CE Symbol Table CODE

MISSING 10 Possible Points

| 10/31/2022

Attempt 1 VIN PROGRESS
Next Up: Submit Assignment



Unlimited Attempts Allowed

∨ Details

Searching

CE: Symbol Table CODE



Learning Objectives

- Create a symbol table and populate it with data from a CSV file.
- Use methods from class ST to obtain specified information.



Overview

This CE consists of two parts.

In part 1, you will compare different symbol table implementations.

In part 2, you will read stock data from a CSV file, enter it in a symbol table, and use existing methods to access specified information.

(<u>https://slcc.instructure.com</u>/courses/817632/modules/items

<u>/18753050</u>)

Submit Assignment

Next >

(<u>https://slcc.instructure.com</u>/courses/817632/modules/items

<u>/18753054</u>)

Part 1

In this first part of the CE, you will write a program that reads stock data from a CSV file, enters the data in a symbol table, and dynamically determines the number of elements in the symbol table.

It starts with a header line followed by Amazon stock data for each of the trading days between May 14th, 1997 until July 12th, 2020.

- Create a package called ceStock.
 Add a class called StocksApp that includes the main method.
 Add a resource folder which includes AMZN.csv.
- In class StocksApp, add a private method called readCSV.
 It should return a symbol table of type ST<Date, Double> and it should have the following three parameters:
 - String fileName, which is the relative path to the CSV file

Hara are avamples of how to avante a Date

- int keyField, which is the index of the column in the CSV file that includes the keyvalue
- int valField, which is the index of the column in the CSV file that includes the value that will be associated with the key.

The method should create a symbol table and fill it with key-value pairs from the data provided in the CSV file. The method should also assume that the CSV file includes a header that needs to be ignored.

Feel free to base your implementation on the code from class <u>LookupCSV</u> (https://algs4.cs.princeton.edu/35applications/LookupCSV.java.html) which provides similar functionality.

You will need to convert the date in the key column (keyField) from a String to a Date.

 ⟨Previous
 Next⟩

 (https://slcc.instructure.com
 Submit Assignment

 /courses/817632/modules/items
 /courses/817632/modules/items

 /18753050)
 /18753054)

ParseException occurs, do the following: catch it, display the line from the CSV file that caused the problem, and print the stack trace. Continue to read in the rest of the CSV file.

At this point, the implementation of the method readCSV should be finished.
 Call it from the main method to create a symbol table that associates the date with the closing price of the stock. The closing price is listed in the column with the header "Close".

Dynamically determine the number of elements in the symbol table and print it. Here is something to <u>compare</u>

Part 2

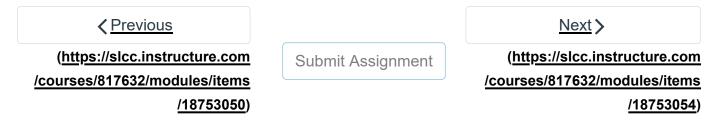
Important:

Avoid the temptation to hard-code. If I change the data in AMZN.csv (e.g. add, remove, modify records) the output needs to change accordingly.

- A. Print the oldest available date and the associated closing price.
- B. Print the newest available date and the associated closing price.
- C. Print the difference between the two closing prices of A) and B).
 Dynamically select the word 'decreased' if the stock went down and 'increased' otherwise.
- D. Print the closing price from 08/31/18.
- E. Print the closing price from 03/10/20.
- F. Print the difference between the closing prices of D) and E) as described above. Notice in the expected output that no minus operator should be displayed when the value decreased.

Avoid code duplication.

G. Print the first available date in 2019 and its closing price.



CE Symbol Table CODE

Give yourself a moment to come up with a plan. When you are done, compare



Expected Output

Number of elements: 5827

- A) Oldest closing price: 1997-05-15 \$1.96.
- B) Newest closing price: 2020-07-10 \$3200.00.
- C) The closing price increased by \$3198.04.
- D) Closing price from 08/31/18: \$2012.71.
- E) Closing price from 03/10/20: \$1891.82.
- F) The closing price decreased by \$120.89.
- G) First available closing price in 2019: 01/02/19 \$1539.13
- H) Last available closing price in 2020: 07/10/20 \$3200.00



Submission

Create a screen recording following the <u>guidelines for lab recordings</u> (<u>https://slcc.instructure.com/courses/817632/pages/guidelines-for-ce-recordings</u>)_.

The video should be **25-50 seconds** long.

Post the video.

✓ Previous

(<u>https://slcc.instructure.com</u>/courses/817632/modules/items

<u>/18753050</u>)

Submit Assignment

Next >

(https://slcc.instructure.com/courses/817632/modules/items

<u>/18753054</u>)

| ⟨⟨ (https://slcc.instructure.com/courses/817632/modules/items/18753041) |
|---|
| ⟨⟨intips://slcc.instructure.com/courses/817632/modules/items/18753042) |
| ⟨⟨ (https://slcc.instructure.com/courses/817632/modules/items/18753043) |
| ⟨⟨ (https://slcc.instructure.com/courses/817632/modules/items/18753044) |
| ⟨⟨ (https://slcc.instructure.com/courses/817632/modules/items/18753045) |
| ⟨⟨intigos://slcc.instructure.com/courses/817632/modules/items/18753046⟩ |
| ⟨⟨ (https://slcc.instructure.com/courses/817632/modules/items/18753047) |
| ⟨ (https://slcc.instructure.com/courses/817632/modules/items/18753048) |

⟨ (https://slcc.instructure.com/api/v1/courses/817632/module_item_redirect/18753050)

♀ (https://slcc.instructure.com/courses/817632/modules/items/18753052)

(https://slcc.instructure.com/courses/817632/modules/items/18753054)

Choose a submission type









⟨ Previous

(<u>https://slcc.instructure.com</u>/courses/817632/modules/items/18753050)

Submit Assignment

Next >

(https://slcc.instructure.com/courses/817632/modules/items/18753054)