Experiment 2

Aim: To implement a Python-based Tic-Tac-Toe game.

Theory: Tic-Tac-Toe is a simple and popular game played on a 3x3 grid. Two players take turns marking spaces in the grid with 'X' and 'O'. The player who succeeds in placing three of their marks in a horizontal, vertical, or diagonal row wins the game. If all spaces are filled and no player has won, the game ends in a draw.

The game follows these rules:

1. The game is played on a 3x3 board.
2. Players take turns to place their marks ('X' or 'O').
3. The game continues until a player wins or the board is full.
4. A player wins if they align three marks in a row, column, or diagonal.

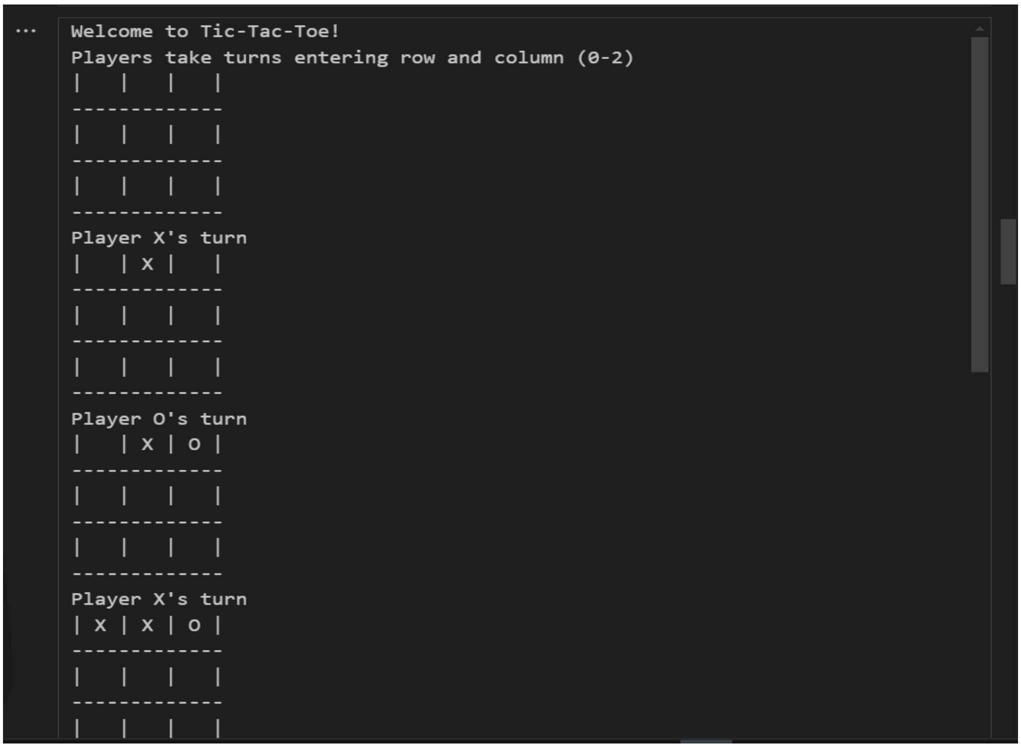
To implement this, we use Python with basic logic and control structures. The game board is represented as a list, and functions handle input validation, game logic, and result checking.

Code:

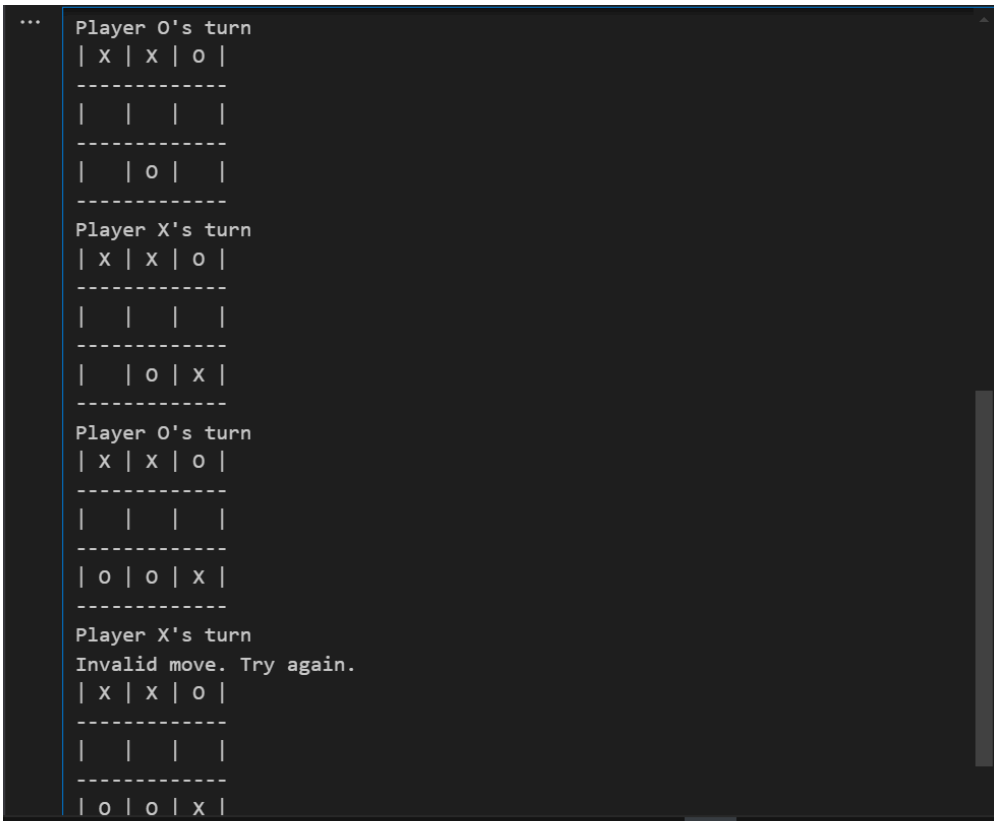
```
def print_board(board):
    for row in board:
        print(" | ".join(row))
        print("-" * 9)

def check_winner(board, player):
    for row in board:
        if all(cell == player for cell in row):
            return True

    for col in range(3):
        if all(board[row][col] == player for row in range(3)):
            return True

    if all(board[i][i] == player for i in range(3)) or all(board[i][2-i] == player for i in range(3)):
        return True

    return False

def is_draw(board):
    return all(cell != " " for row in board for cell in row)
```

```
def tic_tac_toe():
    board = [[" "] * 3 for _ in range(3)]
    players = ["X", "O"]
    turn = 0
    while True:
        print_board(board)
        row, col = map(int, input(f"Player {players[turn % 2]}, enter row and column (0-2):
").split()
        if board[row][col] != " ":
            print("Cell already occupied, try again.")
            continue
        board[row][col] = players[turn % 2]
        if check_winner(board, players[turn % 2]):
            print_board(board)
            print(f"Player {players[turn % 2]} wins!")
            break
        if is_draw(board):
            print_board(board)
            print("It's a draw!")
            break
        turn += 1
# Run the game
tic_tac_toe()
```

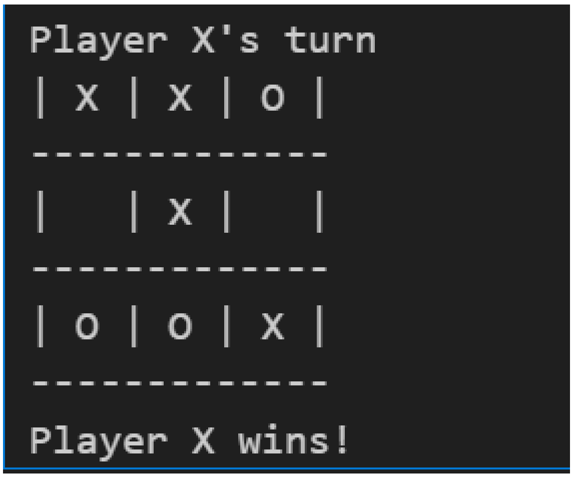
Output:



```
Welcome to Tic-Tac-Toe!
Players take turns entering row and column (0-2)
|   |   |   |
-------------
|   |   |   |
-------------
|   |   |   |
-------------
Player X's turn
|   | X |   |
-------------
|   |   |   |
-------------
|   |   |   |
-------------
Player O's turn
|   | X | O |
-------------
|   |   |   |
-------------
|   |   |   |
-------------
Player X's turn
| X | X | O |
-------------
|   |   |   |
-------------
|   |   |   |
```



```
Player O's turn
| x | x | o |
-------------
|   |   |   |
-------------
|   | o |   |
-------------
Player X's turn
| x | x | o |
-------------
|   |   |   |
-------------
|   | o | x |
-------------
Player O's turn
| x | x | o |
-------------
|   |   |   |
-------------
| o | o | x |
-------------
Player X's turn
Invalid move. Try again.
| x | x | o |
-------------
|   |   |   |
-------------
| o | o | x |
```



```
Player X's turn
| x | x | o |
-------------
|   | x |   |
-------------
| o | o | x |
-------------
Player X wins!
```

Conclusion: In this experiment, we successfully implemented a command-line Tic-Tac-Toe game in Python. The game allows two players to compete, checks for winning conditions, and declares the winner or a draw accordingly.