The Edwardsville Auto Store - Lab 5

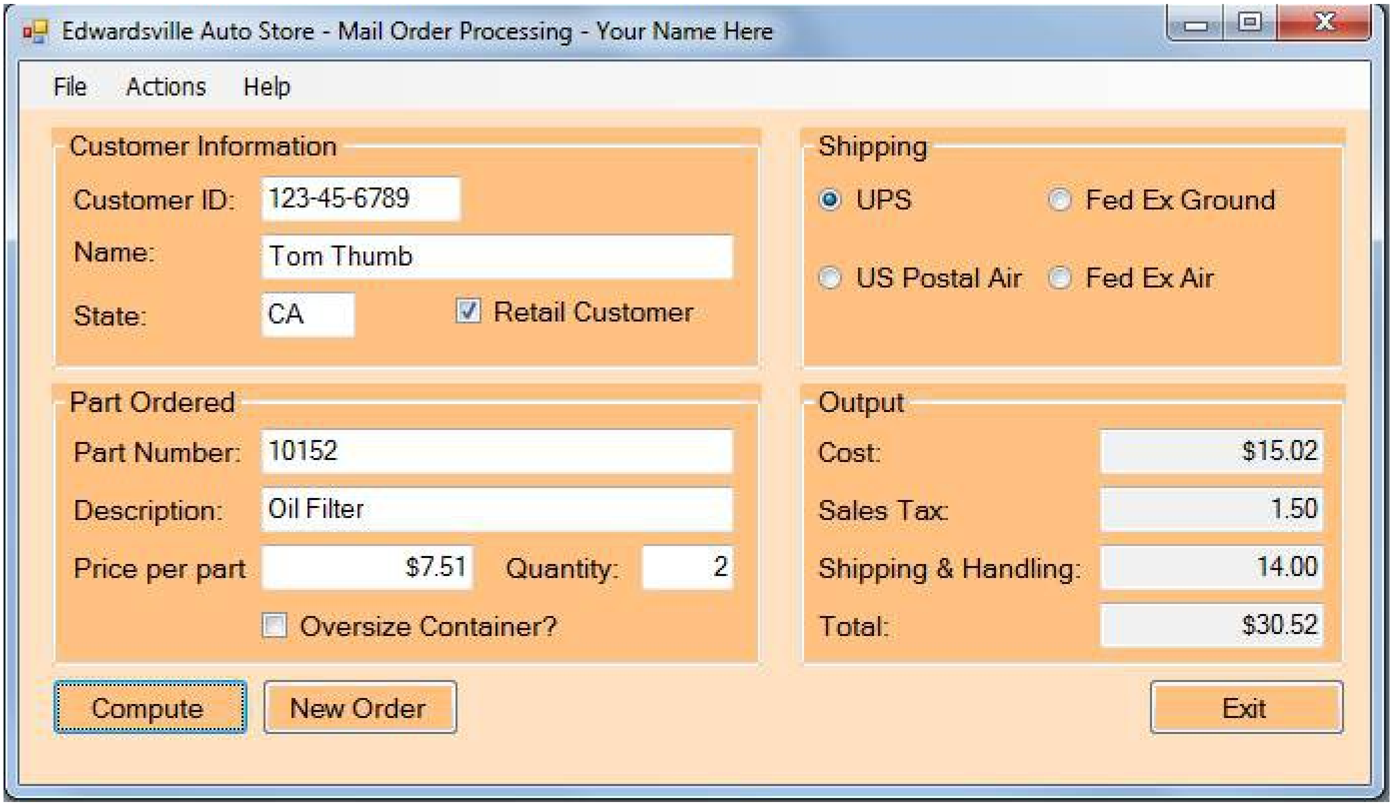
For this assignment, the auto store needs an application that will process mail order sales of repair parts. Parts are sold throughout the United States. The application user will take information from customers by telephone and the application will compute the appropriate output information.

Getting Started.

- Start Visual Studio and create a project as you have for previous projects. Name the project **Lab5-SectionTime-YourLastName-YourFirstName** as you did for previous projects. The form’s **File Name** property should be **Lab5.vb**.
- Save the project as you have for previous projects.

Design Requirements. Develop a form that is similar to the one shown below. You may have minor differences. The "&" symbol in the Shipping &Handling label is displayed by typing two & symbols together, e.g., "&&".

A separate order is processed for each part ordered. Customer information will eventually be retrieved from a database so that only the Customer Identification (Customer ID:), Name, and State Code need be entered on the form. There are two types of customers: Retail and Wholesale. If the customer is a retail customer, this is indicated by checking the Retail CheckBox, while wholesale customers do not have the checkbox checked.



Edwardsville Auto Store - Mail Order Processing - Your Name Here
File
Actions
Help
Compute
New Order
Exit

Customer Information

Customer ID: 123-45-6789

Name: Tom Thumb

State: CA ☑ Retail Customer

Shipping

● UPS ○ Fed Ex Ground

○ US Postal Air ○ Fed Ex Air

Part Ordered

Part Number: 10152

Description: Oil Filter

Price per part $7.51 Quantity: 2

□ Oversize Container?

Output

Cost: $15.02

Sales Tax: 1.50

Shipping & Handling: 14.00

Total: $30.52

Application Use and Special Instructions

Cost. This is simply the price times the quantity rounded to the nearest penny. All quantities are integer values.

Sales Tax. The application user enters customer and part ordered information within the two GroupBox controls as shown in the figure. The firm is located within California. California retail customers (State code = “CA”) are charged a sales tax on purchases – the California tax rate is 10%. Retail customers from New York (state code = “NY”) and Florida (state code = “FL”) are also taxed at a 5% tax rate. Retail customers from all other states do not pay sales tax.

Wholesale customers (indicated by checking not checking the Retail Customer CheckBox control) do not pay sales tax regardless of their state code.

For example, a retail customer placing an order that cost $100.00 that lives in California will be charged $10.00 sales tax. The same order for a customer in New York will be taxed $5.00, while a customer living in Illinois will not pay any sales tax. A wholesale customer will not pay any tax regardless of their state of residence.

<u>Shipping and Handling.</u> All customers must pay for shipping, and there are four shipping methods as indicated by the RadioButton controls on the form. The United Parcel Service (UPS) shipping method is the default method. The cost of shipping is based on the quantity of the shipment and the shipping method selected as indicated in the table given below. The distance for shipment does not matter.

Shipping Charge	**UPS**	**U.S. Postal Air**	**Fed Ex Ground**	**Fed Ex Overnight**
Charge per part	$7.00	$8.50	$9.25	$12.00

There is an extra handling charge of $5.00 per part for an oversize container (the Oversize Container CheckBox is checked).

For example, if a wholesale customer orders Part # 104562, Muffler, 2 each shipped by UPS. Then the charge for shipping is $7.00 times 2 = $14.00. If this part is oversize, then the extra handling charge is $5.00 times 2 = $10.00. If the part is not oversize, then there is no charge for handling. Shipping and handling are combined as a single figure and displayed in the TextBox indicated on the form.

<u>Total.</u> The Total is equal to the **Cost** + **Sales Tax** + **Shipping & Handling**.

Build and Code the Project

Design.

- Use GroupBox and other controls as shown in the figure.
- The form should startup with focus on the customer CustomerIDTextBox control. This control is a MaskedTextBox control with Mask = Social security number.
- Data entered into the TextBox control for the state code should be automatically capitalized.
- The Retail CheckBox control should be checked on startup - most customers are retail.
- The Oversize CheckBox control should NOT be checked on startup.
- The tab order must be top to bottom, left to right within the group box controls.

- Left and right-align data as shown in the figure – the price and quantity price should be right-aligned.
- The output controls in the Output GroupBox control are read-only TextBox controls with TabStop = False, and all output values should be right-aligned.
- You must use **Regions** to organize your code. You decide how many to use and what to name each region.

Programming Events (where the directions require a function to be called, your program must call the appropriate function or you will not receive credit for coding that requirement).

- Add remarks to the program as required.
- You must set OPTION STRICT ON.
- Compute button click event.
  - Your code must properly call a function named **ValidData** to validate the data (see the directions below on data validation rules).
  - Your code must call a function named **SalesTax** to compute the amount of sales tax. You must pass the **Cost** to the function as an argument of the function call. The function must compute the sales tax for customers rounded to the nearest penny, and return the sales tax value to the compute button click event.
  - Your code must call a function named **ShippingHandling** to compute the charge for both shipping and handling of an order. You do not need to pass any arguments to this function because all of the information needed to compute the charges are available from the controls on the form. Return the combined cost of shipping and handling to the compute button click event.
  - Display the computed values formatted as shown in the figure to the nearest penny.
- New Order button click event.
  - Clear all TextBox and MaskedTextBox control contents.
  - Check the Retail Customer CheckBox control.
  - Uncheck the Oversize Container CheckBox control.
  - Reset the Shipping RadioButton controls to the UPS shipping method.
  - Set focus to CustomerIDTextBox control.
- Exit click event.
  - Display a message box asking the application user to confirm to exit the application.
  - Exit the application if the user responds yes.

- Add a menu strip control with menus like those shown in the table here:

&File	&Actions	&Help
E&xit Ctrl-X	&Compute Ctrl-C &New Order Ctrl-N C&olor &Font	&About

  - The menu selections for File-Exit, Actions-Compute, and Actions-New Order should execute the same code procedures for the button controls. **Do NOT duplicate the code.** Rather, use one of the techniques you learned in class to call or execute the code procedures for the buttons.
  - The Help-About menu should use a MessageBox command to display a message box with an information icon that provides the system date and time, your name as the project programmer.
  - The Actions-Color menu should display a Color common dialog box that enables the system user to select a new color. Use the new color to change the BackColor property to the new color for the form or any part of the form that you desire. It should also change the BackColor of the menu strip control
  - The Actions-Font menu should display a Font common dialog box that enables the application user to select a new font. This should cause the font for the form to change. Since you are being tested on your ability to use the control, your code must change the font size for every control on the form to the same size.
- Add a context menu strip control with the following menu options: **Color**, **Font**, and **Exit**. The context menu strip controls should execute same code procedures that you wrote for the menu strip controls. **Do NOT duplicate the code.** Use one of the techniques you learned in class to call or execute the code procedures for the existing menu strip or button control sub procedures.

Data Validation.

- You must code a function named **ValidData** that enforces the following business rules.
  - Rule #1: Customer ID cannot be missing and cannot contain any blank spaces.
  - Rule #2: Customer Name cannot be missing.

  - Rule #3: State code must always be entered as exactly two characters (hint use the Trim and Length methods—you can look these methods up in the MSDN help).
  - Rule #4: Part Number cannot be missing.
  - Rule #5: Description cannot be missing.
  - Rule #6: Price must be a number that is greater than zero.
  - Rule #7: Quantity must be a number that is greater than zero.
- If a business rule is violated, you must display a message box with an appropriate error message, title, OK button, and error icon. You must set the focus to the control that has the error and highlight any invalid data that the control may contain.

Other Requirements. Ensure that you follow all standards for programming that you learned and used in earlier computer programming assignments including the use of comments, proper spacing of code, proper spelling, proper indentation of code, etc., as listed in the assessment section given below.

Test the Lab. Use the assessment guidelines provided below to test your lab work. This is the same set of assessment guidelines that will be used to grade your work.

- Use the test data given here for initial program testing. You will need to use additional test sets to thoroughly test the application.
- Ensure you test each Try-Catch block by entering erroneous data in addition to the test data shown below.

What to Do When You're Finished, How To Save and Submit the Lab.
Ok, you've finished the lab. **Do NOT use the File menu, Save As option**. Instead, follow these steps.

- First, CLOSE Visual Studio—you cannot copy the project to another location if it is open.
- Locate the folder that contains the project. It should be located in the **My Documents** folder where you first saved the project.
- Copy the entire folder to a flash drive.
- Bring the flash drive to the University to either a computer lab or to the computer classroom. Copy the entire folder to drive Y: to the submission folder for your class. You can copy the folder to drive Y: at the beginning of the class period when the project is due.

Assessment of Project. Your work will be evaluated with point deductions as explained below. An assessment (test) plan is provided that lists the different types of errors.

- Major errors – multiple points are deducted for major errors.
- Minor errors – one point is deducted for EACH minor error.

Visual Basic Project Assessment Plan – Lab 5 (50 points possible).

Before Startup – Form Design.

- Submitted late – see the course syllabus for a description of the late penalty.
- Submitted on time but some of the files necessary to run the project are missing – you must resubmit the project, see your instructor if you need assistance submitting the project – your resubmission will be considered a late submission.
- Project should be named correctly: **Lab5-SectionTime-YourLastName-YourFirstName**.
- The form's File Name property is set to a new name (Lab5.vb, NOT Form1.vb).
- Title bar of the form has the required information as shown in the figure.
- Form has a good appearance with all controls aligned and sized appropriately so that the appearance is pleasing and professional. Errors include misspelled words, having too much or not enough gray space around controls, controls not aligned, TextBox controls too small to display all required information.
- Each GroupBox control displays the appropriate Text value.
- All output TextBox controls are read-only with TabStop = False.
- The Shipping GroupBox has UPS RadioButton checked on startup.
- The Retail Customer CheckBox control is checked on startup
- The Oversize CheckBox control is unchecked on startup.
- Program code has the required remarks that identify the program, programmer name, and date programmed.
- Each sub procedure has remarks statements to identify what the sub procedure does.
- All variables and controls are named properly following the naming convention taught in the notes and in class.
- Main Menu is as required (-8 if missing; -1 for each minor error).
- Main Menu or Context Menu properly calls existing button sub procedures (-4 if code is duplicated).
- Correctly used Regions to organize the code.

Examine the Compute Button control's click event.

- Failed to use a Try-Catch block to catch unexpected errors (-2).
- Properly calls ValidData function (-4 if not called properly even if the function is coded correctly).
- Error in parsing either price or quantity (-1 each).
- Error in computing total (-2).
- Properly calls SalesTax function (-2).
- Properly calls ShippingHandling function (-2).
- Error in computing cost (-2); did not use rounding (-1).
- Did not display computed output values formatted as required (-2).

Examine the ValidData function (maximum -8).

- Not coded properly or function is missing (-8).
- Properly enforces the Business Rules (-1 for each business rule not enforced properly).
    - Customer ID, name; state code; part number; description; price; quantity.
- Failed to display proper message box message; failed to set focus properly; failed to highlight any data in a control.
- Fails to return either True or False depending on whether data is or is not valid (-2).

Examine the SalesTax function (-8).

- Not coded properly or function is missing (-8).
- Failed to declare constants for the various tax rates (CA = 107%; NY and FL = 5%).
- Function is coded incorrectly (-2 for computing tax wrong for CA; -2 for computing tax wrong for NY/FL; -2 for charging tax when tax should not be charged (wholesale or states other than CA/NY/FL).
- Fails to use rounding or data conversion properly (-1).
- Fails to return a value (-2).

Examine the ShippingHandling function (-8).

- Not coded properly or function is missing (-8).
- Failed to declare constants for the shipping and handling rates.
- Function is coded incorrectly (-4 Fails to compute shipping charge properly; -2 fails to compute handling charge properly).
- Fails to use rounding or data conversion properly (-1).
- Fails to return a value (-2).

Startup.

- Form starts up centered on the screen. Form is an appropriate size.
- Tab order is correct.

Next Order Button click event (3 points).

- Clears all TextBox and MaskedTextBox controls.
- Check the RetailCustomerCheckBox control.
- Uncheck the OversizeCheckBox control.
- Reset the UPSRadioButton control to True.
- Properly sets focus to CustomerIDTextBox.

Exit Button Click Event.

- Displays message box as specified with both Yes and No buttons.
- Defaults application user response to the No button.
- Exits only if the Yes button is clicked.

Menu Strip control click events (3 points).

- Works, but the code is duplicated (-2).
- The File-Exit, Actions-Compute, and Actions-Next Order work correctly (-1 for each one that does not work).
- Help About menu displays message box.
- Actions-Font menu displays Font common dialog box and properly changes a Font on the form.
- Actions-Color menu displays Color common dialog box and changes the ForeColor of a control on the form.

Context Menu Strip control click events (3 points)

- Displays correctly when right-clicking appropriate area on the form (-2 if does not display-check the ContextMenu property of the form is set; -3 if the Context Menu is missing).
- Has the required Color, Font, or Exit option.
- Color, Font, or Exit options properly call the existing coding procedures without duplicating the code.

Other Errors.

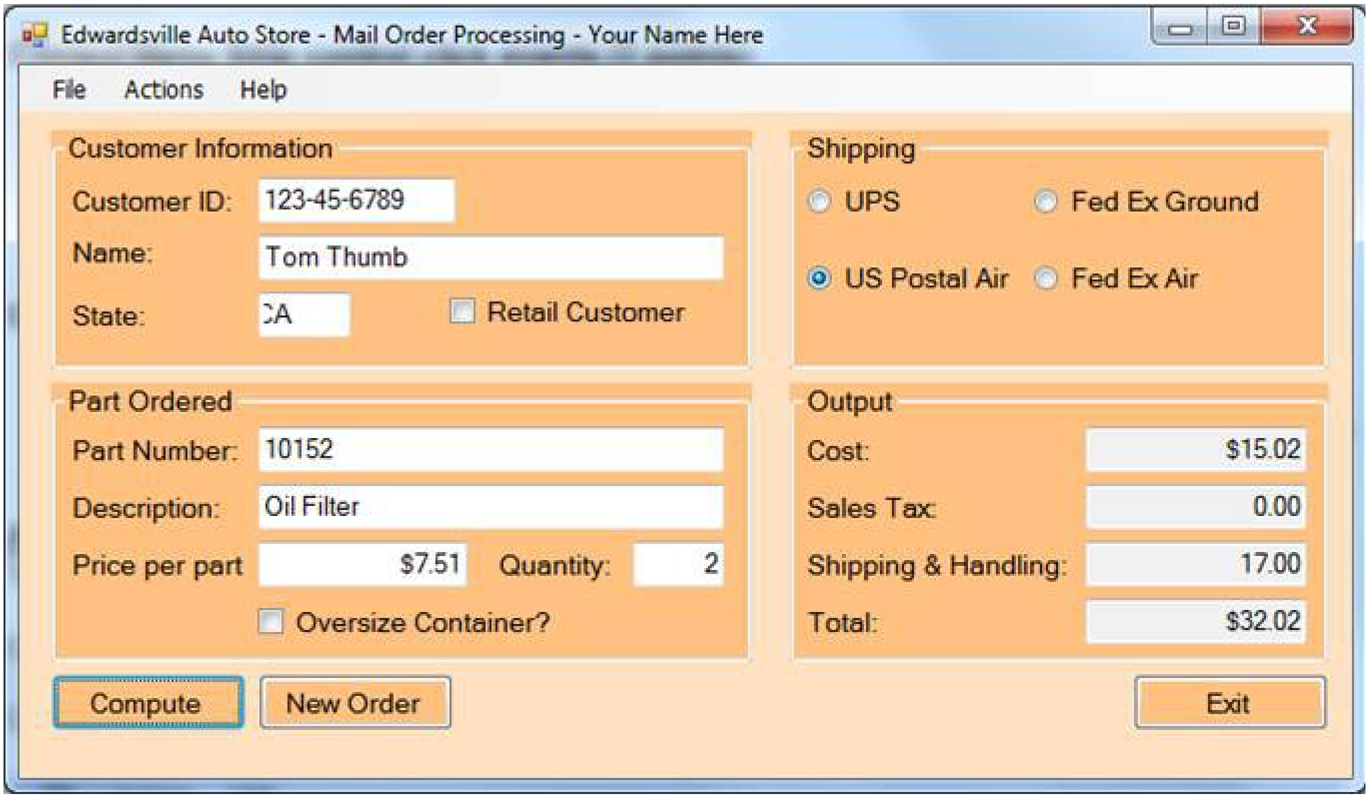
- During program execution, your computer program should compile and execute without generating any error messages – if the program will not compile or generates error messages during execution, you will lose up to 45 points depending on the severity of the error.

<u>Additional Test Data</u>

You can use the following Test data to test your program.

Case #1. See the form at the beginning of this lab assignment.

Case #2. State = CA; wholesale; not oversize; shipping = US Postal Air


Edwardsville Auto Store - Mail Order Processing - Your Name Here
File
Actions
Help
Compute
New Order
Exit

Customer Information

Customer ID: 123-45-6789

Name: Tom Thumb

State: :A

□ Retail Customer

Shipping

○ UPS ○ Fed Ex Ground

● US Postal Air ○ Fed Ex Air

Part Ordered

Part Number: 10152

Description: Oil Filter

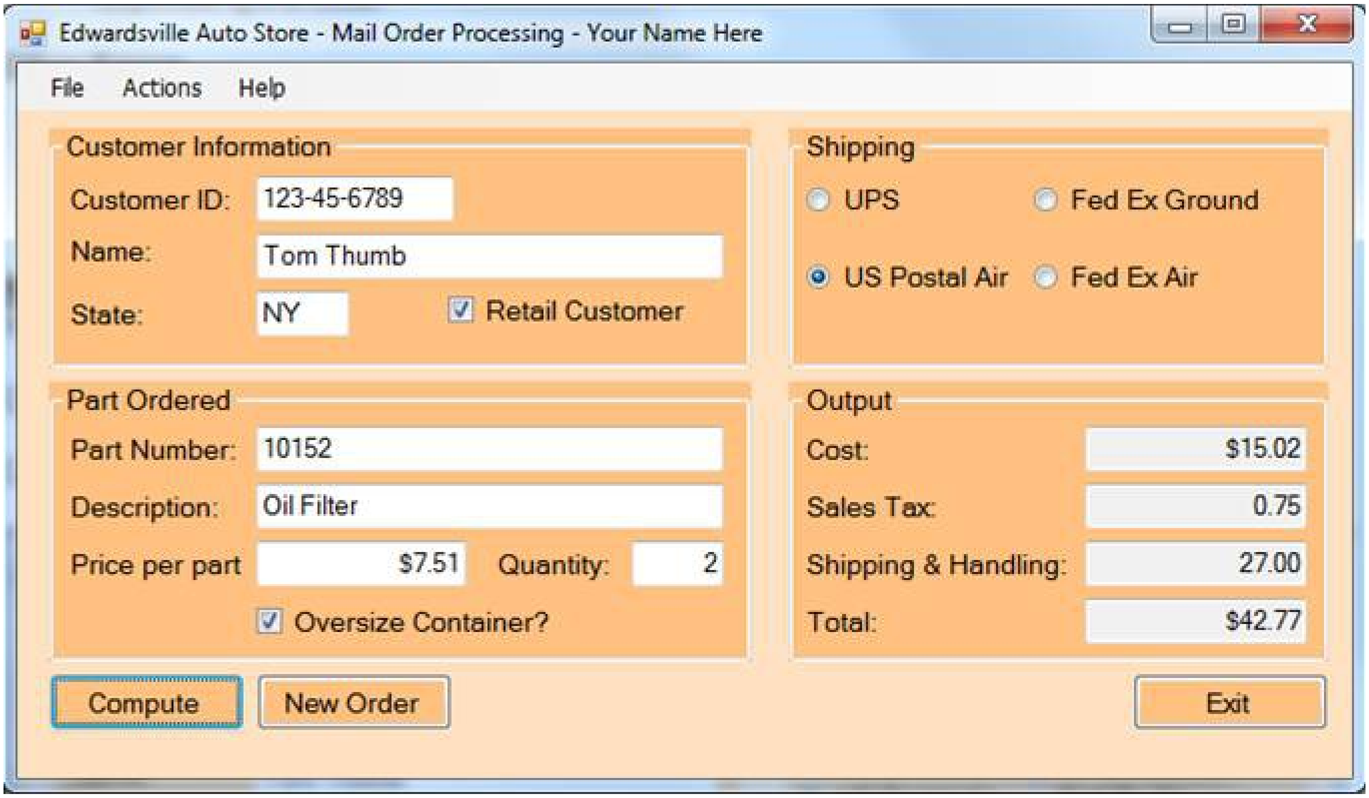
Price per part: $7.51 Quantity: 2

□ Oversize Container?

Output

Cost:	$15.02
Sales Tax:	0.00
Shipping & Handling:	17.00
Total:	$32.02

Case #3. State = NY; retail; oversize; shipping = US Postal Air


Edwardsville Auto Store - Mail Order Processing - Your Name Here
File
Actions
Help
Compute
New Order
Exit

Customer Information

Customer ID: 123-45-6789

Name: Tom Thumb

State: NY

☑ Retail Customer

Shipping

○ UPS ○ Fed Ex Ground

● US Postal Air ○ Fed Ex Air

Part Ordered

Part Number: 10152

Description: Oil Filter

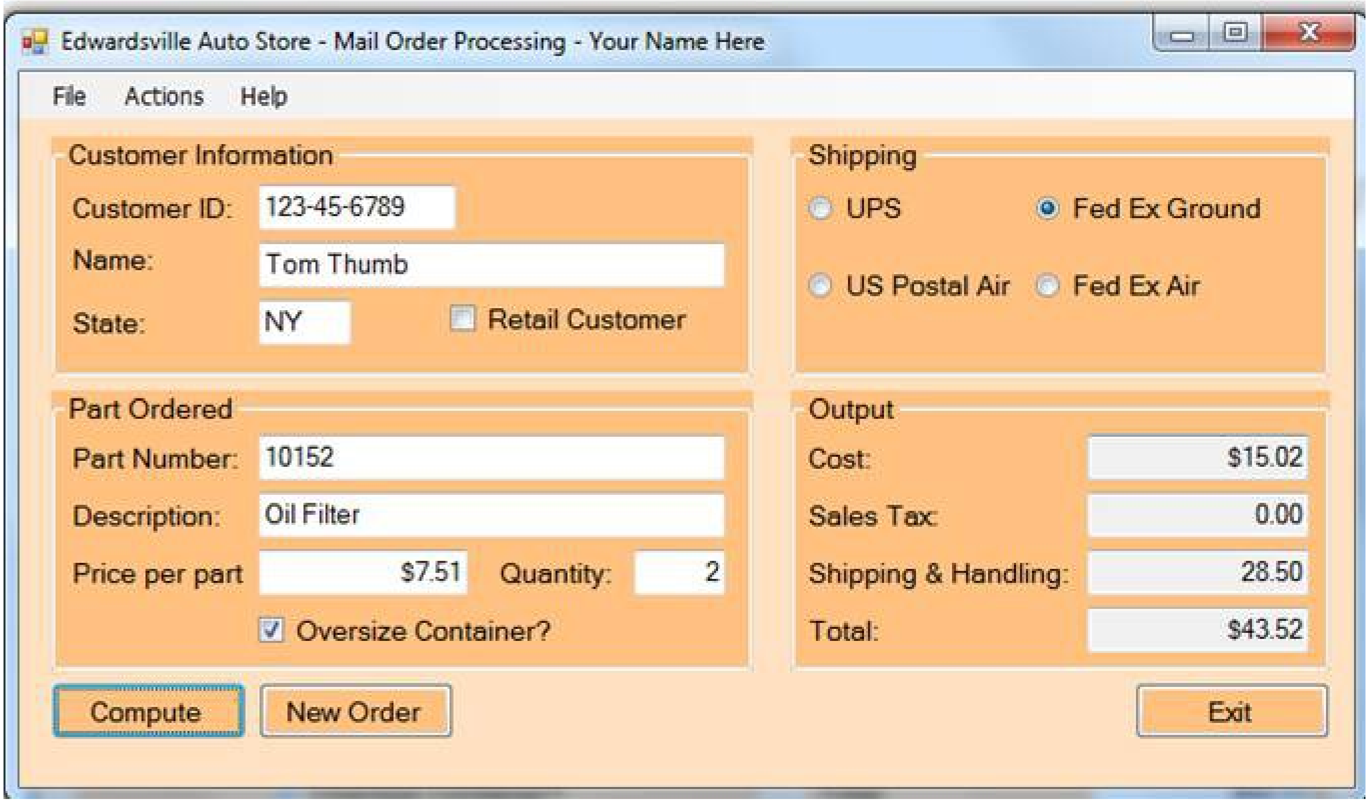
Price per part: $7.51 Quantity: 2

☑ Oversize Container?

Output

Cost:	$15.02
Sales Tax:	0.75
Shipping & Handling:	27.00
Total:	$42.77

Case #4. State = NY; wholesale; oversize; shipping = Fed Ex Ground.



Edwardsville Auto Store - Mail Order Processing - Your Name Here

Customer Information

Customer ID: 123-45-6789

Name: Tom Thumb

State: NY □ Retail Customer

Part Ordered

Part Number: 10152

Description: Oil Filter

Price per part $7.51 Quantity: 2

☑ Oversize Container?

Shipping

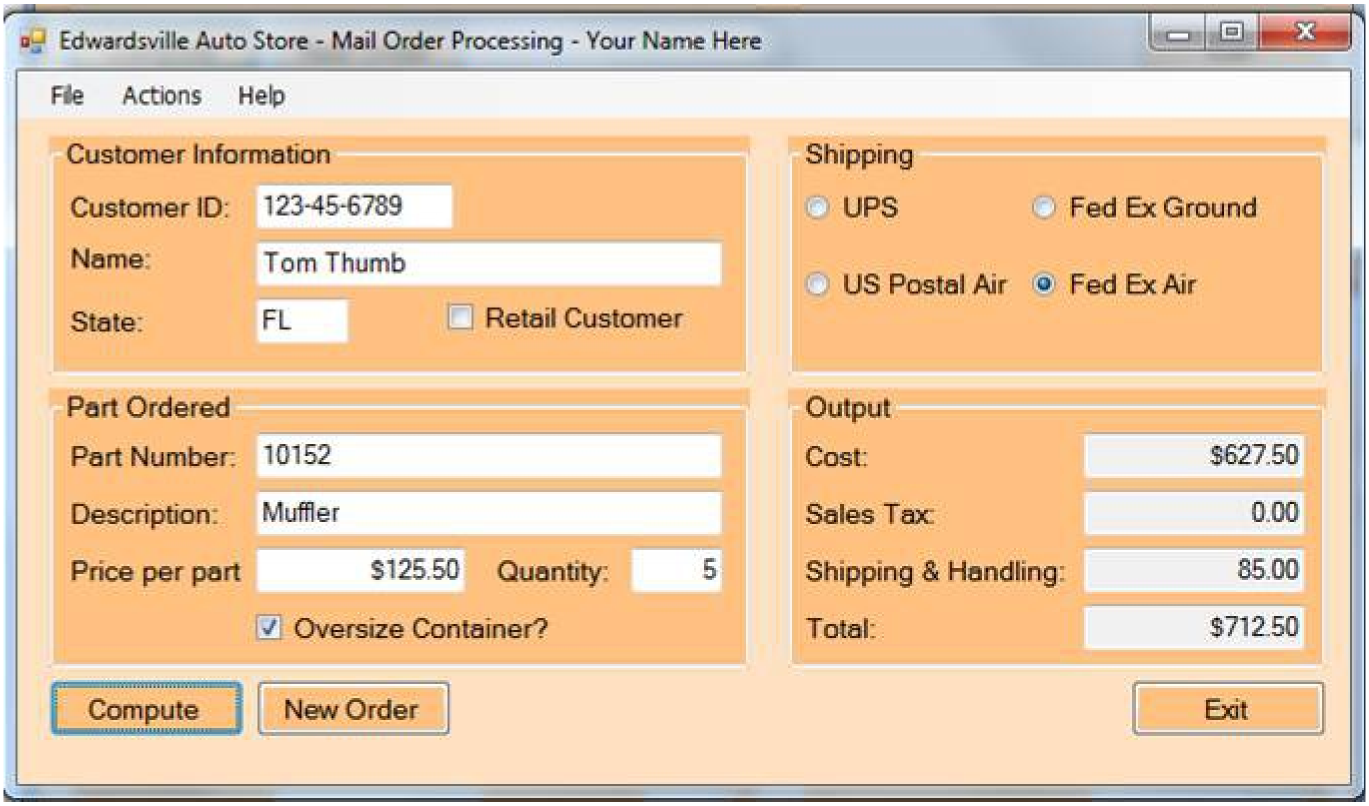
○ UPS ● Fed Ex Ground

○ US Postal Air ○ Fed Ex Air

Output

Cost:	$15.02
Sales Tax:	0.00
Shipping & Handling:	28.50
Total:	$43.52

Case #5. State = FL; wholesale; oversize; shipping = Fed Ex Air. Different part/quantity.


Edwardsville Auto Store - Mail Order Processing - Your Name Here
File
Actions
Help
Compute
New Order
Exit

Customer Information

Customer ID: 123-45-6789

Name: Tom Thumb

State: FL ☐ Retail Customer

Shipping

○ UPS ○ Fed Ex Ground

○ US Postal Air ● Fed Ex Air

Part Ordered

Part Number: 10152

Description: Muffler

Price per part $125.50 Quantity: 5

☑ Oversize Container?

Output

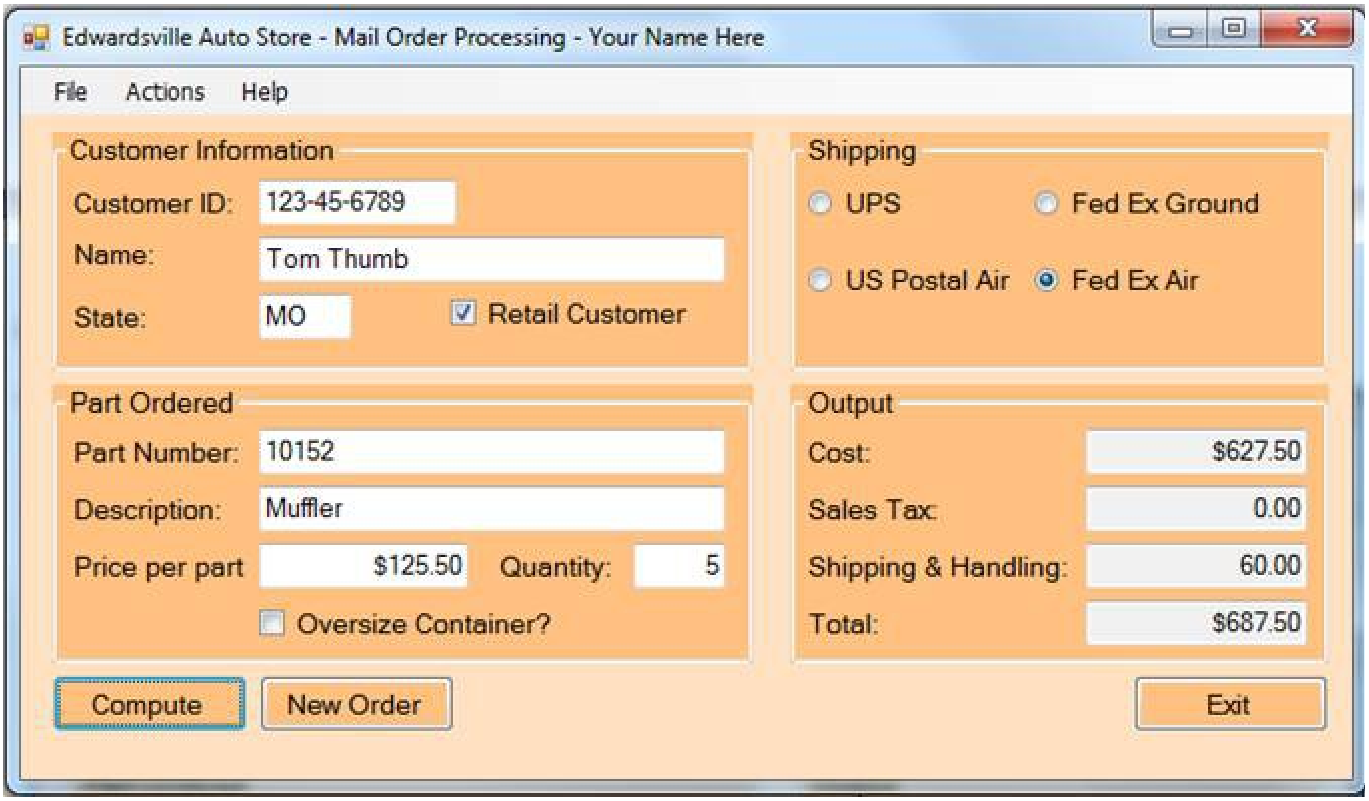
Cost: $627.50

Sales Tax: 0.00

Shipping & Handling: 85.00

Total: $712.50

Case #6. State = MO; retail; not oversize; shipping = Fed Ex Air.


Edwardsville Auto Store - Mail Order Processing - Your Name Here
File
Actions
Help
Compute
New Order
Exit

Customer Information

Customer ID: 123-45-6789

Name: Tom Thumb

State: MO ☑ Retail Customer

Shipping

○ UPS ○ Fed Ex Ground

○ US Postal Air ● Fed Ex Air

Part Ordered

Part Number: 10152

Description: Muffler

Price per part $125.50 Quantity: 5

☐ Oversize Container?

Output

Cost: $627.50

Sales Tax: 0.00

Shipping & Handling: 60.00

Total: $687.50

You should use additional test data to ensure that the data validation works properly. You should also test for data validation with case situations where the:

- Customer identification missing.
- Customer name missing.
- State not two characters
- Part number missing.
- Description missing.
- Price not a number or less than zero.
- Quantity not a number or less than zero.

End of Lab