Project 2: APPs

Toan Truong

213943246

EECS 3421

December 4, 2017

To get started, connect to the yorku computers and import the yrb-create file from Project 1 and my submission for Project 2.

In terminal, type the following:

- source~db2leduc/cshrc.runtime
- db2 connect to c3241a
- db2 -tf yrb-create // if it is already made then you can skip this
- javac CustTotal.java
- java CustTotal [enter customer ID here]

Functionalities of the program

- If you run CustTotal without putting a customer id, the program will tell you how you can use it:

```
red 246 % java CustTotal

Usage: java CustTotal cust#
```

Once the correct parameters are met, the program will check if that id is valid or not. If it is incorrect (as in there is no customer with that ID in the database) then it will return:

```
red 247 % java CustTotal 500
There is no customer #500 in the database.
Please enter a valid ID.
```

From here it will prompt you to enter a valid ID nonstop. Once a valid ID is found, it will return the customer information:

```
red 247 % java CustTotal 500
There is no customer #500 in the database.
Please enter a valid ID.
1

(#1 - Toan Truong - Mississauga)

Would you like to update your information? [Y/N]
```

Here it returns (customer ID – customer name – customer's city) and prompts the customer if they would like to update their information. Before moving forward,

- Inputting anything but NAME or CITY

```
Would you like to update your information? [Y/N] Y
Y
Current Information:
NAME: toan truong
CITY: Toronto
Enter the item that you would like to update:
a
**Please enter NAME or CITY**
g
**Please enter NAME or CITY**
sjsj
**Please enter NAME or CITY**
```

The program will display the customer's data, listing the customer name and city. Those two are the only parameters that can be changed. I did not include the option of changing the CID as that usually is not something that the public can change.

- Entering NAME and a new name to change it:

```
**Please enter NAME or CITY**
sjsj
**Please enter NAME or CITY**
NAME
Please enter the new NAME:
TOAN TRUONG

The changes have been made.

(#1 - TOAN TRUONG - Toronto)

Would you like to do another update? [Y/N]
```

You can see that the current information and the newly returned customer information is different.

- Changing City parameter

```
Current Information:
NAME: toan truong
CITY: Mississauga
Enter the item that you would like to update:
CITY
Please enter the new CITY:
Toronto

The changes have been made.

(#1 - toan truong - Toronto)

Would you like to do another update? [Y/N]
```

Entering in CITY allows you to make changes to the city information of the customer. The program will continue to loop through this, allowing them to make changes as long as they keep entering Y.

- Entering 'N' when asked if you would like to do an update

```
Would you like to update your information? [Y/N] N
N


Categories that we offer:
~~~~~~~~~~~~~~~~~~~~~~~~~
children
cooking
drama
guide
history
horror
humor
mystery
phil
romance
science
travel

Please enter the category you would like to explore:
```

The program will return a list of categories that it has to offer.

- Entering a category that matches the list of categories

```
Please enter the category you would like to explore:
travel

List of books in travel:
~~~~~~~~~~~~~~~~~~~~~~~~
Richmond Underground
Suzy does Yorktown
Lynchburg's Great!
Alpha Centauri
Where's Canada?
Ou est Ce Canada?
Montreal est dans Quebec
Darmstadt
Parthenon
Live in Charlottesville?!
Not That London!
Tropical Blacksburg
Montreal pour les Idiots
Williamsburg Hot Spots

Please enter the TITLE of the book you are looking for:
```

The program will return the list of books of that respective category.

- Entering an invalid category and book title

```
Categories that we offer:
~~~~~~~~~~~~~~~~~~~~~~~~
children
cooking
drama
guide
history
horror
humor
mystery
phil
romance
science
travel

Please enter the category you would like to explore:
afjaf^H^H

List of books in afj:f
~~~~~~~~~~~~~~~~~~~~~~~~

Please enter the TITLE of the book you are looking for:
asdf

*** Invalid parameters ***


Categories that we offer:
~~~~~~~~~~~~~~~~~~~~~~~~
children
cooking
```

It will accept the category submission and look at the database for books in that category. In the instance above, I entered a random set of characters and the program looked for that category. Then it searched for books with the title 'afj;f' in that category which was not found. This returns the invalid parameters statement and then returns the list of categories again. This will repeat until there are valid inputs.

- Inputting a valid book title

```
Please enter the TITLE of the book you are looking for:
Parthenon


Book information:
----------------
Title: Parthenon
Year: 2000
Language: Greek
Weight: 313

The minimum price for Parthenon from the travel category is $29.95

Enter the number of book(s) you would like to buy:
```

A valid book title will bring up it's book information along with the category it is from and the minimum price it can be found at with respect to the person's club. It then asks for the number of books that the customer would like to purchase.

- Entering an invalid amount (a negative value)

```
Enter the number of book(s) you would like to buy:
-1
Please enter a non-negative number
```

- Entering an invalid amount (a positive value)

```
The minimum price for Parthenon from the travel category is $29.95

Enter the number of book(s) you would like to buy:
-1
Please enter a non-negative number
2

Your total cost is $59.90

Would you like to make this purchase? [Y/N]
```

A positive value will calculate the total cost of the purchase and ask if the user would like to make the purchase.

- Rejecting the purchase (entering 'N')

```
Would you like to make this purchase? [Y/N]
N

Have a great day.
```

- Accepting the purchase (entering 'Y')

```
Enter the number of book(s) you would like to buy:
1000

Your total cost is $29950.00

Would you like to make this purchase? [Y/N]
Y
Thank you for your purchase. The transcation has been recorded.
```

Here I opted to purchase 1000 copies of Parthenon. The program ends once an answer has been inputted by the customer.

Running following SQL query:

SELECT *

FROM yrb_purchase P

WHERE P.cid = '1';

to check the list of purchases under yrb_purchase made by me (with a CID of 1) results in the following:

```
CID    CLUB           TITLE                    YEAR  WHEN                          QNTY
------ -------------- ------------------------ ----- ---------------------------- ------
     1 Basic          Darmstadt                 2000 2017-12-04-15.23.39.000000      99
     1 Basic          Parthenon                 2000 2017-12-04-13.50.10.000000       1
     1 Basic          Parthenon                 2000 2017-12-04-14.27.17.000000       2
     1 Basic          Parthenon                 2000 2017-12-04-14.41.26.000000       4
     1 Basic          Parthenon                 2000 2017-12-04-14.58.43.000000      99
     1 Basic          Parthenon                 2000 2017-12-04-15.38.34.000000      55
     1 Basic          Parthenon                 2000 2017-12-04-17.15.01.000000    1000
```

You can see the last entry I made a purchase of 1000 copies of Parthenon.