

Introduction.

1. Network credentials: `Laboratorium-IoT / IoT@laboratorium`

2. Raspberry Pi update:

```
sudo apt install python3-guizero
sudo apt-get update
sudo apt-get upgrade
```

3. Programming IDE - Thonny

Task. 1. GUI introduction.

```
from guizero import App

app = App(title="Hello, world!")
app.display()
```

Task. 2. Image.

```
from guizero import App, Text, Picture

app = App("My image", bg='blue')

picture = Picture(app, image="my_image.gif")
message = Text(app, text='This is an image',
               color='green', size=12, font='verdana')

app.display()
```

Task. 3. Pushbutton.

```
from guizero import App, Box, Text, PushButton
import random

app = App("Dice Rolling Game")

# Roll the dice
def roll_dice():
    dice_value = random.randint(1, 6)
    dice_text.value = str(dice_value)
```

Exercise no 1: Python - Guizero introduction

```
# Create a dice box
dice_box = Box(app, layout="grid")

# Create a dice text
dice_text = Text(dice_box, text="Click to roll", size=40,
grid=[0, 0, 2, 1])

# Create a roll button
roll_button = PushButton(dice_box, text="Roll",
command=roll_dice, grid=[0, 1, 2, 1])

app.display()
```

Task. 4. Random numbers.

```
from guizero import App, Text, PushButton
from faker import Faker

fake = Faker()

# Function to generate a random name
def gen_name():
    name = fake.name()
    names.value = name

app = App("Username Generator", bg='orange')

# Widgets
message = Text(app, text="Click here to generate the username")
button = PushButton(app, gen_name, text="Generate")
button.text_color = '#ffffff'
button.bg = 'red'
names = Text(app, text='', color='#ffffff', size=34)

app.display()
```

Task. 5. Random numbers.

```
from guizero import App, Drawing, Slider, Combo, TextBox

# App
app = App("Task no 5", bg="lightgray")

# Function to draw the meme
def draw_meme():
    meme.clear()
    meme.image(0, 0, "cat-133_128.gif")
    meme.text(
        0, 0, top.value,
        color=color.value,
        size=size.value,
        font=font.value,
    )
    meme.text(
        0, 200, bottom.value,
        color=color.value,
        size=size.value,
        font=font.value,
    )

# TextBox
top = TextBox(app, "top_text", command=draw_meme)
bottom = TextBox(app, "bottom_text", command=draw_meme)

# Drawing
meme = Drawing(app, width="fill", height="fill")

# Combo
color = Combo(app, options=["orange", "red", "blue", "green",
"grey"], command=draw_meme, selected="blue")
font = Combo(app, options=["monospace", "verdana", "helvetica",
"cursive", "sans-serif"], command=draw_meme)

# Slider
size = Slider(app, start=10, end=100, command=draw_meme)

# Call the draw_meme function
draw_meme()
```

```
# Display the app
app.display()
```

Task. 6. Pop-ups.

```
from guizero import App, PushButton, info, warn, error, yesno,
question

# App
app = App("GUI Examples")

# Function for info pop-up
def show_info():
    info("Information", "This is an information pop-up!")

# Function for warning pop-up
def show_warning():
    warn("Warning", "This is a warning pop-up!")

# Function for error pop-up
def show_error():
    error("Error", "This is an error pop-up!")

# Function for yes/no question pop-up
def ask_question():
    response = yesno("Question", "Do you like guizero?")
    if response:
        info("Response", "You clicked 'Yes'!")
    else:
        info("Response", "You clicked 'No'!")

# Button for info pop-up
info_button = PushButton(app, text="Info", command=show_info)

# Button for warning pop-up
warning_button = PushButton(app, text="Warning",
command=show_warning)

# Button for error pop-up
```

Exercise no 1: Python - Guizero introduction

```
error_button = PushButton(app, text="Error",
                           command=show_error)

# Button for yes/no question pop-up
question_button = PushButton(app, text="Question",
                              command=ask_question)

app.display()
```

Task. 7. Calculator.

```
from guizero import App, Box, PushButton, Text

# Function to update the display
def update_display(text):
    display.value += text

# Function to calculate the result
def calculate_result():
    try:
        result = eval(display.value)
        display.value = str(result)
    except:
        display.value = "Error"

# Function to clear the display
def clear_display():
    display.value = ""

# Create the app
app = App("Calculator", width=600, height=800)

# Create the display box
display_box = Box(app, width="fill", height=100, align="top")
display = Text(display_box, text="", size=40)

# Define the button layout
button_layout = [
    ["7", "8", "9", "/"],
    ["4", "5", "6", "*"],
    ["1", "2", "3", "-"],
```

Exercise no 1: Python - Guizero introduction

```
    ["0", ".", "=", "+"],
    ["C"]
]

# Create the button grid
button_box = Box(app, width="fill", height="fill",
layout="grid")

# Create the buttons
buttons = []
for row, row_buttons in enumerate(button_layout):
    for col, label in enumerate(row_buttons):
        button = PushButton(
            button_box, text=label, grid=[col, row],
            width=7, height=5, padx=10, pady=10
        )
        button.bg = "white"
        button.text_color = "black"

        if label == "C":
            button.update_command(clear_display)
        elif label == "=":
            button.update_command(calculate_result)
        else:
            button.update_command(update_display, [label])

    buttons.append(button)

app.display()
```

Task. 8. Final project.

```
from guizero import App, Box, PushButton, Text

# Functions
def clear_board():
    new_board = [[None, None, None],
                  [None, None, None],
                  [None, None, None]]
```

Exercise no 1: Python - Guizero introduction

```
# generate a 3x3 grid
for x in range(3):
    for y in range(3):
        button = PushButton(
            board, text='', grid=[x, y],
            width=3, command=choose_square,
            args=[x,y])
        new_board[x][y] = button
    return new_board

def choose_square(x, y):
    board_square[x][y].text = turn
    board_square[x][y].disable()
    take_turns()
    check_win()

def take_turns():
    global turn
    if turn == "X":
        turn = "O"
    else:
        turn = "X"
    message.value = "It's your turn, " + turn

def check_win():
    winner = None

    # vertical lines
    if(
        board_square[0][0].text == board_square[0][1].text ==
board_square[0][2].text
    ) and board_square[0][2].text in ["X", "O"]:
        winner = board_square[0][0]
```

Exercise no 1: Python - Guizero introduction

```
elif (
    board_square[1][0].text == board_square[1][1].text ==
board_square[1][2].text
) and board_square[1][2].text in ["X", "O"]:
    winner = board_square[1][0]
elif (
    board_square[2][0].text == board_square[2][1].text ==
board_square[2][2].text
) and board_square[2][2].text in ["X", "O"]:
    winner = board_square[2][0]

# horizontal
elif (
    board_square[0][0].text == board_square[1][0].text ==
board_square[2][0].text
) and board_square[2][0].text in ["X", "O"]:
    winner = board_square[0][0]
elif (
    board_square[0][1].text == board_square[1][1].text ==
board_square[2][1].text
) and board_square[2][1].text in ["X", "O"]:
    winner = board_square[0][1]
elif (
    board_square[0][2].text == board_square[1][2].text ==
board_square[2][2].text
) and board_square[2][2].text in ["X", "O"]:
    winner = board_square[0][2]

# diagonals
elif (
    board_square[0][0].text == board_square[1][1].text ==
board_square[2][2].text
) and board_square[2][2].text in ["X", "O"]:
```


Exercise no 1: Python - Guizero introduction

```
winner = board_square[0][0]
elif (
    board_square[2][0].text == board_square[1][1].text ==
board_square[0][2].text
) and board_square[0][2].text in ["X", "O"]:
    winner = board_square[0][2]

if winner is not None:
    message.value = winner.text + " wins!"
elif moves_taken() == 9:
    message.value = "Tie"
    disable_buttons()

def moves_taken():
    moves = 0
    for row in board_square:
        for col in row:
            if col.text == "X" or col.text == "O":
                moves = moves + 1
    return moves

def reset_game():
    global turn
    turn = "X"
    message.value = "It's your turn, " + turn
    for row in board_square:
        for col in row:
            col.text = ""
            col.enable()

def disable_buttons():
    for row in board_square:
        for col in row:
```

Exercise no 1: Python - Guizero introduction

```
col.disable()

turn = "X"

app = App('Tic Tac Toe Game', bg='burlywood')
board = Box(app, layout='grid') # Create the board
board_square = clear_board()
message = Text(app, text="It is your turn, " + turn)
message.text_color = "green"

app.display()
```