TwoDimRaggedArrayUtility:

Pseudocode:

readFile:

temporary 2d string array

scanner object to read from file

int to hold number of rows

int to hold number of columns

while loop to loop until there is no more lines in the file

assign each line to a row in the temporary array

add 1 to number of rows if the row is not a null value

declare 2d output array of doubles with rows being the number of rows

writeToFile:

printwriter object

loops that run for each element in the 2d ragged array param

if statement to verify we are still in the row’s array

print to file: element of array +” “

if we are at the ned of a row:

print to file: element of array +”\n“

close the file

getTotal:

two loops to loop through every element in the array:

add each element to a double total variable

return the total variable

getAverage:

call get total and store it in a double total variable

two loops to loop through every element in the array:

add 1 for each element to a double count variable

return the total/count

getRowTotal:

loop through the specified row in the array:

add each element’s value to the total

return the total

getColumnTotal:

loop through each row in the array:

if the row has the column specified:

add the element’s value to the total

return the total

getHighestInRowIndex:

create highest and index double vars and initialize them to the value at the first elem int the row and the index in the row

loop through the specified row and check if each element in the row is higher than the highest variable:

if it is then index = current index, and highest = current elem’s value

return index

getHighestInRow:

return the element in the array at the specified row and the column returned from getHighestInRowIndex method

getLowestInRowIndex:

create lowest and index double vars and initialize them to the value at the first elem int the row and the index in the row

loop through the specified row and check if each element in the row is lower than the lowest variable:

if it is then index = current index, and lowest = current elem’s value

return index

getLowestInRow:

return the element in the array at the specified row and the column returned from getLowestInRowIndex method

getHighestInColumnIndex:

create and initialize doubles highest, index and a Boolean flag (to false)

loop through each row in the array and check if the row has the specified column

check if highest and index have been initialized to values in the array (use the f lag)

set flag to true

initialize highest and index to the current values

if highest and index have been initialized to values in the array:

check if the current elem is higher than highest and replace highest and index to this elem’s values if it is

return the index

getHighestInColumn:

return the element in the array at the row returned from the getHighestInColumnIndex method and the specified column.

getLowestInColumnIndex:

create and initialize doubles lowest, index and a Boolean flag (to false)

loop through each row in the array and check if the row has the specified column

check if lowest and index have been initialized to values in the array (use the f lag)

set flag to true

initialize lowest and index to the current values

if highest and index have been initialized to values in the array:

check if the current elem is lower than lowest and replace lowest and index to this elem’s values if it is

return the index

getLowestInColumn:

return the element in the array at the row returned from the getLowestInColumnIndex method and the specified column.

getHighestInArray:

initialize double highest to value returned from getHighestInRow method called with the array and the first row

check through each row and compare value returned from getHighestInRow method with the highest value and replace highest if it is higher

return highest

getLowestInArray:

initialize double lowest to value returned from getLowestInRow method called with the array and the first row

check through each row and compare value returned from getLowestInRow method with the Lowest value and replace Lowest if it is lower

return lowest

UML Diagram:

Text

Description automatically generated

HolidayBonus:

Pseudocode:

calculateHolidayBonus:

get the length of the longest row using a loop and comparing each row’s length it to a int var (hirow) which represents the length and replace hirow if the current row’s length is longer

makes output array with the length of the rows in the data array

two int vars for highest index and lowest index

initialize output array with length the amount of rows in the param array

loop through the length of the longest row:

highestind = value returned from getHighestInColumnIndex method called with the current column

lowestind = value returned from getlowestInColumnIndex method called with the current column

loop through the output array and add the bonus amount based off of each index:

index = highest index add the high amount to the current index in output array

index = lowest index add the low amount to the current index in output array

index is not in that row in param array then add nothing

else: add the other param value to the elem at the current index in the output array

calculateTotalHolidayBonus

create double bonuses array to hold array returned from calculateHolidayBonus method

create double total var

loop through bonuses array and add each value to total

return total

UML Diagram:

Graphical user interface, text, application

Description automatically generated

TestCases:

TwoDimRaggedArrayUtility:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Input Array:  [4,5,6  7,3,-1] | testGetTotal: | | testGetAverage: | testGetRowTotal: | testGetColumnTotal: | testGetHighestInArray: | testWriteToFile: |
| expected | | 24.0 | 4.0 | 9.0  15.0 | 11.0  8.0  5.0 | 7.0 | 4  5  6  7  3  -1 |
| actual | | 24.0 | 4.0 | 9.0  15.0 | 11.0  8.0  5.0 | 7.0 | 4  5  6  7  3  -1 |

A screenshot of a computer

Description automatically generated with medium confidence

HolidayBonus:

|  |  |  |  |
| --- | --- | --- | --- |
| Input Array:  [4,5,6  7,3,-1]  High=5000  Low=1000  Other=2000 | testCalculateHolidayBonus: | | testCalculateTotalHolidayBonus: |
| expected | | 8000.0  7000.0  9000.0 | 24000.0 |
| actual | | 8000.0  7000.0  9000.0 | 24000.0 |

A screenshot of a computer

Description automatically generated

Input1:

Graphical user interface, application

Description automatically generated

Table

Description automatically generated with medium confidence

Input2:

Graphical user interface, text, application

Description automatically generated

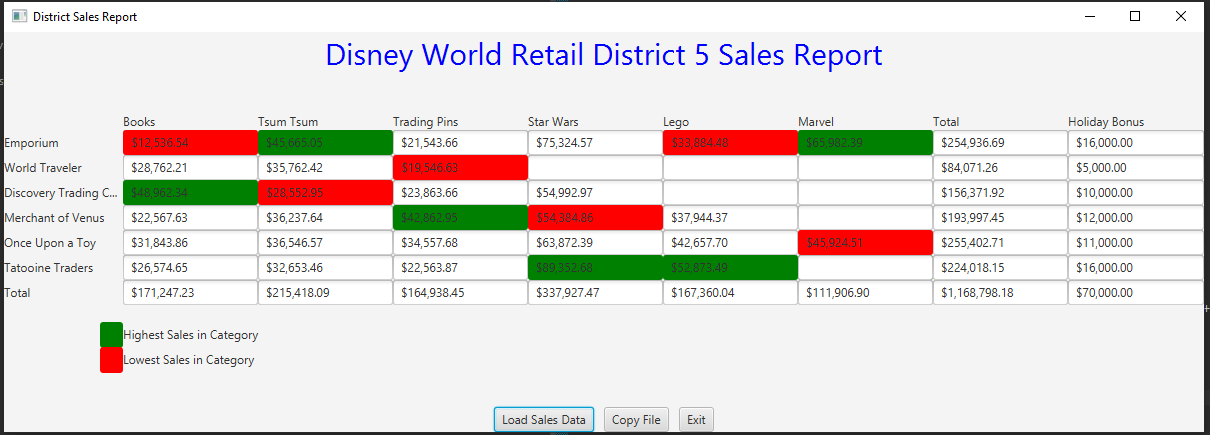
A picture containing timeline

Description automatically generated

District5:

Text

Description automatically generated



* Learning Experience: highlight your lessons learned and learning experience from working on this project.
* What have you learned?
  + I’ve learned about writing test cases, and manipulating 2d arrays
* What did you struggle with?
  + I struggled with getting the holidaybonuses to work with (- numbers)
* What will you do differently on your next project?
  + I will try to put more planning into my coding before I start doing it
* Include what parts of the project you were successful at, and what parts (if any) you were not successful at.
  + I was successful in all parts of the project except for passing dataset4 in the tests for holidaybonuses. I’m still not sure why my code failed those tests or if the tests themselves were flawed.