**Assignment 3 – Node’s Module System**

**Instructions**

1. ~~Copy the sports-app solution from week two and add it to your week three directory.~~
2. ~~Create a new file named team.js.~~

**Additional Programming Requirements**

1. ~~Create a class named Team with properties for name, mascot, and playerCount.~~
2. ~~Initialize the classes properties in the constructor.~~
3. Export the class using Node’s module system.
4. ~~Create a new file named team-manager.js.~~

**Additional Programming Requirements**

1. ~~Create an array of five Team objects and assign it to a variable named teams. For this step, make sure you use the Team object from the team.js file to populate each element in the array. There must be a minimum of five team objects in the array.~~
2. ~~Create a function named getTeams and return the array of Team objects. Export the function using Node’s module system.~~
3. ~~Create a function named getTeam with a single parameter for name. Using JavaScript’s built-in find() function, return a single Team object from the teams array. Use the parameter value in the find() function. Export the function using Node’s module system.~~
4. ~~Create a function named displayTeam with a single parameter for a team object. In the body of the function, return a formatted string that uses the team objects properties (name, mascot, and playerCount). The expected format is in Exhibit 1.~~

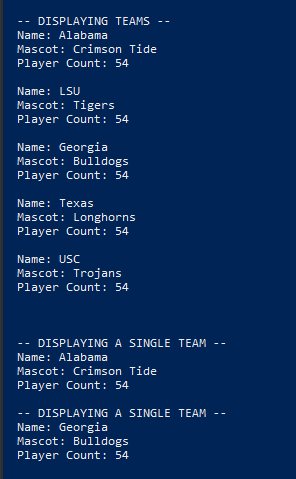
**Exhibit 1.**



**Special note**. You will need to concatenate the string together with the + operator and use the \n operator to start each entry on a separate line.

1. In the index.js file, import the TeamManager module from the team-manager.js file and write the necessary code to display the array of Team objects and two single team objects. For the single team objects, create two local variables and call the getTeam() function to populate them. For all output, you must use the dispalyTeam() function from the team-manager.js file.

**Exhibit 2. Expected Output (use your own data)**



**Special note**. Your output must closely match mine. Pay attention to line spaces, indention, and spelling. The courses GitHub repository has an example project (week-3/fruits) that will help you get through this week’s assignment.