

## Test Procedure: LOAFR Execution

### **Test Case 1** Single CSV File read

Purpose:

Find all logfile entries where the player is on the Bengals team

Owner:

Test/Tools Group

Expected Results:

The data will be loaded without errors and the user will be able to narrow down the search to return just the entries which contain players from the Bengals. In this logfile of ~ 30 players, 3 entries contain Bengals players.

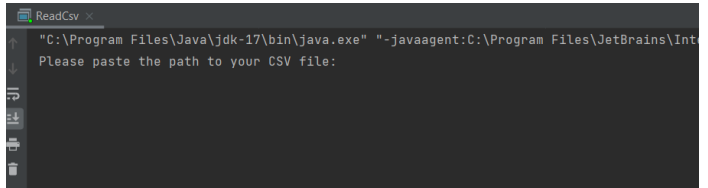
Test Data:     Data Directory: /Loafr-Project/testFiles  
                  Datafile: testSingleCsv.csv

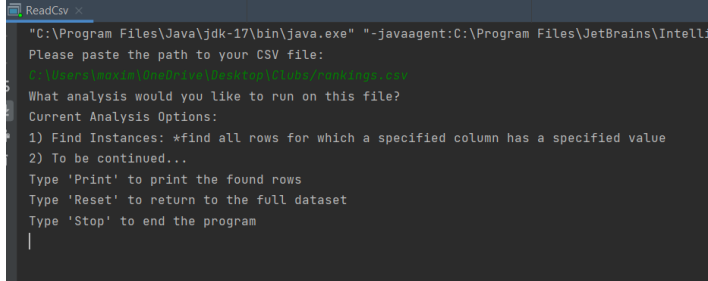
Test Tools:     JUnit5 Testing Framework

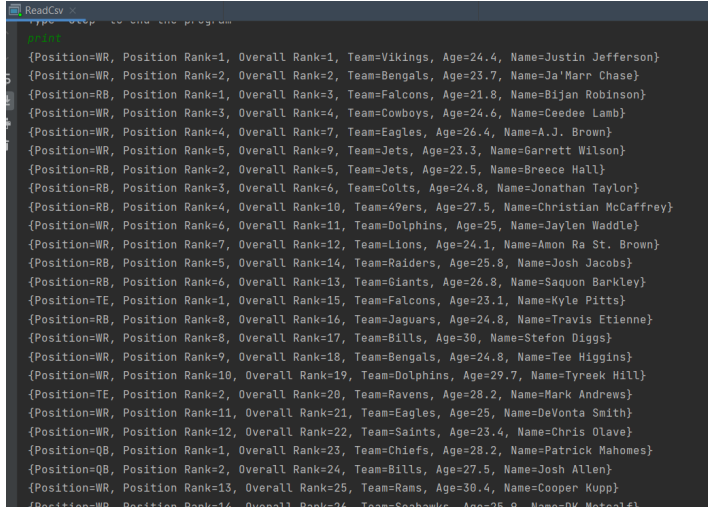
Dependencies: N/A

Initialization: LOAFR System program is built in IDE of choice.

Scenario: The user has a csv file which contains a list of a subset of NFL players and information about them. They would like to see which players in the list play for the Bengals. The user's goal is to narrow down the list of players to only those whose team is the Bengals so that they don't have to search through the whole file.

<b>Step 1</b>	The user runs the loafr system
Expected Result	The output terminal will open and prompt the user to input a csv file or file path
Scenario Result	

<b>Step 2</b>	The user adds a path to the rankings csv file on their computer
Expected Result	The output terminal will accept the file and open the list of options
Scenario Result	 <pre> ReadCsv "C:\Program Files\Java\jdk-17\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA\bin\idea_rt.jar=5000:C:\Program Files\Java\jdk-17\bin" -Dfile.encoding=UTF-8 Please paste the path to your CSV file: C:\Users\jason\Documents\jason\top\jason\rankings.csv What analysis would you like to run on this file? Current Analysis Options: 1) Find Instances: *find all rows for which a specified column has a specified value 2) To be continued... Type 'Print' to print the found rows Type 'Reset' to return to the full dataset Type 'Stop' to end the program 1 </pre>

<b>Step 3</b>	The user types "Print" to see the entire dataset they are working with
Expected Result	The output terminal will print all of the data from the csv
Scenario Result	 <pre> ReadCsv Type 'Print' to see the entire dataset they are working with Print {Position=WR, Position Rank=1, Overall Rank=1, Team=Vikings, Age=24.4, Name=Justin Jefferson} {Position=WR, Position Rank=2, Overall Rank=2, Team=Bengals, Age=23.7, Name=Ja'Marr Chase} {Position=RB, Position Rank=1, Overall Rank=3, Team=Falcons, Age=21.8, Name=Bijan Robinson} {Position=WR, Position Rank=3, Overall Rank=4, Team=Cowboys, Age=24.6, Name=Ceedee Lamb} {Position=WR, Position Rank=4, Overall Rank=7, Team=Eagles, Age=26.4, Name=A.J. Brown} {Position=WR, Position Rank=5, Overall Rank=9, Team=Jets, Age=23.3, Name=Garrett Wilson} {Position=RB, Position Rank=2, Overall Rank=5, Team=Jets, Age=22.5, Name=Breece Hall} {Position=RB, Position Rank=3, Overall Rank=6, Team=Colts, Age=24.8, Name=Jonathan Taylor} {Position=RB, Position Rank=4, Overall Rank=10, Team=49ers, Age=27.5, Name=Christian McCaffrey} {Position=WR, Position Rank=6, Overall Rank=11, Team=Dolphins, Age=25, Name=Jaylen Waddle} {Position=WR, Position Rank=7, Overall Rank=12, Team=Lions, Age=24.1, Name=Amon Ra St. Brown} {Position=RB, Position Rank=5, Overall Rank=14, Team=Raiders, Age=25.8, Name=Josh Jacobs} {Position=RB, Position Rank=6, Overall Rank=13, Team=Giants, Age=26.8, Name=Saquon Barkley} {Position=TE, Position Rank=1, Overall Rank=15, Team=Falcons, Age=23.1, Name=Kyle Pitts} {Position=RB, Position Rank=8, Overall Rank=16, Team=Jaguars, Age=24.8, Name=Travis Etienne} {Position=WR, Position Rank=8, Overall Rank=17, Team=Bills, Age=30, Name=Stefon Diggs} {Position=WR, Position Rank=9, Overall Rank=18, Team=Bengals, Age=24.8, Name=Tee Higgins} {Position=WR, Position Rank=10, Overall Rank=19, Team=Dolphins, Age=29.7, Name=Tyreek Hill} {Position=TE, Position Rank=2, Overall Rank=20, Team=Ravens, Age=28.2, Name=Mark Andrews} {Position=WR, Position Rank=11, Overall Rank=21, Team=Eagles, Age=25, Name=DeVonta Smith} {Position=WR, Position Rank=12, Overall Rank=22, Team=Saints, Age=23.4, Name=Chris Olave} {Position=QB, Position Rank=1, Overall Rank=23, Team=Chiefs, Age=28.2, Name=Patrick Mahomes} {Position=QB, Position Rank=2, Overall Rank=24, Team=Bills, Age=27.5, Name=Josh Allen} {Position=WR, Position Rank=13, Overall Rank=25, Team=Rams, Age=30.4, Name=Cooper Kupp} {Position=RB, Position Rank=14, Overall Rank=26, Team=Seahawks, Age=25.9, Name=DK Metcalfe} </pre>

<b>Step 4</b>	The options list is back on the terminal, so the user chooses option one and types 'Find Instances'
---------------	---

Expected Result	The output terminal will ask for the column they'd like to look for data in, and then it will ask for the keyword that the user wants to see in that column
Scenario Result	<pre> {Position=RB, Position Rank=14, Overall Rank=38, Team=Cowboys, Age=123.9, Name=Tony Pollard} {Position=QB, Position Rank=4, Overall Rank=39, Team=Bengals, Age=123.9, Name=Joe Burrow} What analysis would you like to run on this file? Current Analysis Options: 1) Find Instances: *find all rows for which a specified column has a specified value 2) To be continued... Type 'Print' to print the found rows Type 'Reset' to return to the full dataset Type 'Stop' to end the program Find Instances </pre>

<b>Step 5</b>	The user chooses the column "Team" and the keyword "Bengals" to filter for Bengals players like they want
Expected Result	The output terminal will ask for the column they'd like to look for data in, and then it will ask for the keyword that the user wants to see in that column. The options menu comes back up, but the new working dataset is the filtered set that the user wanted
Scenario Result	<pre> Type 'Print' to print the found rows Type 'Reset' to return to the full dataset Type 'Stop' to end the program Find Instances Please choose the column of data that you'd like to filter Team Choose the keyword/value that you're searching for in that column Bengals What analysis would you like to run on this file? Current Analysis Options: 1) Find Instances: *find all rows for which a specified column has a specified value 2) To be continued... Type 'Print' to print the found rows Type 'Reset' to return to the full dataset Type 'Stop' to end the program </pre>

<b>Step 6</b>	The user types "Print" to see their filtered set
Expected Result	The output terminal print the entries of the original column for which the players are on the Bengals team

Scenario Result	<pre> Bengals What analysis would you like to run on this file? Current Analysis Options: 1) Find Instances: *find all rows for which a specified column has a specified value 2) To be continued... Type 'Print' to print the found rows Type 'Reset' to return to the full dataset Type 'Stop' to end the program Reset {Position=WR, Position Rank=2, Overall Rank=2, Team=Bengals, Age=23.7, Name=Ja'Marr Chase} {Position=WR, Position Rank=9, Overall Rank=18, Team=Bengals, Age=24.8, Name=Tee Higgins} {Position=QB, Position Rank=4, Overall Rank=39, Team=Bengals, Age=123.9, Name=Joe Burrow} What analysis would you like to run on this file? Current Analysis Options: 1) Find Instances: *find all rows for which a specified column has a specified value 2) To be continued... Type 'Print' to print the found rows Type 'Reset' to return to the full dataset Type 'Stop' to end the program </pre>
-----------------	---

Step 7	The user types "Stop" to end the program
Expected Result	The program will end
Scenario Result	<pre> 1) Find Instances: *find all rows for which a specified column has a specified value 2) To be continued... Type 'Print' to print the found rows Type 'Reset' to return to the full dataset Type 'Stop' to end the program Stop Process finished with exit code 0 </pre>

### Requirements Met to the Required Extent for this Limited Implementation

- see SRS-2nd rev from P2 for requirements documentation

REQ-1  
 REQ-2  
 REQ-3  
 REQ-5  
 REQ-16  
 REQ-17  
 REQ-18 (half of it)  
 REQ-24  
 REQ-25  
 REQ-28  
 REQ-30  
 REQ-33  
 REQ-35  
 REQ-36

## **Unit Test Cases 2-9**

Purpose:	Verify that the LOAFR Software works as intended
Owner:	Test/Tools Group
Expected Results:	Vary per function; read() will return a list of hashmaps that consist of the data in the csv file, findInstances() will return a list of hashmaps that consist of the data you specified, and printList() will print every element in the current list
Test Data:	Data Directory: /Loafr-Project/testFiles Datafile: testSingleCsv.csv, testMultipleCsv.csv
Test Tools:	JUnit5 Testing Framework
Dependencies:	None
Initialization:	Initialize objects from the ReadCsv, Analyze, and Output classes. Then setup the expected lists that will be produced by each class function
Description:	The datafiles are manipulated and tested against the expected results to see whether the function works as intended

Function	Test Cases	Findings
read()	2, 3, 4	Random unidentifiable character would append to a header from the csv file, successfully able to replace it with an empty character since it was never input in the string in the first place
findInstances()	5, 6, 7	None
printList()	8, 9	None

Test Case	Date Last Run	Pass or Fail
2: Single CSV Entry read()	11-21-23	P
3: Multiple CSV Entries read()	11-21-23	P
4: read() Error	11-21-23	P
5: Single CSV Entry findInstances()	11-21-23	P
6: Multiple CSV Entries findInstances()	11-21-23	P
7: findInstances() Error	11-21-23	P
8: Single CSV Entry printList()	11-21-23	P
9: Multiple CSV Entries printList()	11-21-23	P