Project 6: Cable Accounts

Assigned: October 5, 2010
Due: Sunday, October 9, 2010 by 11:59 PM

Deliverables:

The following project files must be uploaded to Web-CAT by the due date and time specified above (see the Lab Guidelines for information on submitting project files). If you do not submit your project by Web-CAT or email before 11:59 PM, then you will not be given credit for that submission without an AU acceptable excuse (no exceptions). **Make sure that you have at least 1 submission on the graded Web-CAT submission by 7 PM before the deadline in case you have any issues with Web-CAT and are unable to submit via e-mail.**

- CableAccount.java
- AccountCalculatorPanel.java
- AccountCalculator.java

Directions:

CableAccount.java

- **Requirements**: The cable account class will represent a cable service account for a residence with one or more owners. Each cable account has a type with pricing as shown on the right.

Type	Cost
No Cable	$ 0.00
Basic	$50.00
Extended	$20.00
Premium	$15.50
Premium Plus	$20.70
"Best Deal"	$30.50

When calculating the cost for a service, the price of that service is summed with the price of all services below it. For example, if a customer had Premium cable, then their monthly charge would be 15.5 + 20 + 50 = $85.50.
In order to pick up certain channels, a TV must be connected to a box. The cost of each box depends on how many boxes the user already has. The first box is $10, and the cost of each box after that is 10% off the cost of the previous box. So if the user has 4 boxes, then the cost would be 10 + 9 + 8.1 + 7.29 = $34.39. The account is always charged for at least one cable box (so, even if the user does not own a cable box then they are charged $10.

- **Design**: There should be public int constants representing type: `NO_CABLE, BASIC, EXTENDED, PREMIUM, PREMIUM_PLUS, BEST_DEAL`. Your CableAccount class must contain a constructor that accepts a string as parameter representing the first owner's name. Include the following list of methods as well.

    - `addOwner`: adds an owner's name (type String) to the list of owners if the name does not consist of all white space. The name should also not be added if it is already present in the list of names. Each account can contain an unlimited list of owners.

- `deleteOwner`: deletes the owner from the list represented by the String parameter unless there is only one owner left in the list (each account must have at least 1 owner).

- `setService`: sets the service type (an int parameter) if it is one of the valid types.

- `setCableBoxes`: sets the number of cable boxes included in the service (an int parameter) unless the number is less than 0 or greater than 15.

- `getServiceString`: returns a String representation of the customer's service type (no parameters). String representations are No Cable, Basic, Extended, Premium Plus, and Best Deal.

- `totalCost`: returns the total cost of the service including the cost for the service and cable boxes present (see the requirements section for information on cost.

- `costPerOwner`: returns the total cost of the service divided by the number of owners on the account.

- `toString`: Returns a String containing all user's names, the total cost, and the cost per user. Label all values and make sure that your output is

- **Code & Test**: Use a loop to calculate the cost of the cable boxes. You'll probably need a variable for the current box price as well as a running sum. Remember to include constants to prevent unnamed literals (costs, for example). Test multiple values for the number of boxes and test the price of all services. **Use Checkstyle as you code, no just when you need to submit; it can help to prevent logic errors in your code that will produce unexpected output.**

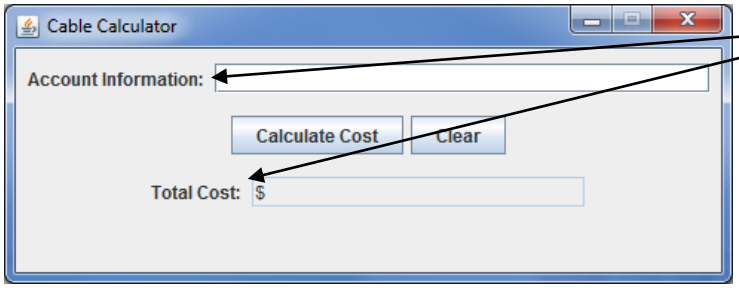
Directions:

AccountCalculatorPanel.java & AccountCalculator.java

- **Requirements**: The GUI for this project allows a user to enter a customer's name in the format last name and first name as well as the number of cable boxes they want. The price should then be calculated with the assumption that the user has the basic cable service.

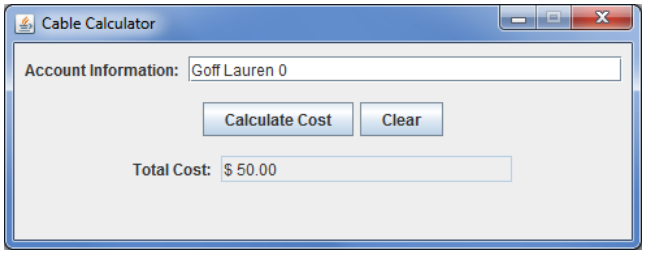
- **Design**: You will be required to add functionality for the two buttons to calculate cost and to clear the display. The cost must always show 2 decimal places. You must also change the two labels for the text fields. All changes should be made to the AccountCalculatorPanel class.

The GUI should look like the screenshot below when it is first run.

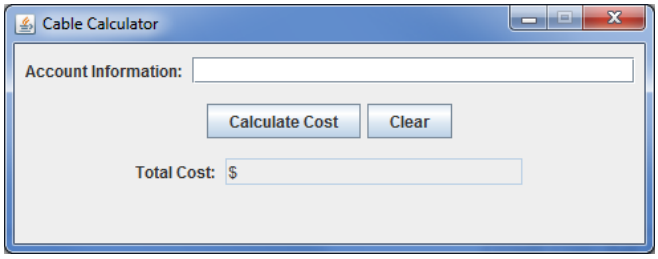


Change the text for the labels to the left of the text fields in the AccountCalculatorPanel class.

Below is an example after information is entered and the Calculate Cost button is pressed.



The GUI should look like the screenshot below when the clear button is pressed.



- **Code & Test**: The ProcessClearListener class determines what will happen when one of the two buttons is pressed. You may want to add a support method to your class that creates a cable account given the information and calculates the cost. Javadoc comments will not be required for any support methods that you add.