Plagiarism Scan Report

Report Title	Abdullah report
Generated Date	20-Mar-2024
Total Words	361
Total Characters	3613
Report Generated By	Plagiarismchecker.co
Excluded URL	None

Plagiarised	Unique	Total Words Ratio
8%	92%	87.52%

Content Checked For Plagiarism

```
class TicTacToe:
def __init__(self):
self.board = [" ", " ", " ", " ", " ", " ", " ", " ", " "]
self.player1 = ""
self.movess = 0
self.player1Position = []
self.player2 = ""
self.player2Position = []
self.winner = None
self.depth = []
self.moves = []

def drawBoard(self):
print(" %c | %c | %c " % (self.board[0], self.board[1], self.board[2]))
print("___|___|___")
print(" %c | %c | %c " % (self.board[3], self.board[4], self.board[5]))
print("___|___|___")
print(" %c | %c | %c " % (self.board[6], self.board[7], self.board[8]))
print(" | | ")

def choice(self):
```

```
player = input("Which character you want to play as in the game ? (choose between x or o) ")
if player == "X" or player == "x":
self.player1 = "X"
self.player2 = "O"
else:
self.player1 = "O"
self.player2 = "X"

def avMov(self):
for i in range(0, len(self.board)):
if self.board[i] == " ":
self.moves.append(i)
return self.moves

def result(self):
self.gamewin()
if self.winner == self.player2:
print("Player2 Wins Congratulations You Have succeeded in your task")
exit(0)
elif self.winner == self.player1:
print("Player1 Wins Congratulations You Have succeeded in your task")
exit(0)

def gamewin(self):
winningPositions = [{0, 1, 2}, {3, 4, 5}, {6, 7, 8},
{0, 4, 8}, {2, 4, 6}, {0, 3, 6},
{1, 4, 7}, {2, 5, 8}]
for position in winningPositions:
if position.issubset(self.player1Position):
self.winner = self.player1
return True
elif position.issubset(self.player2Position):
self.winner = self.player2
return True
if self.board.count(" ") == 0:
self.winner = "tie"
return True
self.winner = None
return False

def minimax(self, isMaximizing):
self.depth = {
self.player1: -1,
self.player2: 1,
```

```
"tie": 0
}
if self.gamewin():
return self.depth[self.winner]
if isMaximizing:
bestScore = float("-inf")
for i in self.avMov():
self.player2Position.append(i)
self.board[i] = self.player2
bestScore = max(self.minimax(False), bestScore)
return bestScore
else:
bestScore = float("inf")
for i in self.avMov():
self.player1Position.append(i)
self.board[i] = self.player1
bestScore = min(self.minimax(True), bestScore)
return bestScore

def play(self):
self.choice()
while not self.result():
if self.movess % 2 == 0:
pos = int(input("Where would you like to play mention the location on grid? (0-8) "))
self.player1Position.append(pos)
self.board[pos] = self.player1
else:
pos = int(input("Where would you like to play mention the location on grid ? (0-8) "))
self.player2Position.append(pos)
self.board[pos] = self.player2
self.movess += 1
print("\n")
self.drawBoard()
print("Thanks for playing :)")

if __name__ == '__main__':
game = TicTacToe()
game.play()
```

9% **Python tic tac toe class**

Apr 26, 2018 — ... 5), (6, 7, 8), (0, 4, 8), (2, 4, 6), (0, 3, 6), (1, 4, 7), (2, 5, 8)) self.moves_count = 0 def create_board(self): print() print(self.board[0] ...

https://codereview.stackexchange.com/questions/193004/python-tic-tac-toe-class