Lab 21

Name : Shashank Bagda **Date : 19 / 11 / 22**

Enrollment No : 92100133020

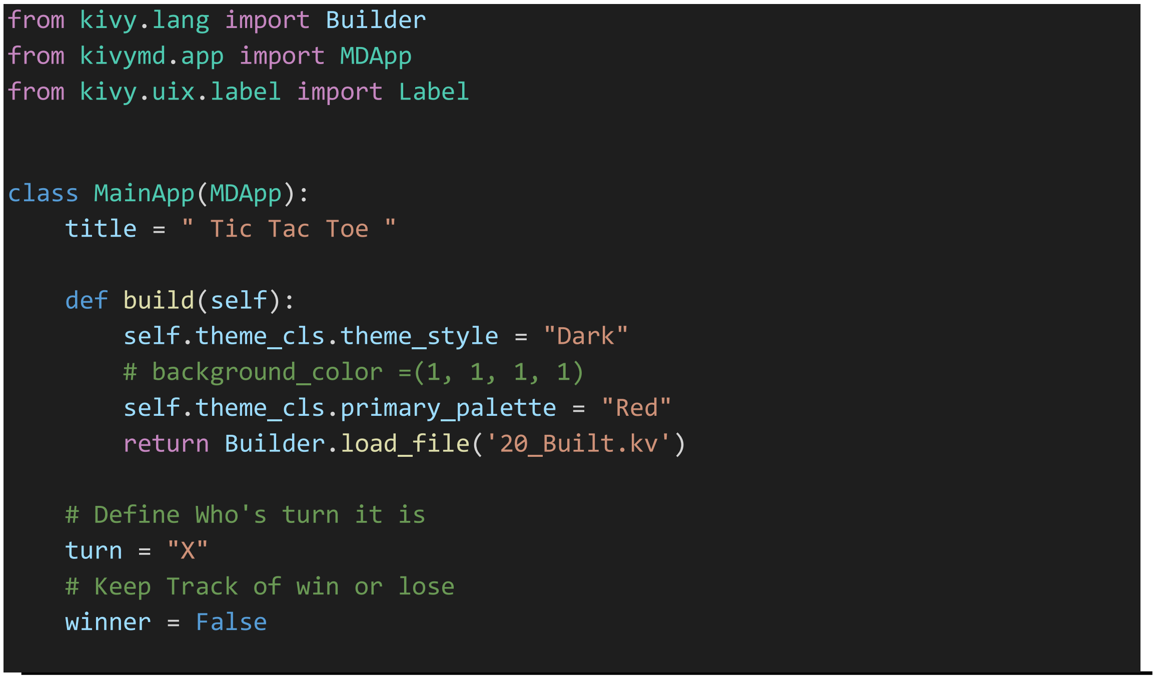
: CODE :

Library Installation
pip install kivy
pip install kivymd

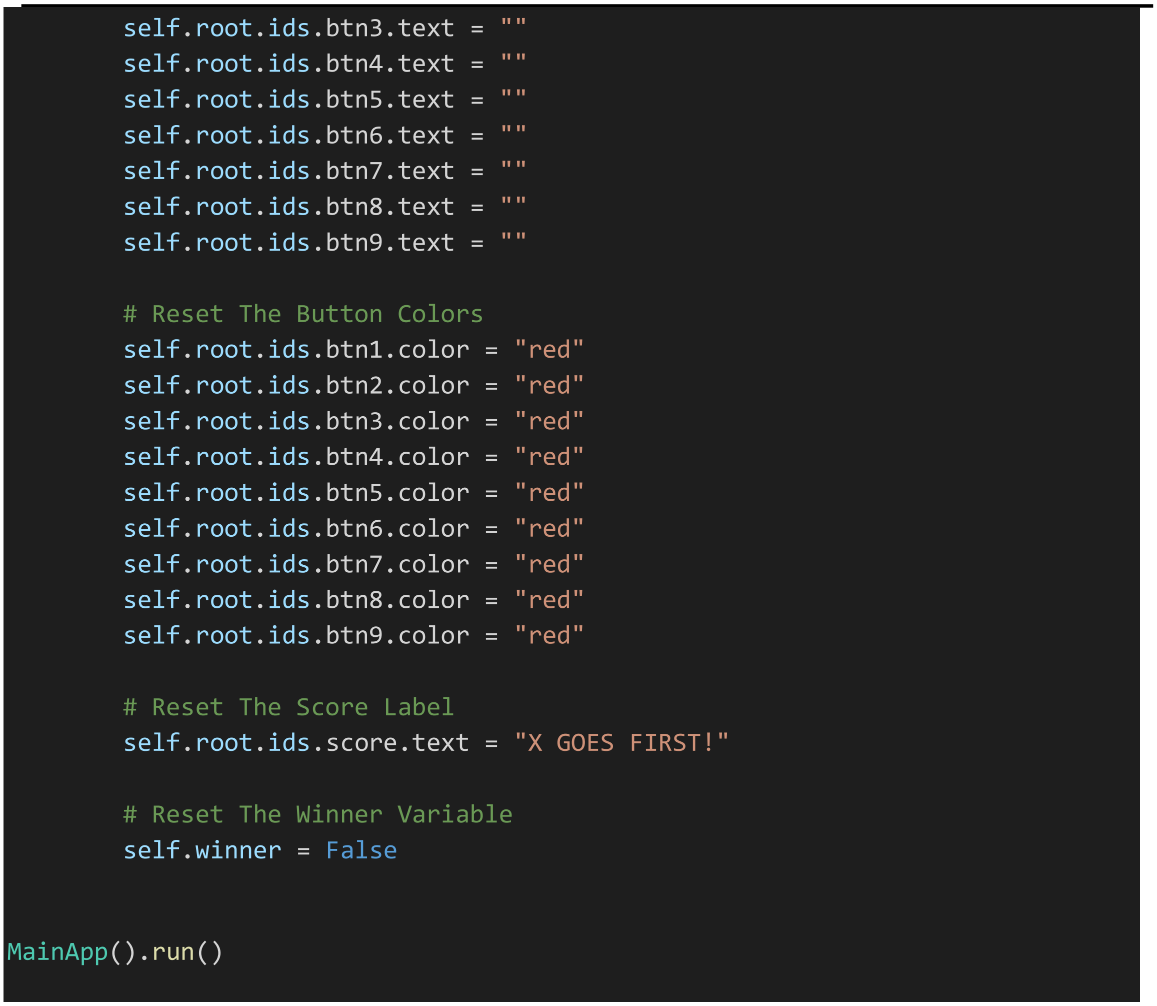
Reference Link: https://www.edureka.co/blog/kivy-tutorial/
https://kivycoder.com/build-a-tic-tac-toe-game-part-1-python-kivy-gui-tutorial-63/

1. Write a GUI program create Tic-tac-toe in python

Python code:



```python
from kivy.lang import Builder
from kivymd.app import MDApp
from kivy.uix.label import Label


class MainApp(MDApp):
    title = " Tic Tac Toe "

    def build(self):
        self.theme_cls.theme_style = "Dark"
        # background_color =(1, 1, 1, 1)
        self.theme_cls.primary_palette = "Red"
        return Builder.load_file('20_Built.kv')

    # Define Who's turn it is
    turn = "X"
    # Keep Track of win or lose
    winner = False
    # Keep track of winners and losers
    X_win = 0
    O_win = 0

    # No Winner
    def no_winner(self):
        if self.winner == False and \
                self.root.ids.btn1.disabled == True and \
                self.root.ids.btn2.disabled == True and \
                self.root.ids.btn3.disabled == True and \
                self.root.ids.btn4.disabled == True and \
                self.root.ids.btn5.disabled == True and \
                self.root.ids.btn6.disabled == True and \
                self.root.ids.btn7.disabled == True and \
                self.root.ids.btn8.disabled == True and \
                self.root.ids.btn9.disabled == True:
            self.root.ids.score.text = " IT'S A TIE "

    # End The Game
    def end_game(self, a, b, c):
        self.winner = True
        a.color = "green"
        b.color = "green"
        c.color = "green"

        # Disable the buttons
        self.disable_all_buttons()

        # Set Label for winner
        self.root.ids.score.text = f"{a.text} Wins!"

        # Keep track of winners and loser
        if a.text == "X":
            self.X_win = self.X_win + 1
        else:
            self.O_win = self.O_win + 1

        self.root.ids.game.text = f"X Wins = {self.X_win}  |  O Wins = 
{self.O_win}"

    def disable_all_buttons(self):
        # Disable The Buttons
        self.root.ids.btn1.disabled = True
        self.root.ids.btn2.disabled = True
        self.root.ids.btn3.disabled = True
        self.root.ids.btn4.disabled = True
        self.root.ids.btn5.disabled = True
        self.root.ids.btn6.disabled = True
        self.root.ids.btn7.disabled = True
        self.root.ids.btn8.disabled = True
        self.root.ids.btn9.disabled = True

    def win(self):
        # Across
        if self.root.ids.btn1.text != "" and self.root.ids.btn1.text ==
self.root.ids.btn2.text and self.root.ids.btn2.text ==
self.root.ids.btn3.text:
            self.end_game(self.root.ids.btn1,
                          self.root.ids.btn2, self.root.ids.btn3)

        if self.root.ids.btn4.text != "" and self.root.ids.btn4.text ==
self.root.ids.btn5.text and self.root.ids.btn5.text ==
self.root.ids.btn6.text:
            self.end_game(self.root.ids.btn4,
                          self.root.ids.btn5, self.root.ids.btn6)

        if self.root.ids.btn7.text != "" and self.root.ids.btn7.text ==
self.root.ids.btn8.text and self.root.ids.btn8.text ==
self.root.ids.btn9.text:
            self.end_game(self.root.ids.btn7,
                          self.root.ids.btn8, self.root.ids.btn9)
        # Down
        if self.root.ids.btn1.text != "" and self.root.ids.btn1.text ==
self.root.ids.btn4.text and self.root.ids.btn4.text ==
self.root.ids.btn7.text:
            self.end_game(self.root.ids.btn1,
                          self.root.ids.btn4, self.root.ids.btn7)

        if self.root.ids.btn2.text != "" and self.root.ids.btn2.text ==
self.root.ids.btn5.text and self.root.ids.btn5.text ==
self.root.ids.btn8.text:
            self.end_game(self.root.ids.btn2,
                          self.root.ids.btn5, self.root.ids.btn8)

        if self.root.ids.btn3.text != "" and self.root.ids.btn3.text ==
self.root.ids.btn6.text and self.root.ids.btn6.text ==
self.root.ids.btn9.text:
            self.end_game(self.root.ids.btn3,
                          self.root.ids.btn6, self.root.ids.btn9)

        # Diagonal
        if self.root.ids.btn1.text != "" and self.root.ids.btn1.text ==
self.root.ids.btn5.text and self.root.ids.btn5.text ==
self.root.ids.btn9.text:
            self.end_game(self.root.ids.btn1,
                          self.root.ids.btn5, self.root.ids.btn9)

        if self.root.ids.btn3.text != "" and self.root.ids.btn3.text ==
self.root.ids.btn5.text and self.root.ids.btn5.text ==
self.root.ids.btn7.text:
            self.end_game(self.root.ids.btn3,
                          self.root.ids.btn5, self.root.ids.btn7)

        self.no_winner()

    def presser(self, btn):
        if self.turn == 'X':
            btn.text = "X"
            btn.disabled = True
            self.root.ids.score.text = "O's Turn!"
            self.turn = "O"
        else:
            btn.text = "O"
            btn.disabled = True
            self.root.ids.score.text = "X's Turn!"
            self.turn = "X"

        # Check To See if won
        self.win()

    def restart(self):
        # Reset Who's Turn It Is
        self.turn = "X"

        # Enable The Buttons
        self.root.ids.btn1.disabled = False
        self.root.ids.btn2.disabled = False
        self.root.ids.btn3.disabled = False
        self.root.ids.btn4.disabled = False
        self.root.ids.btn5.disabled = False
        self.root.ids.btn6.disabled = False
        self.root.ids.btn7.disabled = False
        self.root.ids.btn8.disabled = False
        self.root.ids.btn9.disabled = False

        # Clear The Buttons
        self.root.ids.btn1.text = ""
        self.root.ids.btn2.text = ""
```



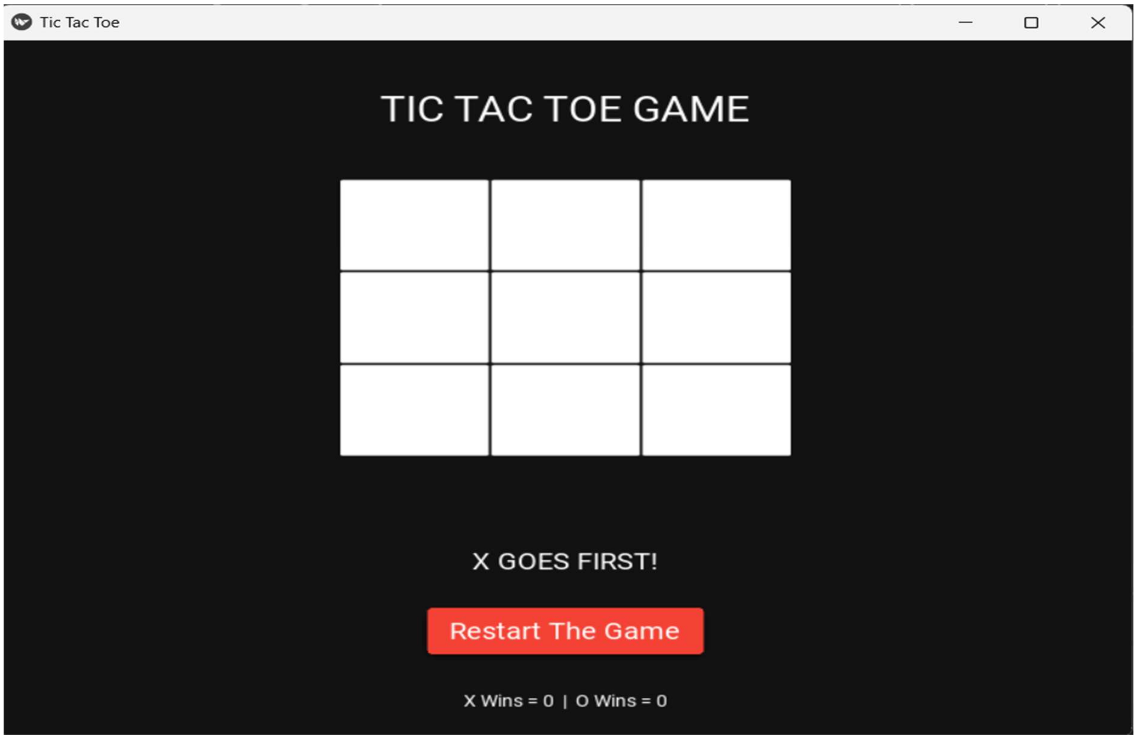
```
        self.root.ids.btn3.text = ""
        self.root.ids.btn4.text = ""
        self.root.ids.btn5.text = ""
        self.root.ids.btn6.text = ""
        self.root.ids.btn7.text = ""
        self.root.ids.btn8.text = ""
        self.root.ids.btn9.text = ""

        # Reset The Button Colors
        self.root.ids.btn1.color = "red"
        self.root.ids.btn2.color = "red"
        self.root.ids.btn3.color = "red"
        self.root.ids.btn4.color = "red"
        self.root.ids.btn5.color = "red"
        self.root.ids.btn6.color = "red"
        self.root.ids.btn7.color = "red"
        self.root.ids.btn8.color = "red"
        self.root.ids.btn9.color = "red"

        # Reset The Score Label
        self.root.ids.score.text = "X GOES FIRST!"

        # Reset The Winner Variable
        self.winner = False


MainApp().run()
```

Kivy code:

```
MDFloatLayout:
        MDGridLayout:
                size_hint: .4, .4
                pos_hint: {'center_x': .5, 'center_y': .6}
                cols: 3
                rows: 3


                Button:
                        id: btn1
                        #RGB - A
                        background_color: (101, 88, 88)
                        text: ""
                        font_size: "45sp"
                        on_release: app.presser(btn1)

                Button:
                        id: btn2
                        #RGB - A
                        background_color: (101, 88, 88)
                        text: ""
                        font_size: "45sp"
                        on_release: app.presser(btn2)

                Button:
                        id: btn3
                        #RGB - A
                        background_color: (101, 88, 88)
                        text: ""
                        font_size: "45sp"
                        on_release: app.presser(btn3)

                Button:
                        id: btn4
                        #RGB - A
                        background_color: (101, 88, 88)
                        text: ""
                        font_size: "45sp"
                        on_release: app.presser(btn4)

                Button:
                        id: btn5
                        #RGB - A
                        background_color: (101, 88, 88)
```

```
            text: ""
            font_size: "45sp"
            on_release: app.presser(btn5)

        Button:
            id: btn6
            #RGB - A
            background_color: (101, 88, 88)
            text: ""
            font_size: "45sp"
            on_release: app.presser(btn6)

        Button:
            id: btn7
            #RGB - A
            background_color: (101, 88, 88)
            text: ""
            font_size: "45sp"
            on_release: app.presser(btn7)

        Button:
            id: btn8
            #RGB - A
            background_color: (101, 88, 88)
            text: ""
            font_size: "45sp"
            on_release: app.presser(btn8)

        Button:
            id: btn9
            #RGB - A
            background_color: (101, 88, 88)
            text: ""
            font_size: "45sp"
            on_release: app.presser(btn9)

    MDLabel:
        id: score
        font_size: "20sp"
        text: "X GOES FIRST!"
        halign: "center"
        pos_hint: {"center_y": .25}

    MDRaisedButton:
        id: restart
  font_size: "20sp"
        text: "Restart The Game"
        pos_hint: {'center_x': .5, 'center_y': .15}
        on_release: app.restart()

MDLabel:
        id: game
        font_size: "14sp"
        text: "X Wins = 0  |  O Wins = 0"
        halign: "center"
        pos_hint: {"center_x": .5, "center_y": .05}

MDLabel:

        font_size: "30sp"
        text: " TIC TAC TOE GAME "
        halign: "center"
        pos_hint: {"center_x": .5, "center_y": .9}
```

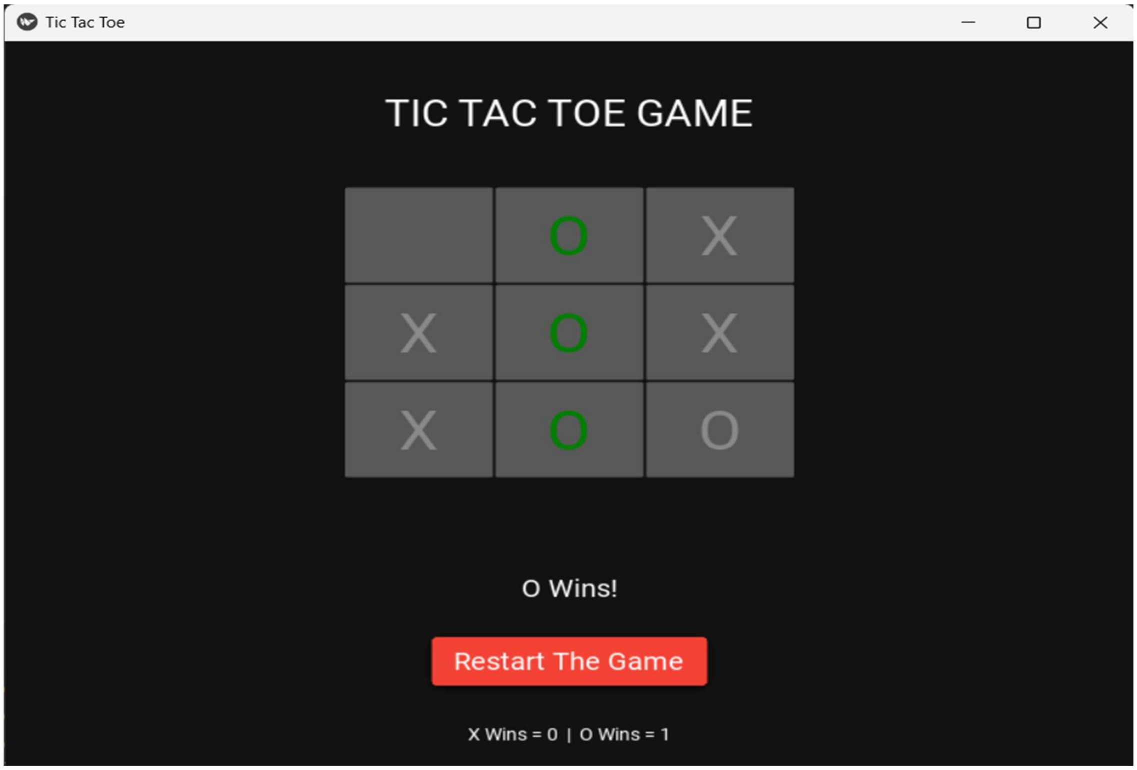
Output:



TIC TAC TOE GAME

X GOES FIRST!

X Wins = 0 | O Wins = 0



TIC TAC TOE GAME

	O	X
X	O	X
X	O	O

O Wins!

X Wins = 0 | O Wins = 1