CPT 167—Intro to Programming Logic

Final Project

PROGRAM DESCRIPTION

The main attraction of Summerville SeaWorld™ is no longer simply a proposition anymore: the Porpoise Pond is a great hit with the local water-fun crowd! Your planning was critical to getting the project off the ground, so to speak, and lots of grinning kiddies all wet. Congratulations! But only now has the boss discovered a missing link in the business plan: a way to analyze the receipts and categorize exactly how well the attraction is doing. Time for one more great JavaScript program. This is sure to cement your reputation, again so to speak☺, and ensure you will be in high demand by businesses all over the Low Country.

The company ticket sales information is all sent to a data file to record each day's activity. But since the data arrives electronically from approximately 7 – 10 sales booths, there is no way to record anything but the basic sales data. That's the data we would like to process and accumulate overall totals for the entire park. Unfortunately, since JavaScript lacks the capability to process external data files, we will have to process the data by manually entering the recorded numbers.

RECORD DESCRIPTION

Each sales entry in the listing contains data on one sale to one customer. Here is a description of each of the data recorded in the listing:

Date of operation, for example, June 21, 2014.

Then, the individual sales transactions are listed in columns like this:

TICKET TYPE	QUANTITY	CHARGE
A	2	23.00
C	8	0.00
J	6	**33.75**
J	6	45.00
C	2	0.00
J	9	67.50
A	5	57.50

(NOTE: the third cost is wrong, a fact that your program will need to be on the lookout for!) Your program should first ask for the current date (you can just input it as a String), and then prompt for each line of data one at a time. Quit when the user has entered all of the lines of data. Make sure you give them a simple value to enter to stop the program.

SPECIFIC DIRECTIONS

Start the entire report with the name of the company (of course), and the current date. Then, as each set of data is entered, that is one individual sales transaction, you need to display a short summary that includes appropriate headers and shows the following information: a line number; the type of ticket spelled out, for example Adult; the ticket quantity; the purchase price; and the difference, if any, between the actual cost and what the user should have been charged. (If the ticket is a Child ticket, leave the actual charge and delta columns blank: there is no use displaying any of that for free tickets.) The line number will merely be a counter that you increment for each sales item entered.

The reason for the final column, the delta, is that we have had some malfunctions in the ticket-selling equipment—at least, that's what those minimum-wage ticket operators are saying—and we think that some tickets have been "underpriced". So, after you display the charged amount, the dollar value that was in the listing, compute the actual amount that should have been charged. Then, subtract the computed amount from the actual charge and display the difference. (To remind you, adult tickets are $11.50 and junior tickets are $7.50.) As an example, if the transactions were listed exactly as shown in the table above, the first three lines of output would look this way. After the first set of data was entered:

```
#     TYPE      QTY     ACTUAL CHG       DELTA
1     Adult     2       23.00             0.00
```

Once the second set of data was entered:

```
#     TYPE      QTY     ACTUAL CHG       DELTA
2     Child     8
```

And for the third set:

```
#     TYPE      QTY     ACTUAL CHG       DELTA
3     Junior    6       33.75           -11.25
```

The last output reflects that the 6 Junior tickets were not charged correctly. The 6 tickets should have cost 6 * 7.5 = 45.00. Since they were actually purchased for 33.75, that means the park lost $11.25 on that sale. The park management wants to know that sort of thing! Note that the line counter is accurately counting the number of sales, which will allow you to compute some statistics later on. Also, note that all dollar items are displayed in two decimal places, and that there are no entries for the Child tickets after the quantity column.

When you're finished processing individual sales entries, clear the screen and display a final summary containing the following information (see a sample report at the end of these instructions):

- Number of records included in the listing
- Totals for each type of ticket: for each type display the total number of tickets of that type and the total dollars received for that type. (So, this means six separate totals, two for each ticket type.)
- Total dollar sales overall.

Separate functions required. The following functions are mandatory for this program's design and implementation:

main
Main driver for the program.

getTicketType
Just like the one in the last program. Present a menu of the three ticket types. Input the users choice and make sure it is a valid choice. If it is invalid, then ask for another choice. Keep asking until the user enters a valid ticket type. Return to main the users choice.

getQuantity
Input and return the quantity of tickets sold.

getCharge
Input and return the charge for this set of tickets.

findTicketName
Accepts the ticket type code (A, J, or C) and returns the name associated with that letter so you can display it.

displayOneOrder
Accepts the five values to be displayed for one ticket and displays their summary line (including the headers).

displayFinalReport
Format and display the final totals.

Zip your project directory together. The name of the project you create in Eclipse should be this: ***name* Final Project**, where *name* is your last name. And as always, when you add the HTML file, remember to give it some sort of meaningful name for this program.

TURN IN

Your data dictionary, a complete flowchart, and the zipped project directory. Submit your exam files using the Final Project dropbox. And don't miss the deadline: there is no such thing as late submission for any exam programs or projects!

Important: this is an exam. You are not to discuss any aspect of this exam with anyone but your instructor.

[Sample final report for the data listed on page 1]

```
SUMMERVILLE SEA WORLD

Summary report for June 21, 20124

Number of records:  7

SALES TOTALS
---------------------

TICKET TYPE        TOTAL TICKETS         TOTAL REVENUE
--------------------   ------------------------   -------------------------
Adult                    7                  80.50
Junior                  21                 146.25
Child                   10                   0

TOTAL REVENUE:  $226.75
```