```
#TheekThaakToe

#Structure
box1=[['','',''],['','',''],['','','']]
box2=[['','',''],['','',''],['','','']]
box3=[['','',''],['','',''],['','','']]
box4=[['','',''],['','',''],['','','']]
box5=[['','',''],['','',''],['','','']]
box6=[['','',''],['','',''],['','','']]
box7=[['','',''],['','',''],['','','']]
box8=[['','',''],['','',''],['','','']]
box9=[['','',''],['','',''],['','','']]

matrix=[box1,box2,box3,box4,box5,box6,box7,box8,box9]
scoreCard={}
#Defining a function to display the matrix
def display():
    print(21*'--')
    for i in range(0,8,3):
        for j in range (0,3):
            for m in range(0,3):
                print('|',end='')
                for k in range(0,3):
                    if matrix[i+m][j][k]=='':
                        print(matrix[i+m][j][k],end='  |')
                    else:
                        print(matrix[i+m][j][k],end=' |')
                print('\t',end='')
            if j<2:
                print()
                print(21*'- ')
        print()
        print(21*'--')
    print('X: ',list(scoreCard.values()).count('X'),end='\t')
    print('O: ',list(scoreCard.values()).count('O'))

#Defining a function to check 3 chars in a row
def checkwin(box):
    if (box[0]+box[1]+box[2]).count(char)>=3:
        if box[0][0]==box[0][1]==box[0][2]!='':
            return True
        elif box[1][0]==box[1][1]==box[1][2]!='':
            return True
```

```
        elif box[2][0]==box[2][1]==box[2][2]!='':
            return True
        elif box[0][0]==box[1][0]==box[2][0]!='':
            return True
        elif box[0][1]==box[1][1]==box[2][1]!='':
            return True
        elif box[0][2]==box[1][2]==box[2][2]!='':
            return True
        elif box[0][0]==box[1][1]==box[2][2]!='':
            return True
        elif box[0][2]==box[1][1]==box[2][0]!='':
            return True
        else:
            return False
    else:
        return False

print('Hello','\n','Rules: To play press the key on your num pad corresponding to the box')
display()
box_no=4
turn_no=0

#Actual GAME
while True:
    try:
        num=int(input('Enter  '))
    except:
        print('ERROR')
        break
    if num not in range(1,10):
        print('Number not within 1 and 9. Try Again')
        continue
    if num in (7,8,9):
        num-=6
    elif num in (1,2,3):
        num+=6
    if turn_no%2==0:
        char='X'
    else:
        char='O'
    if matrix[box_no][(num-1)//3][(num-1)%3] ==':
        matrix[box_no][(num-1)//3][(num-1)%3]=char
    else:
```

```
        print('Box already occupied')
        continue
    display()
    if box_no+1 not in scoreCard and checkwin(matrix[box_no]):
        print(char,'Wins this box')
        scoreCard[box_no+1]=char
        if list(scoreCard.values()).count(char)==5:
            print(char,' Wins Congrats')
            break
    box_no=num-1
    turn_no+=1
```

```
#Regular Tic-Tac-Toe Code
import random
#list ----> prints
#displays the board in the form of a tic tac toe board
def displaygrid(board):
    print("-------------")
    print("|{0:^3}|{1:^3}|{2:^3}|".format(board[6],board[7],board[8]))
    print("-------------")
    print("|{0:^3}|{1:^3}|{2:^3}|".format(board[3],board[4],board[5]))
    print("-------------")
    print("|{0:^3}|{1:^3}|{2:^3}|".format(board[0],board[1],board[2]))
    print("-------------")

#no parameter ----> returns nothing
#keeps asking till the player gets ready
def are_you_ready():
    is_ready = 'n'
    while is_ready.lower() != 'y':
        is_ready = input("\nAre you ready to begin playing? (Y/N): ")
        if is_ready.lower() != 'y':
```

```
            print("No Problem! I will wait :)\n")

#no parameter ----> list
#asks the first player for the symbol [X/O] it wants to use
#and then returns a list with player 1 symbol at list[0]
def receive_signs():
    sign = input("Player 1: Will you play with 'X' or 'O'?: ")
    if sign.lower() == 'x':
        return ['X', 'O']
    elif sign.lower() == 'o':
        return ['O', 'X']

#int, list, string ----> int
#takes in the player no. [0/1], the board and player symbol
#and returns the position where it wants to place the symbol.
#Note: Keeps asking for position till the player chooses an empty slot.
def take_input(player, board, sign):
    while True:
        pos = int(input("Player {}: Choose an empty slot for your '{}' [1-9]: ".format(player, sign)))
        if 1 <= pos <= 9 and board[pos-1] == '':
            break
    return pos

#list ----> boolean
#takes in the moves list and determines if
#any of the player has won the game or not.
def somebody_wins(board):
    if board[0] == board[1] == board[2] and board[0] != '':
        return True
    elif board[3] == board[4] == board[5] and board[3] != '':
        return True
    elif board[6] == board[7] == board[8] and board[6] != '':
        return True
    elif board[0] == board[3] == board[6] and board[0] != '':
        return True
    elif board[1] == board[4] == board[7] and board[1] != '':
        return True
    elif board[2] == board[5] == board[8] and board[2] != '':
        return True
    elif board[0] == board[4] == board[8] and board[0] != '':
        return True
    elif board[2] == board[4] == board[6] and board[2] != '':
        return True
```

```
    return False

#list, int ----> boolean
#determines a draw in the game
def draw(board, turn_no):
    for item in board:
        if item == '':
            return False
    return True

#int, string, list ----> list
#takes the desired position, symbol and the board
#and places the players symbol on that position in the
#list and finnaly returns the new list
def modify_grid(pos, sign, board):
    board[pos-1] = sign
    return board

#int ----> int
#determines the winner
def who_wins(turn_no):
    if turn_no % 2 == 0:
        return 1
    return 2


#flow of game
def mainFunction():
    play_more = 'y'
    while play_more.lower() == 'y':
        print("\n"*50)
        print("************************")
        print("*Welcome to Tic Tac Toe*")
        print("************************")
        #contains the status of moves
        board = ['']*9

        #sign_list[0] will contain the symbol chosen by player 1
        #sign_list[1] will contain the symbol given to player 2
        sign_list = receive_signs()

        are_you_ready()
        print("\n"*100)
```

```
    turn_no = random.randint(0,1)
    print('Congratulations Player {}, You get to make the first turn'.format(turn_no+1))
    while not somebody_wins(board)  and  not draw(board, turn_no) :
        player_info = {
            0: (1, sign_list[0]),
            1: (2, sign_list[1])
        }
        player_no, sign = player_info[turn_no % 2]
        pos = take_input(player_no,board, sign)
        board = modify_grid(pos, sign, board)
        print("\n"*100)
        displaygrid(board)
        turn_no += 1
    print("\n" *100)
    displaygrid(board)
    if somebody_wins(board):
        wins = who_wins(turn_no-1)
        print("Congratulations! Player {} wins the game !".format(wins))
    else:
        print("Match Tied!")
        print("Well played both the players!")
    play_more = input(("Do you guys want to play again? [Y/N]: "))

mainFunction()
```