000

Count = 6   Average Price = $186,366.67

Assignment 1 Real Estate Transactions

All Transactions

	City	Address	Bedrooms	Bathrooms	SurfaceArea	HouseType	Price	
▶	ANTELOPE	8020 WALERGA RD	2	2	836	Condo	115000	
	ANTELOPE	3828 BLACKFOOT WAY	3	2	1088	Residential	126640	
	ANTELOPE	3604 KODIAK WAY	3	2	1206	Residential	142000	
	ANTELOPE	8636 LONGSPUR WAY	3	2	1670	Residential	157296	
	ANTELOPE	5708 RIDGEPOINT DR	2	2	1043	Residential	161250	

Count = 985   Average Price = $234.144.26

Filters   Reset Filters

Cities
- POLLOCK PINES
- RANCHO CORDOVA
- RANCHO MURIETA
- RIO LINDA
- ROCKLIN
- ROSEVILLE
- SACRAMENTO
- SHINGLE SPRINGS

Bedrooms
0 1 2 3 4 5 6 8

Bathrooms
0 1 2 3 4 5

House Types
- Condo
- Multi-Family
- Residential
- Unkown

Price   Min [   ]   Surface Area   Min [   ]

Max [   ]   Max [   ]

☐ Search on Price   ☐ Search on Surface Area

Selected Transactions

	City	Address	Bedrooms	Bathrooms	SurfaceArea	HouseType	Price	
▶	POLLOCK PINES	5615 LUPIN LN	3	2	1380	Residential	265000	
	POLLOCK PINES	2900 BLAIR RD	2	2	1284	Residential	280908	
	RANCHO CORD...	11150 TRINITY RIVER DR Unit 1...	2	2	941	Condo	94905	
	RANCHO CORD...	2361 LA LOMA DR	3	2	1115	Residential	116000	
	RANCHO CORD...	2460 EL ROCCO WAY	3	2	1671	Residential	175000	
	RANCHO CORD...	2421 BERRYWOOD DR	3	2	1157	Residential	180000	
	RANCHO CORD...	2909 SHAWN WAY	3	2	1452	Residential	193000	

Count = 167   Average Price = $184.514.01

All Transactions

	City	Address	Bedrooms	Bathrooms	SurfaceArea	HouseType	Price	
▶	ANTELOPE	8020 WALERGA RD	2	2	836	Condo	115000	
	ANTELOPE	3828 BLACKFOOT WAY	3	2	1088	Residential	126640	
	ANTELOPE	3604 KODIAK WAY	3	2	1206	Residential	142000	
	ANTELOPE	8636 LONGSPUR WAY	3	2	1670	Residential	157296	
	ANTELOPE	5708 RIDGEPOINT DR	2	2	1043	Residential	161250	

Count = 985   Average Price = $234,144.26

Filters   Reset Filters

Cities
- POLLOCK PINES
- RANCHO CORDOVA
- RANCHO MURIETA
- RIO LINDA
- ROCKLIN
- ROSEVILLE
- SACRAMENTO
- SHINGLE SPRINGS

Bedrooms
- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 8

Bathrooms
- 0
- 1
- 2
- 3
- 4
- 5

House Types
- Condo
- Multi-Family
- Residential
- Unkown

Price   Min   100000   Surface Area   Min   1300

Max   300000   Max   2000

☑ Search on Price   ☑ Search on Surface Area

Selected Transactions

	City	Address	Bedrooms	Bathrooms	SurfaceArea	HouseType	Price	
▶	POLLOCK PINES	5615 LUPIN LN	3	2	1380	Residential	265000	
	RANCHO CORD...	2460 EL ROCCO WAY	3	2	1671	Residential	175000	
	RANCHO CORD...	2909 SHAWN WAY	3	2	1452	Residential	193000	
	RANCHO CORD...	11130 EEL RIVER CT	2	2	1406	Residential	242000	
	SACRAMENTO	4800 WESTLAKE PKWY Unit 2109	2	2	1304	Condo	152000	
	SACRAMENTO	3301 PARK DR Unit 1914	3	2	1678	Condo	238000	
	SACRAMENTO	4236 NATOMAS CENTRAL DR	3	2	1672	Condo	265000	

Count = 78   Average Price = $196,903.97

Reset Filter Button

The Reset Filters button resets the form to the initial state when clicked.

Assignment 1 Real Estate Transactions

All Transactions

	City	Address	Bedrooms	Bathrooms	SurfaceArea	House Type	Price	
▶	ANTELOPE	8020 WALERGA RD	2	2	836	Condo	115000	
	ANTELOPE	3828 BLACKFOOT WAY	3	2	1088	Residential	126640	
	ANTELOPE	3604 KODIAK WAY	3	2	1206	Residential	142000	
	ANTELOPE	8636 LONGSPUR WAY	3	2	1670	Residential	157296	
	ANTELOPE	5708 RIDGEPOINT DR	2	2	1043	Residential	161250	

Count = 985   Average Price = $234.144.26

Filters   Reset Filters

Cities | Bedrooms | Bathrooms | House Types

ROCKLIN / ROSEVILLE / SACRAMENTO / SHINGLE SPRINGS / SLOUGHHOUSE / WALNUT GROVE / WEST SACRAMENTO / WILTON

Bedrooms: 0 1 2 3 4 5 6 8

Bathrooms: 0 1 2 3 4 5

House Types: Condo / Multi-Family / Residential / Unkown

Price   Min [ ]   Surface Area   Min [ ]

Max [ ]   Max [ ]

☐ Search on Price   ☐ Search on Surface Area

Selected Transactions

	City	Address	Bedrooms	Bathrooms	SurfaceArea	House Type	Price	
▶	ANTELOPE	8020 WALERGA RD	2	2	836	Condo	115000	
	ANTELOPE	3828 BLACKFOOT WAY	3	2	1088	Residential	126640	
	ANTELOPE	3604 KODIAK WAY	3	2	1206	Residential	142000	
	ANTELOPE	8636 LONGSPUR WAY	3	2	1670	Residential	157296	
	ANTELOPE	5708 RIDGEPOINT DR	2	2	1043	Residential	161250	
	ANTELOPE	3536 SUN MAIDEN WAY	3	2	1711	Residential	161500	
	ANTELOPE	3228 BAGGAN CT	3	2	1392	Residential	165000	

Count = 985   Average Price = $234.144.26

Error Handling

When reading the XML data file, try/catch MUST be used to handle exceptions for parsing the data. Do not assume the data is in the correct format.

More importantly, exceptions for Price and SurfaceArea must be handled. For example, if one types in a letter instead of a number and a Checkbox is set, an exception should be thrown, and the Checkbox should be cleared. The TextBoxes can remain the same so the user can edit them and correct the error.

Setting Up and Using DataGridViewControls

The following properties should be set for the DataGridView controls. The following is an example for a DataGridView d setting the properties programmatically. However, feel free to use Designer to do this, but make sure the properties are set correctly.

```
d.ReadOnly = true;  // no cell editing allowed
d.AllowUserToAddRows = false;    // no rows can be added or deleted
d.AllowUserToDeleteRows = false;
d.AutoSizeRowsMode = DataGridViewAutoSizeRowsMode.DisplayedCells;
d.RowHeadersWidth = 30;   // shorten the width of the row header
```

// right justify everything

```
d.ColumnHeadersDefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleRight;
d.DefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleRight;
```

Other properties will have to be used to set up Columns, as well as other features of the controls.

Note that when the application runs the user can sort on any column just by clicking the header. This allows to look at the data a little more in depth.

Setting Up and Using ListBox controls

Make sure the selection mode is set to multiextended, which will allow the user to select multiple items.

```
listBoxCities.SelectionMode = SelectionMode.MultiExtended;
```

The SelectedIndexChanged event will have to be handled and the SelectedItems will contain a list of selected Cities, Bedrooms, Bathrooms, or HouseTypes.

As already noted, to populate the ListBox controls, a unique list of values needs to be generated from the list of transactions. LINQ must be used for this.

One of the oddities of ListBox is to select all items in the control while in the application, the following must be done:

- Move mouse to first item in the control, and select it
- Scroll to last item
- Hold down shift key, and select last item. All items will now be selected
- Hitting the control key and selecting an item will **deselect** it. This is the usual Windows behavior.

Submission

Find the directory that CONTAINS the .sln file. The directory should be the same name as your project. Right click on this directory, and Send To Compressed File. Name the file the same as your project. It will end in .zip (for example, **AS1ProjectTeam02.zip**). Upload this to Blackboard under the assignment 1 area.

The due date is posted on Blackboard, and the .zip file must be received by that date/time.

The project lead is the only one who should submit the project.

Email submission direct to the instructor is not allowed.

Grading

The assignment will be graded on a scale of 7 points.

Criteria	Grading
Project submitted and named properly. Naming Conventions followed.	.5 points
Program fully commented.	.5 points
Form design completed, with all controls laid out correctly, including labels. Required properties all set correctly.	.5 points
Data file parsed and read into list of house objects correctly, and displayed in the top DataGridView control. Data is sorted using LINQ. Statistics coded and reported correctly.	1 point
ListBox controls implemented correctly using LINQ, and display selected transactions in lower DataGridView control. Data is filtered and sorted using LINQ. Statistics coded and reported correctly.	2.5 points
Price and SurfaceArea controls implemented correctly, and display selected transactions in lower DataGridView control. Data is filtered and sorted using LINQ.	1 point
ResetFilters Button implemented correctly. Events and exceptions handled correctly.	1 point
TOTAL	**7 points**

Criteria will be scored on a sliding scale based on quality, completeness, and correctness.

All team members will receive the same grade. If a team member has been found to have not made a proper contribution to the project, that team member will be graded significantly lower. Complete lack of a team member's participation will earn a zero grade.