#Thomas Morris

#September 12, 2019

#Assignment 1

#Tic Tac Toe

class ttt:

#two demensional list to hold board

board = [[' ',' ',' '],[' ',' ',' '],[' ',' ',' ']]

#boolean to hold win

win = False

#String to hold winner

winner = ""

def getWinner():

return ttt.winner

#reset board after game

def resetBoard():

ttt.board = [[' ',' ',' '],[' ',' ',' '],[' ',' ',' ']]

#prints the board to the console

def printBoard():

for i in range(3):

print("-----------")

for j in range(3):

if j==0:

print("|", end='')

print(" " + ttt.board[i][j] + "|", end='')

if i==2:

print()

print("-----------", end='')

print()

print()

#checks if someone won

def checkWin():

#check rows for winner

for i in range(3):

if(ttt.board[i][0] == ttt.board[i][1] and ttt.board[i][0] == ttt.board[i][2] and ttt.board[i][0] != ' '):

ttt.winner = ttt.board[i][0]

return True

#check cols for winner

for i in range(3):

if(ttt.board[0][i] == ttt.board[1][i] and ttt.board[0][i] == ttt.board[2][i] and ttt.board[0][i] != ' '):

ttt.winner = ttt.board[0][i]

return True

#check first diagonal

if(ttt.board[0][0] == ttt.board[1][1] and ttt.board[0][0] == ttt.board[2][2] and ttt.board[0][0] != ' '):

ttt.winner = ttt.board[0][0]

return True

#check second diagonal

if(ttt.board[0][2] == ttt.board[1][1] and ttt.board[0][2] == ttt.board[2][0] and ttt.board[0][2] != ' '):

ttt.winner = ttt.board[0][2]

return True

#check for tie

count = 0

for r in range(3):

for c in range(3):

if(ttt.board[r][c] != ' '):

count += 1

if(count == 9):

ttt.winner = "tie"

return True

return False

def printWinner():

if(ttt.winner == ""):

return

if(ttt.winner == "tie"):

print("It's a tie!")

elif (ttt.winner == 'X'):

print("X wins!")

elif (ttt.winner == 'O'):

print("O wins!")

else:

print("I don't know how you got here.")

#method to control x's turn

def xTurn():

xTurnOver = False

while xTurnOver == False:

#collect integer input from user

xr = int(input("Enter X row position: "))

while xr < 1 or xr > 3:

print("Invalid input. Please enter a number 1-3")

xr = int(input("Enter X row position: "))

xc = int(input("Enter X col position: "))

while xr < 1 or xr > 3:

print("Invalid input. Please enter a number 1-3")

xc = int(input("Enter X col position: "))

#check if the position is empty

if ttt.board[xr-1][xc-1] == ' ':

ttt.board[xr-1][xc-1] = 'X'

xTurnOver = True

else:

print("Position taken. Please try again")

xTurnOver = False

#method to control O's turn

def yTurn():

yTurnOver = False

while yTurnOver == False:

#collect integer input from user

yr = int(input("Enter O row position: "))

while yr < 1 or yr > 3:

print("Invalid input. Please enter a number 1-3")

yr = int(input("Enter O row position: "))

yc = int(input("Enter O col position: "))

while yr < 1 or yr > 3:

print("Invalid input. Please enter a number 1-3")

yc = int(input("Enter O col position: "))

#check if the position is empty

if ttt.board[yr-1][yc-1] == ' ':

ttt.board[yr-1][yc-1] = 'O'

yTurnOver = True

else:

print("Position taken. Please try again")

yTurnOver = False

class playGame:

ttt.printBoard()

while ttt.checkWin() == False:

if(ttt.checkWin() == False):

#runs X's turn

ttt.xTurn()

ttt.checkWin()

ttt.printBoard()

ttt.printWinner()

if(ttt.checkWin() == False):

#runs O's turn

ttt.yTurn()

ttt.checkWin()

ttt.printBoard()

ttt.printWinner()