PAGE :
DATE :

Tic Tac Toe

Step1 : Create a 3×3 matrix and initilize all the elements to empty using '-'

Step2 : Create a function to check the winner

Step3 : And create 3 more function

1) check row
2) check column
3) check Diagonal.

if has the same value return winner
else none

Step4 : ~~Functon~~ Function to check whether the board is full or not
print : Draw
else : false

Step5 : Create another function ~~choice~~ called random.choice and main function tic_tac_toe()

S6:- If human plays turn ("X")
Prompt the player to enter row & column
else :
If the coumputer's turn ("O"), it randomly picks a row & column using Coumputer_function

S7: Call the respective function to declare whether the game is draw or succesful or error.

O		
1	1	
O		

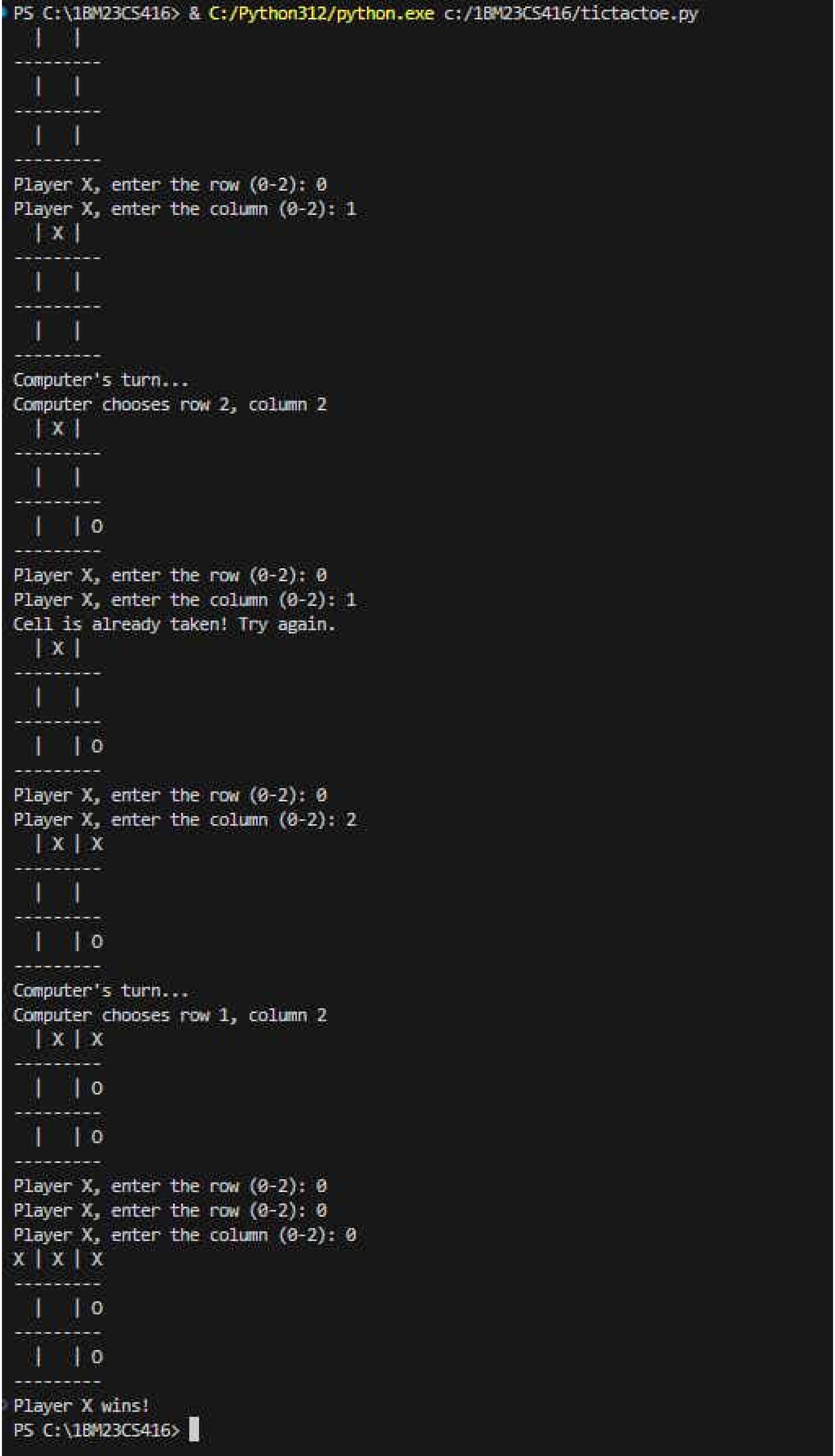
O	✓	✓
✓	.	.
✓	1.	1

```
import random
def print_board(board):
    for row in board:
        print(" |'.join(row))
        print("-" + 9)
def check_winner(board):
    ~~# Check~~
    for i in range(3):
        if board[i][0] == board[i][1] == board
                        [i][2] != " ";
            return board[i][0]
        if board[0][i] == board[1][i] == board[2][i]
                                != " ":
            return board[0][i]
    if board[0][0] == board[1][1] == board[2][2]
                            != " ":
        return board[0][0]
    if board[0][2] == board[1][1] == board[2][0]
                            != " ":
        return board[0][2]
    return None
def is_full(board):
    return all(cell != " " for row in board
                for cell in row)
def get_computer_move(board):
    empty_cells = [(i, j) for ~~j~~ i in range(3)
        for j in range(3) if board[i][j] == " "]
    return random.choice(empty_cells)
def tic_tac_toe():
    board = [[" " for _ in range(3)] for _ in rang
                                              (3)]
```

```
current_player = "x" .
computer_player = "O"
while True:
    print_board (board)
    if current_player == "x":
        row = int (input ("Player x, enter
                 the row (0-2): "))
        col = int (input ("Player x, enter
                 the column(0-2):"))
    else:
        print ("computer's turn...")
        row, col = get_computer_move(board)
        print (f"computer chooses row
          {row}, column {col}")

    if board [row][col] == " ":
        board [row][col] = current_player
    else:
        print (" cell is already taken!
                  Try again ")
        continue
    winner = check_winner (board)
    if winner:
        print_board (board)
        print (f" Player {winner} wins!")
        break
    if is_full (board):
        Print_board (board)
        print ("It's a tie !")
        break
    current_Player = computer_player
    if current-player == "x" else "x"
if __name__ == "__main__":
    tic_tac_toe()
```

Sneha



```
PS C:\1BM23CS416> & C:/Python312/python.exe c:/1BM23CS416/tictactoe.py
  |   |  
---------
  |   |  
---------
  |   |  
---------
Player X, enter the row (0-2): 0
Player X, enter the column (0-2): 1
  | X |  
---------
  |   |  
---------
  |   |  
---------
Computer's turn...
Computer chooses row 2, column 2
  | X |  
---------
  |   |  
---------
  |   | O
---------
Player X, enter the row (0-2): 0
Player X, enter the column (0-2): 1
Cell is already taken! Try again.
  | X |  
---------
  |   |  
---------
  |   | O
---------
Player X, enter the row (0-2): 0
Player X, enter the column (0-2): 2
  | X | X
---------
  |   |  
---------
  |   | O
---------
Computer's turn...
Computer chooses row 1, column 2
  | X | X
---------
  |   | O
---------
  |   | O
---------
Player X, enter the row (0-2): 0
Player X, enter the row (0-2): 0
Player X, enter the column (0-2): 0
X | X | X
---------
  |   | O
---------
  |   | O
---------
Player X wins!
PS C:\1BM23CS416> 
```