By submitting this assignment, I agree to the following:
"Aggies do not lie, cheat, or steal, or tolerate those who do."
"I have not given or received any unauthorized aid on this assignment."
#
Names: Aaron Garcia
Dhruv Manihar
Ethan Nguyen
Mariana Sanchez
Section: 514
Assignment: Lab 7a-1
Date: 18 10 2021
#
#
#
#

Checkers moves - using lists of lists in 2-D matrix
Create program that sets up a standard checkers board and lets users make moves
Structure of the game/how program should work
Several small functions

Specify system
User input instructions
Dark squares numbered 1 through 32 start top left and continuing right and down
8x8 board: make list of lists to store current board
Display current state of each board before every move
Each empty square should just be a period/each square with a piece should have identifier
        lower-case/light piece upper-case/dark piece
When piece moved/make that empty after and occupies new square takes other piece out
Continue let them move pieces until they enter stop

Restrictions:
User tries to move piece where there is no piece “try again”
Only move diagonally “try again”

Instructions
#first make the variables
- Make the identifiers(o,O)
- Make the positions on the board (an empty space is a “.”)

#create the board
- Make width and length of board and columns
- Lists for each row

#loop
- While the user input remains true

- Print the board
- Use for loop to keep the board in range

Ask for where they want to move (user input)

#if else statement

- If user chooses "." print error message
- Else it will replace that space with the"."
    - Ask for player moves

      # player moves
        - Start with light or dark colors
            - Prompt user to move a piece of their choice (user input)
            - Ask for row
            - Ask for column
                - Making sure they are picking a piece
                - Moving diagonally
                - Print error message if those don't comply(error message)
        - next player moves either light or dark
            - Prompt user to move a piece of their choice (user input)
            - Ask for row
            - Ask for column
                - Making sure they are picking a piece
                - Moving diagonally
                - Print error message if those don't comply(error message)