

CSIS 3280 – Lab 05 – Team Roster

Visual Studio Code

1. Download the lab template from Blackboard
2. Extract it
3. Fill in the relevant code to create the solution and meet the requirements

Solution

The flow of the program is as follows:

1. The user will be prompted to upload a CSV file. When the file is uploaded the roster will be displayed with the players sorted by last name and a count at the bottom, please see the appendix.
2. The user will be able to sort the data by any of the items in the file such as player no, firstname, lastname, batting average, age, throwing hand, height, weight, birthplace.

Requirements:

1. The team will display a total of the number of player objects it contains when printed
2. The team will be sorted by last name
3. The CSV file will be parsed into Player objects
4. The player objects will all be added to a team object. All logic for sorting the roster will be in the team object
5. The HTML form will prompt the user to upload a CSV file (you will use data/orioles.csv)
6. Your code must only be using classes (except for the controller file)
7. You must use a try/catch clause any time you are the data file
8. You must close your file handle when you are finished with the file.
9. You must not leave any errors on the page.
10. You must use a static class for all classes except the Player and the Team.
11. The user can upload the file as many times as they want.
12. If the user chooses no sort order the data should be sorted by age.

Files and Template

The following classes and respective files are required for this lab:

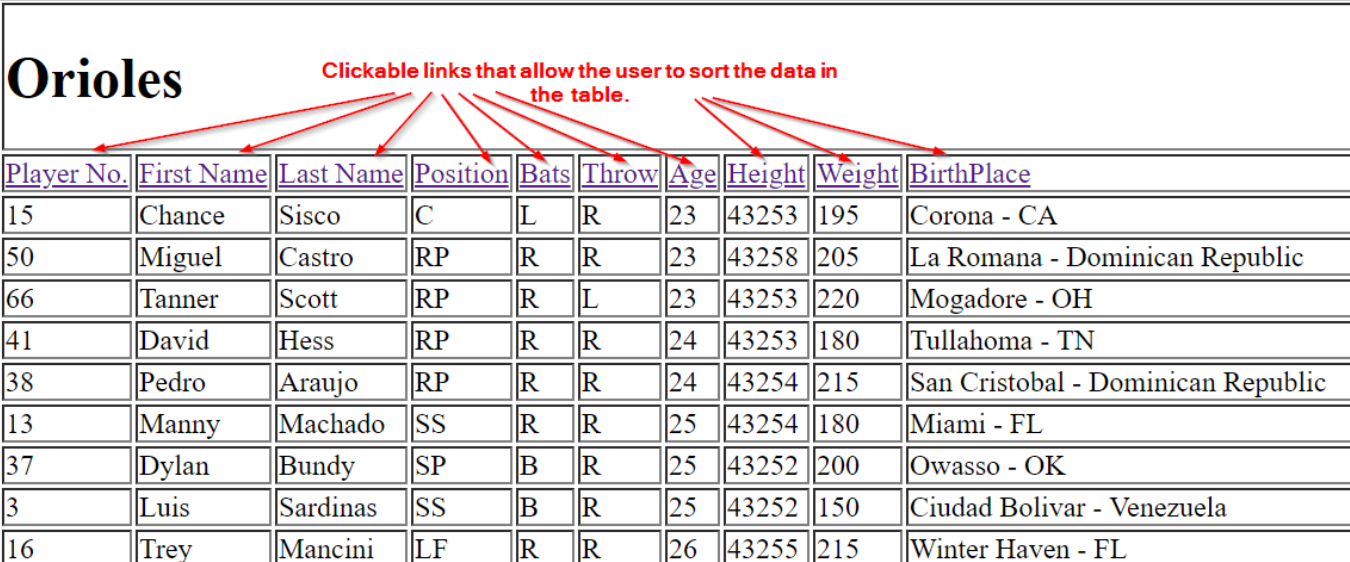
Class Name	File Name	Description
Player	inc/player.class.php	Contains the Player class, must have a constructor and toString methods.
Team	inc/team.class.php	Contains the team with storage for the players in the team
FileAgent	inc/FileAgent.class.php	This is the static utility class for getting the file contents
TeamParser	inc/TeamParser.class.php	This is a static utility class for parsing the CSV file

Class Name	File Name	Description
Page	inc/Page.class.php	This is the static Page class that is used to display parts of the page (and HTML) to the browser.
None	inc/config.inc.php	Stores where the appData file should be stored

Hints:

- Refer to the demo code for Week 5
- Remember you should not have any errors, warnings, or info messages in your application
- You must not have any logic that should be handled by your classes in the controller file.
- Keep a clean separation between the view and the controller.
- Ensure you use the appropriate class types.
- You may have to store the file somewhere when it is originally uploaded.

Appendix:



Orioles

Clickable links that allow the user to sort the data in the table.

Player No.	First Name	Last Name	Position	Bats	Throw	Age	Height	Weight	BirthPlace
15	Chance	Sisco	C	L	R	23	43253	195	Corona - CA
50	Miguel	Castro	RP	R	R	23	43258	205	La Romana - Dominican Republic
66	Tanner	Scott	RP	R	L	23	43253	220	Mogadore - OH
41	David	Hess	RP	R	R	24	43253	180	Tullahoma - TN
38	Pedro	Araujo	RP	R	R	24	43254	215	San Cristobal - Dominican Republic
13	Manny	Machado	SS	R	R	25	43254	180	Miami - FL
37	Dylan	Bundy	SP	B	R	25	43252	200	Owasso - OK
3	Luis	Sardinas	SS	B	R	25	43252	150	Ciudad Bolivar - Venezuela
16	Trey	Mancini	LF	R	R	26	43255	215	Winter Haven - FL

STOP! - This is a pre-assignment submission checklist!

- Did you remove any **debug output such as var_dump**
- Is your submission clear of any **ORANGE** output from XDebug (Warnings, Info or Errors)?
- Did you follow the naming convention for your files?!
- Did you follow the naming convention for your folder?!
- Does your submission work on a lab computer?!
- Double check **before** submitting

Copyright (c)2018 Rahim Virani and others. NOT FOR REDISTRUBUTION.
STUDENTS FOUND REDISTRUBUTING COURSE MATERIAL WILL BE IN VIOLATION OF ACADEMIC
INTEGRITY POLICIES AND WILL FACE DISCIPLINARY ACTION BY COLLEGE ADMINISTRATION.

