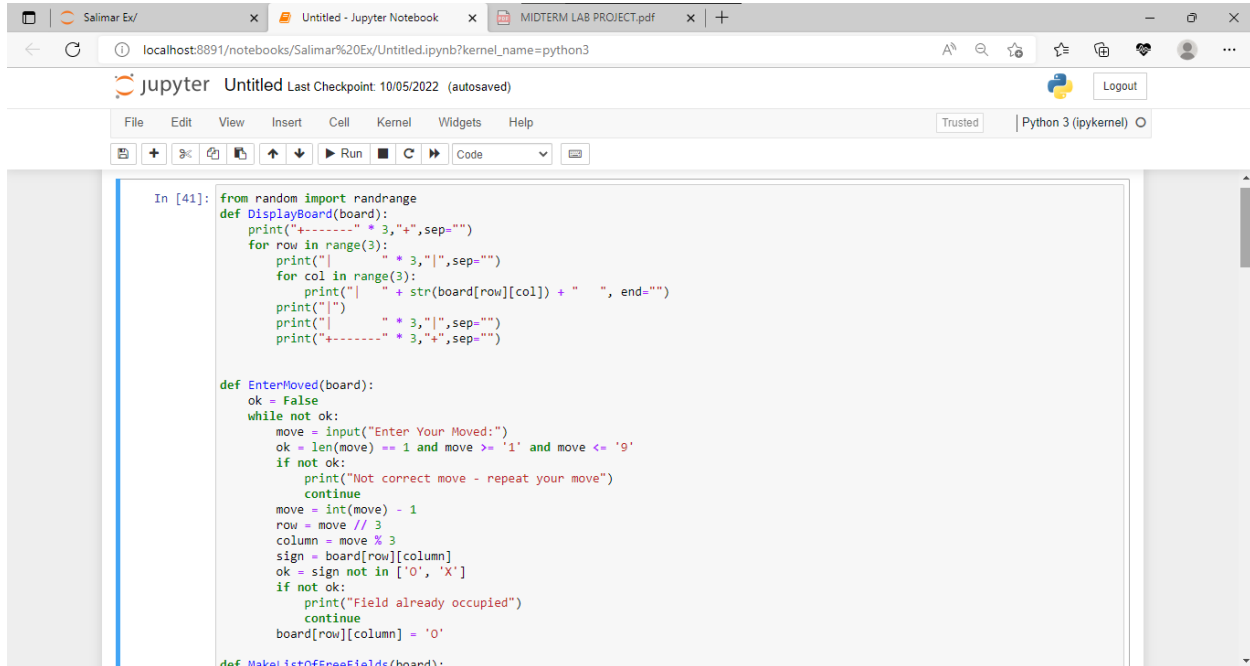


Marvin B. Waro | BSCS 3B

– CS ELECTIVE 2 LABORATORY MIDTERM PROJECT: TIC-TAC-TOE

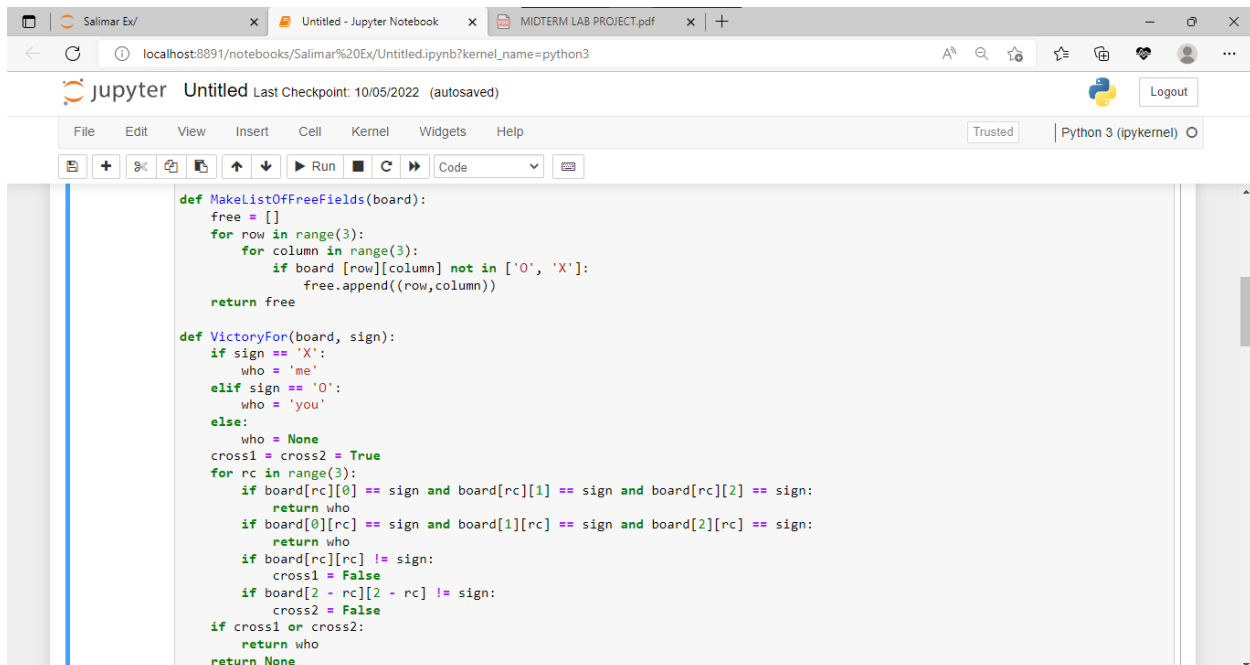
SCREENSHOTS IN CHRONOLOGICAL ORDER:



```
In [41]: from random import randrange
def DisplayBoard(board):
    print("+-----" * 3, "+", sep="")
    for row in range(3):
        print("|" * 3, "|", sep="")
        for col in range(3):
            print("|" + str(board[row][col]) + " ", end="")
        print("|")
    print("+-----" * 3, "+", sep="")

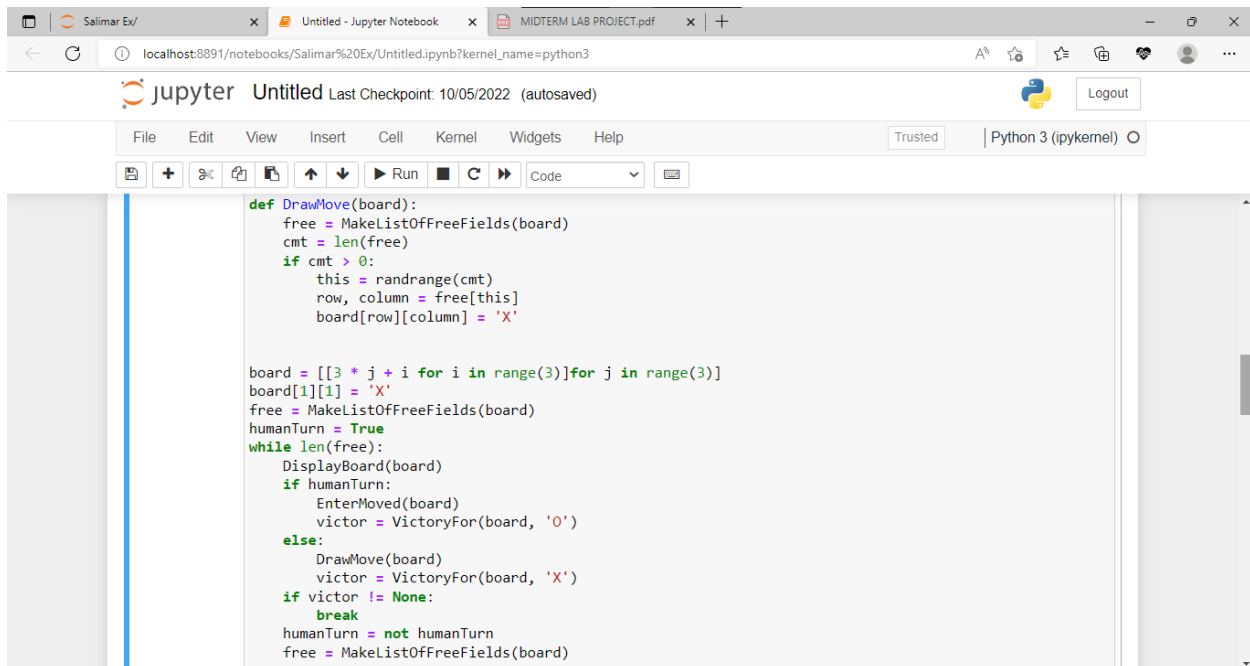
def EnterMoved(board):
    ok = False
    while not ok:
        move = input("Enter Your Moved:")
        ok = len(move) == 1 and move >= '1' and move <= '9'
        if not ok:
            print("Not correct move - repeat your move")
            continue
        move = int(move) - 1
        row = move // 3
        column = move % 3
        sign = board[row][column]
        ok = sign not in ['O', 'X']
        if not ok:
            print("Field already occupied")
            continue
        board[row][column] = 'O'

def MakeListOfFreeFields(board):
```



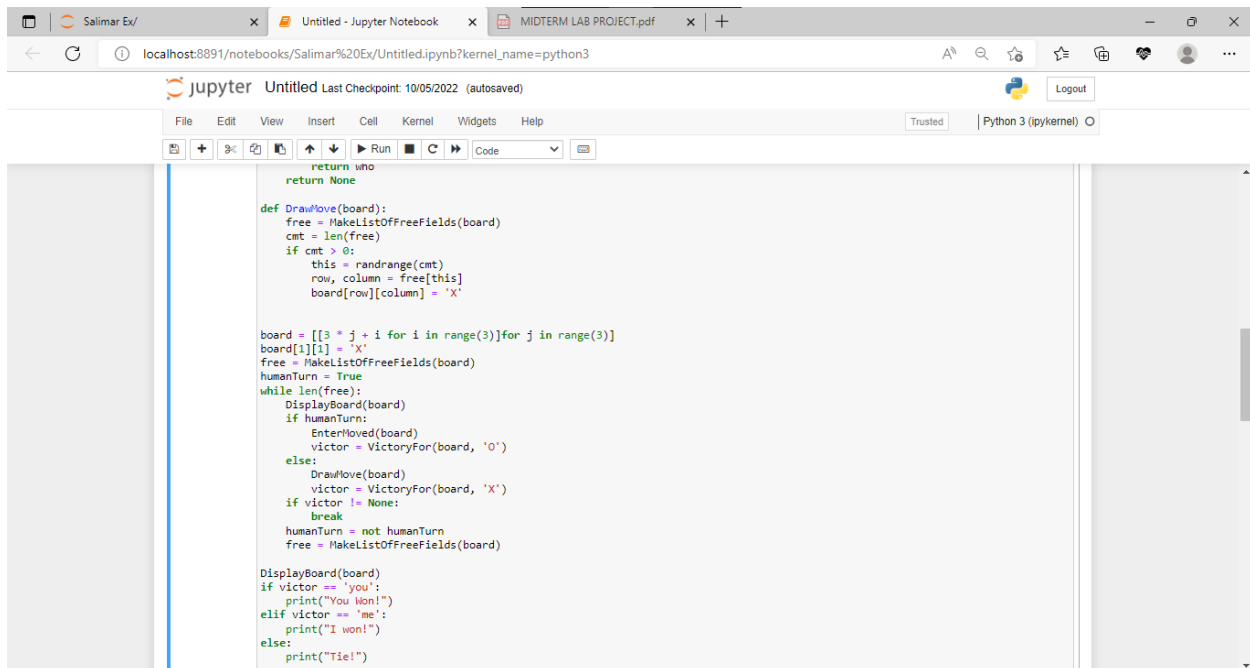
```
def MakeListOfFreeFields(board):
    free = []
    for row in range(3):
        for column in range(3):
            if board[row][column] not in ['O', 'X']:
                free.append((row, column))
    return free

def VictoryFor(board, sign):
    if sign == 'X':
        who = 'me'
    elif sign == 'O':
        who = 'you'
    else:
        who = None
    cross1 = cross2 = True
    for rc in range(3):
        if board[rc][0] == sign and board[rc][1] == sign and board[rc][2] == sign:
            return who
        if board[0][rc] == sign and board[1][rc] == sign and board[2][rc] == sign:
            return who
        if board[rc][rc] != sign:
            cross1 = False
        if board[2 - rc][2 - rc] != sign:
            cross2 = False
    if cross1 or cross2:
        return who
    return None
```



```
def DrawMove(board):
    free = MakeListOfFreeFields(board)
    cmt = len(free)
    if cmt > 0:
        this = randrange(cmt)
        row, column = free[this]
        board[row][column] = 'X'

board = [[3 * j + i for i in range(3)] for j in range(3)]
board[1][1] = 'X'
free = MakeListOfFreeFields(board)
humanTurn = True
while len(free):
    DisplayBoard(board)
    if humanTurn:
        EnterMoved(board)
        victor = VictoryFor(board, 'O')
    else:
        DrawMove(board)
        victor = VictoryFor(board, 'X')
    if victor != None:
        break
    humanTurn = not humanTurn
    free = MakeListOfFreeFields(board)
```



```
return None
return None

def DrawMove(board):
    free = MakeListOfFreeFields(board)
    cmt = len(free)
    if cmt > 0:
        this = randrange(cmt)
        row, column = free[this]
        board[row][column] = 'X'

board = [[3 * j + i for i in range(3)] for j in range(3)]
board[1][1] = 'X'
free = MakeListOfFreeFields(board)
humanTurn = True
while len(free):
    DisplayBoard(board)
    if humanTurn:
        EnterMoved(board)
        victor = VictoryFor(board, 'O')
    else:
        DrawMove(board)
        victor = VictoryFor(board, 'X')
    if victor != None:
        break
    humanTurn = not humanTurn
    free = MakeListOfFreeFields(board)

DisplayBoard(board)
if victor == 'you':
    print("You Won!")
elif victor == 'me':
    print("I Won!")
else:
    print("Tie!")
```


Salimar Ex/ x Untitled - Jupyter Notebook x MIDTERM LAB PROJECT.pdf x +

localhost:8891/notebooks/Salimar%20Ex/Untitled.ipynb?kernel_name=python3

jupyter Untitled Last Checkpoint: 10/05/2022 (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Enter Your Moved:2

0	0	2
3	X	5
6	X	8
0	0	2
X	X	5
6	X	8

Enter Your Moved:3

Salimar Ex/ x Untitled - Jupyter Notebook x MIDTERM LAB PROJECT.pdf x +

localhost:8891/notebooks/Salimar%20Ex/Untitled.ipynb?kernel_name=python3

jupyter Untitled Last Checkpoint: 10/05/2022 (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

6	X	8
0	0	0
X	X	5
6	X	8

You Won!

In []: