Thursday: Introduction; Tic Tac Toe; Bulleter

Introduction

So we have covered a lot of ground so far. Let us use some of the skills we have learnt to build a few projects.

Tic Tac Toe

Create a tic tac toe game where two users can can compete with each other. **Here are the rules (http://web.cecs.pdx.edu/~bart/cs541-fall2001/homework/tictactoe-rules.html)** .

Features:

1. Outline all the behaviors your application will have.
2. Create a Player class for both players.
3. Draw the tic tac toe grid and update it when users play on a side.
4. Create unit tests for all the behaviors.
5. Create doc-strings for the modules, functions and methods

Bulleter

Take a file and add `*` in front of every line and capitalize the first word. Write updated items in a another file. Also copy the new file to the clipboard

1. Outline all the behaviors your application will have.
2. Create unit tests for all the behaviours.
3. Create doc-strings for the modules, functions and methods

CSV parser

Comma Separated Values are files that store a large number of variables and data. They act Like a simple spreadsheet. Parsing through CSVs and manipulating data is one of a Python developers greatest skills. Using this **tutorial (http://www.pythonforbeginners.com/systems-programming/using-the-csv-module-in-python/)** Learn how to use the CSV module in Python.

When you have completed that Tutorial You will download this file **Job Data (https://drive.google.com/file/d/0B7-p5eNhYCPwS2NIMjhKMkR5cm8/view?usp=sharing)** . This is a CSV file that contains different jobs from a Jobs Board.

Your task Is to parse the CSV file and group all the Jobs according to the company that posted the jobs. So you create a folder that will contain all the jobs according to the company that posted

the jobs.

When that is done you will go into those individual folders and categorize the jobs from that company into:

1. Employment-type.
2. Location.

3. Industry.

4. Outline all the behaviors your application will have.

5. Create unit tests for all the behaviors.
6. Create doc-strings for the modules, functions and methods

Copyright 2017 Moringa School.