

```
1 import tkinter as tk
2 from tkinter import messagebox
3
4
5 class TicTacToe:
6     def __init__(self, root):
7         self.root = root
8         self.root.title("Tic Tac Toe")
9         self.reset_game()
10        self.create_widgets()
11
12    def reset_game(self):
13        self.board = [["_ " for _ in range(3)]
14        for _ in range(3)]
15        self.current_player = "X"
16        self.game_over = False
17
18    def create_widgets(self):
19        self.buttons = [[None for _ in range(
20        3)] for _ in range(3)]
21        for i in range(3):
22            for j in range(3):
23                button = tk.Button(self.root
24                , text="", font=('normal', 40), width=5,
25                height=2,
26                command=
27                lambda row=i, col=j: self.on_click(row, col))
28                button.grid(row=i, column=j)
29                self.buttons[i][j] = button
30
31        self.reset_button = tk.Button(self.
32        root, text="Reset", font=('normal', 20),
33        command=self.reset_board)
34        self.reset_button.grid(row=3,
35        columnspan=3)
```

```
28
29     def on_click(self, row, col):
30         if not self.game_over and self.board[
row][col] == "":
31             self.board[row][col] = self.
current_player
32             self.buttons[row][col].config(
text=self.current_player)
33             if self.check_winner():
34                 self.game_over = True
35                 messagebox.showinfo("Tic Tac
Toe", f"Player {self.current_player} wins!")
36             elif self.check_draw():
37                 self.game_over = True
38                 messagebox.showinfo("Tic Tac
Toe", "It's a draw!")
39             else:
40                 self.current_player = "0" if
self.current_player == "X" else "X"
41
42     def check_winner(self):
43         for row in self.board:
44             if row[0] == row[1] == row[2] !=
"":
45                 return True
46         for col in range(3):
47             if self.board[0][col] == self.
board[1][col] == self.board[2][col] != "":
48                 return True
49             if self.board[0][0] == self.board[1][
1] == self.board[2][2] != "":
50                 return True
51             if self.board[0][2] == self.board[1][
1] == self.board[2][0] != "":
52                 return True
```

```
53         return False
54
55     def check_draw(self):
56         for row in self.board:
57             for cell in row:
58                 if cell == "":
59                     return False
60         return True
61
62     def reset_board(self):
63         self.reset_game()
64         for i in range(3):
65             for j in range(3):
66                 self.buttons[i][j].config(
67                     text="")
68
69 if __name__ == "__main__":
70     root = tk.Tk()
71     app = TicTacToe(root)
72     root.mainloop()
73
```