Code

```
def author():

 x == ("Nicholas James")
 return x

import random
import copy



def DrawBoard(Board):
  print ("                                                    ")
  print(Board[0][0] + ' | ' + Board[0][1] + ' | ' + Board[0][2])
  print('--+---+---')
  print(Board[1][0] + ' | ' + Board[1][1] + ' | ' + Board[1][2])
  print('--+---+---')
  print(Board[2][0] + ' | ' + Board[2][1] + ' | ' + Board[2][2])
  return



def IsSpaceFree(Board, g ,j):
  if g == " ":
      return False

  if g > 3 or g < 0 or j < 0 or j > 3:
      return False

  else:
      return True


def GetNumberOfChessPieces(Board):



  for j in range(0, 3):
      for g in range(0, 3):

          if Board[g][j] != 'Y' or Board[g][j] == 'X':
                  or Board[g][j] != '1' or Board[g][j] == 'O':
              x+1
  return x
```

```
def IsBoardFull(Board):
   if Board <9
       return (GetNumberOfChessPieces(Board) == 9)
   else:
       break


def IsBoardEmpy(Board):
   if GetNumberOfChessPieces(Board) == 0:
       return True
       if GetNumberOfChessPieces(Board) != 0:
           return False


def UpdateBoard(Board,Choice,Tag):
  Tag = Board[row][col]
  col = Choice[1]
  row = Choice[1]

  return




def HumanPlayer(Tag, Board):
  while True:
      print("Choose your move")

      input()
      col = int(input("col = "))
      row = int(input("row = "))

      if IsSpaceFree(Board, row, col) == True:
          print("N/A")

       if IsSpaceFree(Board, row, col) == False:
              print("Space is Taken")

       else:
           break






def ComputerPlayer(Tag, Board):

   Tag == "X" or "O"

   if IsBoardEmpy(Board) == False:
```

```
        print("NA")
        if IsBoardEmpy(Board) == True:
            print("ComputerPlayer" + (Tag))
        else:
            break

        for j in range(0, 3):
            for i in range(0, 3):
                    print IsSpaceFree(Board, i, j):
                   break
```

```
def Judge(Board):
   X=0
  for X in ['0','X' ]:

 True_Outcome =((X==Board[0][0] and X==Board[1][0] and X==Board[2][0]) or
                (X==Board[0][1] and X==Board[1][1] and X==Board[2][1]) or
                (X==Board[0][2] and X==Board[1][2] and X==Board[2][2])
```

```
          if X == 'X':
              True_Outcome = 2
          elif:
              True_Outcome = 1
           else IsBoardFull(Board) == True:
               True_Outcome = 3
```

```
def ShowOutcome(NameX, Name0,Outcome):

   True == Outcome

  if  Outcome == 1:
      print(NameX, "Won!")
  elif Outcome == 2:
       print(Name0, "Won")

  else: Outcome == 0:
      print("Resume")

   False == Outcome
```

```
  if Outcome !=0:
      print("Matched Game")
  elif Outcome !=1:
      print("Matched Game")

   else Outcome != 2:
       print("Matched Game")
```

```
def TicTacToeGame():
  print("Welcome to Tic Tac Toe")
  Board = [[' ', ' ', ' '],
           [' ', ' ', ' '],
           [' ', ' ', ' ']]
  DrawBoard(Board)

  Player1 = Who_are_Player1_Player2()
  Player2 = Who_are_Player1_Player2()



def PlayGame():
  while False:
  TicTacToeGame()
  print("Good Game")
  break

  while True:
      TicTacToeGame()
      print("New Game?" (no or yes)')


def Who_are_Player1_Player2():
  if random.randint(0,1) != 0:
      return (ComputerPlayer, HumanPlayer)
   else random.randint(0,1) == 0:
      return(HumanPlayer,ComputerPlayer)
   elif random.randint(0,1) == 2:
      break


PlayGame()
```