CSE PYTHON PROJECT

NAME :KHANPARA MIHIR SURESHBHAI

REG. NO. : 25BCE10458

PROJECT TOPIC : **TIC TAC TOE GAME**

PURPOSE : ENTERTAINMENT

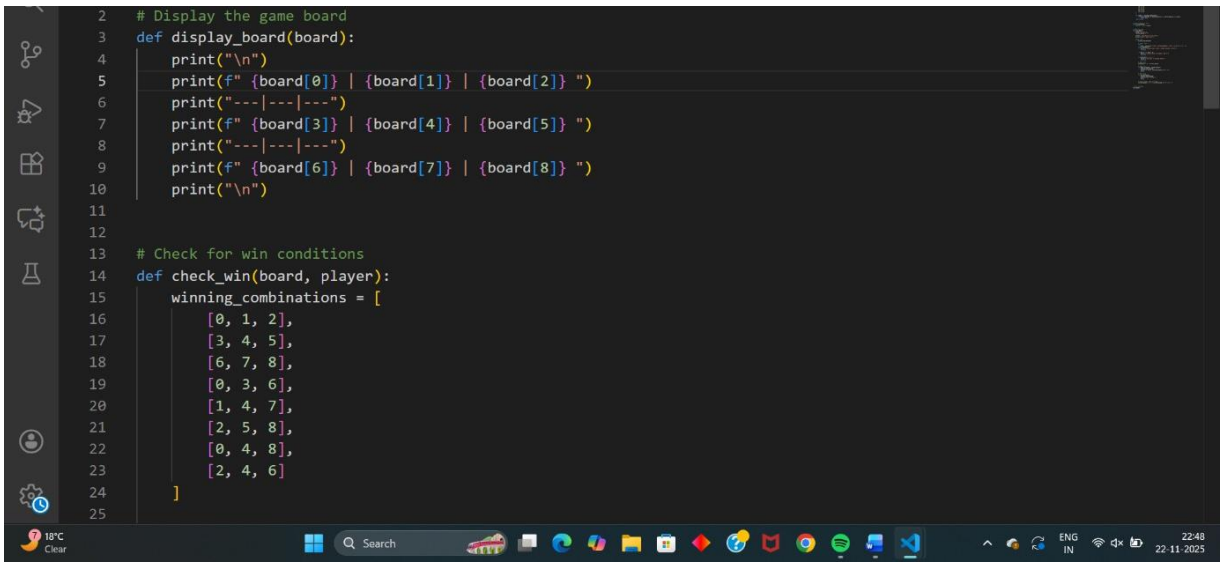
Here is a clear, simple, and project-ready Introduction for a Python Tic-Tac-Toe (Tik Tak) Game:

📌 Project Introduction: Tic-Tac-Toe Game in Python

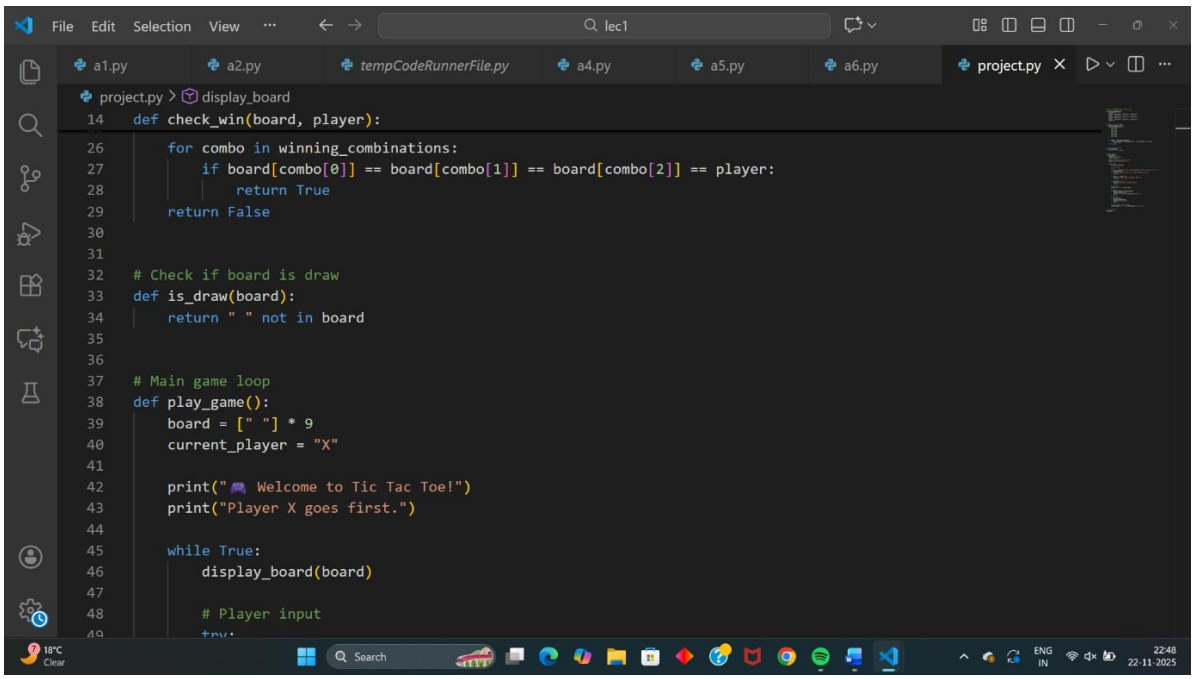
The Tic-Tac-Toe (Tik Tak) game is a classic two-player board game played on a 3×3 grid. The objective is simple: players take turns marking the grid with ‘X’ and ‘O’, and the first player to align three of their symbols horizontally, vertically, or diagonally wins the game.

This Python project aims to recreate the Tic-Tac-Toe experience using basic programming concepts. It helps beginners understand, conditionals, lists, functions loops, and user input handling. The game runs in the console and allows two players to compete by entering their moves through keyboard input.

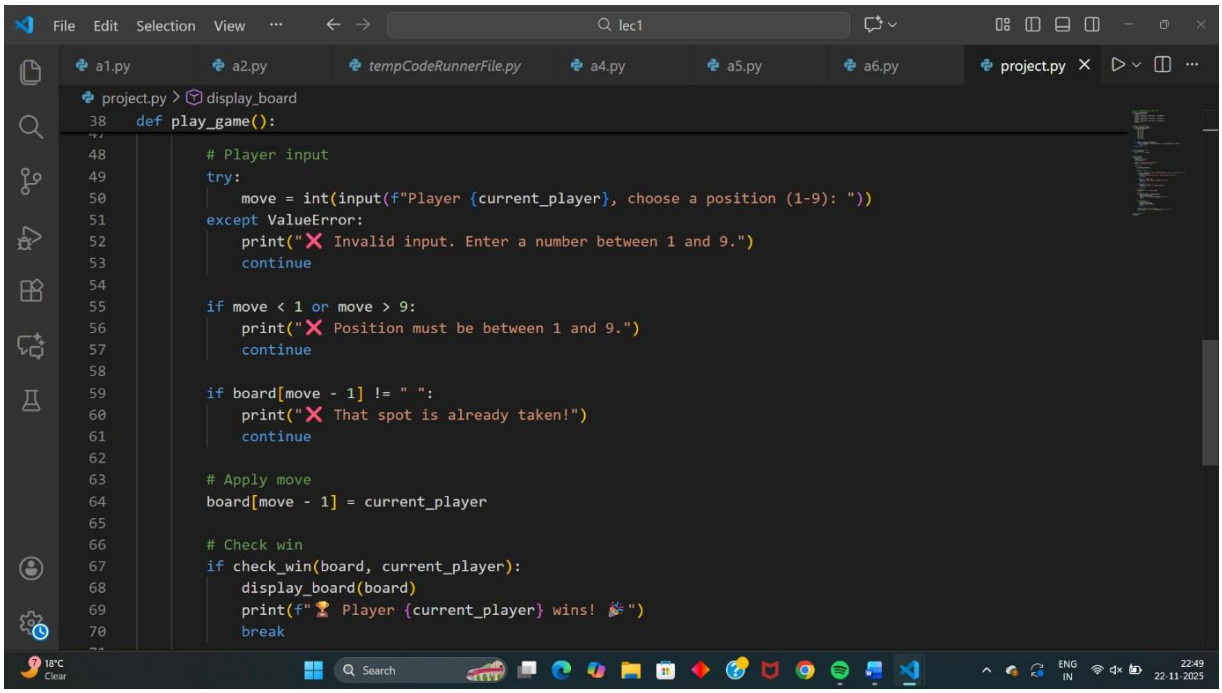
Developing this project improves problem-solving skills and introduces the idea of checking winning conditions, validating moves, and managing game flow. The result is a clean, interactive Python program that demonstrates core logic-building techniques in a fun way.

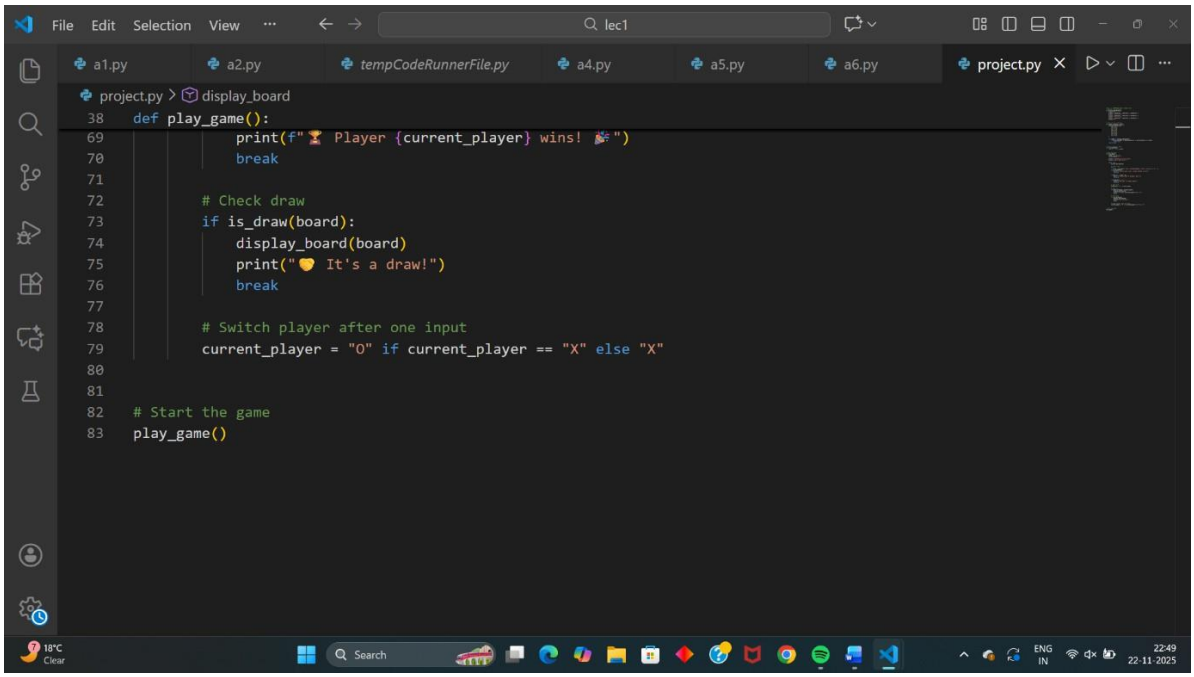


```python
# Display the game board
def display_board(board):
    print("\n")
    print(f" {board[0]} | {board[1]} | {board[2]} ")
    print("---|---|---")
    print(f" {board[3]} | {board[4]} | {board[5]} ")
    print("---|---|---")
    print(f" {board[6]} | {board[7]} | {board[8]} ")
    print("\n")


# Check for win conditions
def check_win(board, player):
    winning_combinations = [
        [0, 1, 2],
        [3, 4, 5],
        [6, 7, 8],
        [0, 3, 6],
        [1, 4, 7],
        [2, 5, 8],
        [0, 4, 8],
        [2, 4, 6]
    ]
```

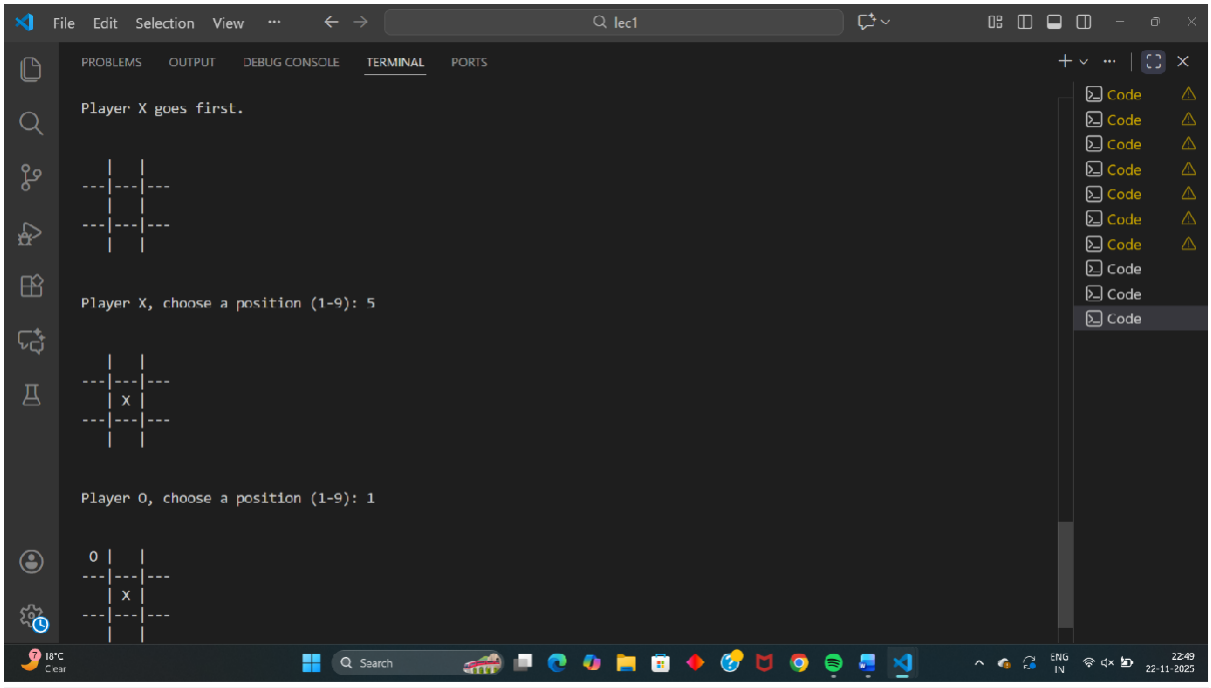


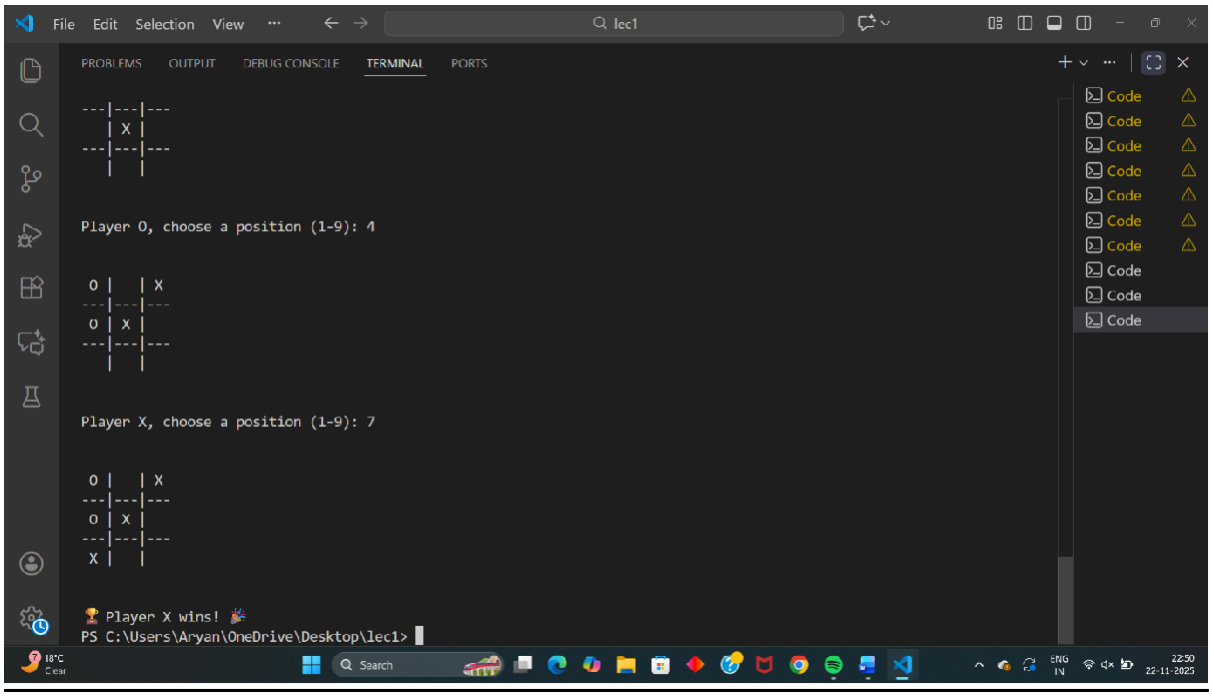
```python
def check_win(board, player):
    for combo in winning_combinations:
        if board[combo[0]] == board[combo[1]] == board[combo[2]] == player:
            return True
    return False


# Check if board is draw
def is_draw(board):
    return " " not in board


# Main game loop
def play_game():
    board = [" "] * 9
    current_player = "X"

    print("🎮 Welcome to Tic Tac Toe!")
    print("Player X goes first.")

    while True:
        display_board(board)

        # Player input
```


File Edit Selection View
lec1
a1.py
a2.py
tempCodeRunnerFile.py
a4.py
a5.py
a6.py
project.py
project.py > display_board
def play_game():
# Player input
try:
move = int(input(f"Player {current_player}, choose a position (1-9): "))
except ValueError:
print("❌ Invalid input. Enter a number between 1 and 9.")
continue
if move < 1 or move > 9:
print("❌ Position must be between 1 and 9.")
continue
if board[move - 1] != " ":
print("❌ That spot is already taken!")
continue
Apply move
board[move - 1] = current_player
# Check win
if check_win(board, current_player):
display_board(board)
print(f"🏆 Player {current_player} wins! 🎉")
break
Search
ENG IN
22:49
22-11-2025


File Edit Selection View
lec1
a1.py
a2.py
tempCodeRunnerFile.py
a4.py
a5.py
a6.py
project.py
project.py > display_board
def play_game():
print(f"🏆 Player {current_player} wins! 🎉")
break
# Check draw
if is_draw(board):
display_board(board)
print("🤝 It's a draw!")
break
# Switch player after one input
current_player = "O" if current_player == "X" else "X"
# Start the game
play_game()
Search
ENG IN
22:49
22-11-2025

OUTPUT:


File Edit Selection View
lec1
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Player X goes first.
Player X, choose a position (1-9): 5
Player O, choose a position (1-9): 1
Code
Search


File Edit Selection View
lec1
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
---|---|---
| X |
---|---|---
| |
Player O, choose a position (1-9): 4
O		X
O | X |
---|---|---
| |
Player X, choose a position (1-9): 7
O		X
O | X |
---|---|---
X | |
Player X wins!
PS C:\Users\Aryan\OneDrive\Desktop\lec1>
Code
Search
ENG
IN
22:50
22-11-2025