# TIC-TAC-TOE GAME

**Code**:

```
#include <iostream>
#include <vector>

using namespace std;

void displayBoard(const vector<char>& board) {
   cout << "\n";
   for (int i = 0; i < 9; i += 3) {
      cout << " " << board[i] << " | " << board[i + 1] << " | " << board[i + 2] << " \n";
      if (i < 6) cout << "---|---|---\n";
   }
   cout << "\n";
}

bool checkWin(const vector<char>& board, char player) {
   const int winPatterns[8][3] = {
      {0, 1, 2}, {3, 4, 5}, {6, 7, 8}, // Rows
      {0, 3, 6}, {1, 4, 7}, {2, 5, 8}, // Columns
      {0, 4, 8}, {2, 4, 6}          // Diagonals
   };
```

```
    for (auto& pattern : winPatterns) {
        if (board[pattern[0]] == player &&
            board[pattern[1]] == player &&
            board[pattern[2]] == player) {
            return true;
        }
    }
    return false;
}

bool checkDraw(const vector<char>& board) {
    for (char cell : board) {
        if (cell == ' ') return false;
    }
    return true;
}

void switchPlayer(char& currentPlayer) {
    currentPlayer = (currentPlayer == 'X') ? 'O' : 'X';
}

void resetBoard(vector<char>& board) {
    for (int i = 0; i < 9; ++i) {
        board[i] = ' ';
    }
}
```

```
void playTicTacToe() {
    vector<char> board(9, ' ');
    char currentPlayer = 'X';
    bool gameOver = false;

    while (!gameOver) {
        displayBoard(board);

        int move;
        cout << "Player " << currentPlayer << ", enter your move (1-9): ";
        cin >> move;

        if (move < 1 || move > 9 || board[move - 1] != ' ') {
            cout << "Invalid move. Try again.\n";
            continue;
        }

        board[move - 1] = currentPlayer;

        if (checkWin(board, currentPlayer)) {
            displayBoard(board);
            cout << "Player " << currentPlayer << " wins!\n";
            gameOver = true;
        } else if (checkDraw(board)) {
            displayBoard(board);
```

```
            cout << "The game is a draw!\n";
            gameOver = true;
        } else {
            switchPlayer(currentPlayer);
        }
    }

    char playAgain;
    cout << "Do you want to play again? (y/n): ";
    cin >> playAgain;

    if (playAgain == 'y' || playAgain == 'Y') {
        resetBoard(board);
        playTicTacToe();
    } else {
        cout << "Thanks for playing!\n";
    }
}

int main() {
    cout << "Welcome to Tic-Tac-Toe!\n";
    playTicTacToe();
    return 0;
}
```

Output:

Welcome to Tic-Tac-Toe!

```
  |   |
---|---|---
  |   |
---|---|---
  |   |
```

Player X, enter your move (1-9): 5

```
  |   |
---|---|---
  | X |
---|---|---
  |   |
```

Player O, enter your move (1-9): 2

```
  | O |
---|---|---
  | X |
---|---|---
```

```
  |   |

Player X, enter your move (1-9): 4

  | O |
---|---|---
X | X |
---|---|---
  |   |

Player O, enter your move (1-9): 8

  | O |
---|---|---
X | X |
---|---|---
  | O |

Player X, enter your move (1-9): 6

  | O |
---|---|---
X | X | X
---|---|---
  | O |
```

Player X wins!

Do you want to play again? (y/n): y

```
  |   |
---|---|---
  |   |
---|---|---
  |   |
```

Player X, enter your move (1-9): 5

```
  |   |
---|---|---
  | X |
---|---|---
  |   |
```

Player O, enter your move (1-9): 3

```
  |   | O
---|---|---
  | X |
---|---|---
  |   |
```

Player X, enter your move (1-9): 2

```
  | X | O
---|---|---
  | X |
---|---|---
  |   |
```

Player O, enter your move (1-9): 3

Invalid move. Try again.

```
  | X | O
---|---|---
  | X |
---|---|---
  |   |
```

Player O, enter your move (1-9): 6

```
  | X | O
---|---|---
  | X | O
---|---|---
  |   |
```

Player X, enter your move (1-9): 8

```
  | X | O
---|---|---
  | X | O
---|---|---
  | X |


Player X wins!
Do you want to play again? (y/n): n
Thanks for playing!


=== Code Execution Successful ===
```