

# Database-Driven Web Design & Development

## Assessment1 - 60 points (Assessing material and concepts from Modules 1, 2, and 3)

In this exercise, you will be reading data from the following input file, **aoeCivInfo.txt**, which you can assume is in the same folder as the .php file you are coding in.

Britons:1:Longbowman:Yeoman,Warwolf:80450  
Magyars:1:Magyar Huszar:Mercenaries,Recurve Bow:45718  
Italians:1:Genoese Crossbowman:Pavise,Silk Road:55030  
Slavs:2:Boyar:Orthodoxy,Druzhina:28223  
Saracens:2:Mameluke:Madrasah,Zealotry:19426  
Turks:2:Janissary:Sipahi,Artillery:32677  
Franks:2:Throwing Axeman:Chivalry,Bearded Axe:30078  
Vikings:1:Berserk:Chieftains,Berserkergang:45472

Each line (record) in the file has fields separated by a delimiter character that represent the following data values in order for a civilization in the Age of Empires computer game:

- Name
- Team number the civ belonged to in the last played game
- Special unit
- Unique technologies (note this field contains a list of two values separated by another delimiter character)
- Game score each civ achieved in the last played game

### Your tasks are as follows:

- Rename the **Assessment1\_startFiles** folder to **Assessment1\_username** (where username should be replaced with your CVTC username).
- Create a new PHP file in the same folder as your .txt files named **aoeCivs.php**.

#### In aoeCivs.php:

- Use **html:5** in VSCode to create the HTML5 template in your file. Put a <div> with a class of “wrapper” inside the <body> tag. All other tags and content you create will be put inside the wrapper div.
- Link in your **aoeCivs.css** external stylesheet.
- Initialize two variables named **\$highScore** (to 0) and **\$highScoreCiv** (to empty string).

## Time for File I/O:

- Open the `aoeCivInfo.txt` file in read-only mode and store a file pointer to it in `$fp`.
- Use a **while** loop to read in data from the file one line at a time (storing each line in a variable named `$civLine` - get rid of any leading AND trailing white space characters) until you reach end-of-file.
- Don't forget to check for the issue with the input file having an extra blank line.

## If the line of data you just read is valid (not empty) manipulate and store the read data:

- Break up each read line of data into fields storing the values from left-to-right in variables named `$civName`, `$team`, `$specialUnit`, `$uniqueTechs`, and `$score`.
- Break up `$uniqueTechs` (which is a string value with two fields in it) into its individual field values storing them in an array named `$uniqueTechnologies`.
- Store the individual pieces of data into associative arrays by **civilization name** for later access using the following array names: `$civTeams`, `$civSpecialUnits`, `$civTechs`, and `$civScores`.

Regarding these associative arrays, you are *storing each piece of data by civilization name* as these names are unique and tie all of the data values about each civilization together.

- In the same loop where you are reading the file's data, sum up each team's score in an associative array named `$teamScores` whose keys are the team number and values are the cumulative (totaled) scores for the team.
- In the same loop where you are reading the file's data, determine the high score and the civilization name that has the high score storing each in the variables you initialized earlier (`$highScore` and `$highScoreCiv`).
- **After the file data has been completely read and stored**, use a single **foreach** loop to step through your `$civTeams` associative array storing the **key** of each element into a variable named `$civ` and the **value** of each element into a variable named `$civTeam`.

### Inside the above foreach loop:

- *Convert the civ name to lowercase* using a PHP string-related function storing the resulting string in a variable named `$civLowerCase`. You will use the variable to build the image file name for the civ's Wonder image.
- *Jump out of PHP* and print out the following data for each civilization that was stored in your associative arrays using the HTML tag structure shown below (tag structure will be graded). \*\*\*The underlined text below is where your data values will need to be provided using PHPs *shortcut for the echo command* by jumping back into PHP and then back out again:

```
<section class="civ">
  <aside class="shadowEffect">
    
  </aside>

  <div class="civInfo">
    <p class="civName">civ name here</p>
    <ul id="mainUL">
      <li>Unique Unit: unit value here</li>
      <li>Unique Technologies:
        <ul>
          <li>first unit technology here</li>
          <li>second unit technology here</li>
        </ul>
      </li>
      <li>Game Score: score here</li>
    </ul>
  </div>
</section>
```

<div class="scores"> Note: this line of HTML code should be **AFTER** the above foreach loop!

- After the previous foreach loop, code another **foreach** loop to step through your `$teamScores` associative array and print out the following output inside an `<h3>` tag (note: `\t(s)` and `\n` needed). Again, the underlined text is where your output data values need to be output.

Team #team number here had a cumulative score of team score here

- Finally, print the following output inside an `<h3>` tag **once at the end** (note: `\t(s)` needed).

The high score civ name here had the high score of high score here points

followed by ending `</div>` tag for `div.scores`:

`</div>`

Be sure to carefully follow the above specs as I will use these in grading the details of this assessment. When finished, zip up your project folder and submit it via the provided dropbox in Canvas. Be sure your username is part of the folder name.