**Basics of PyAutoGUI**

PyAutoGUI is a Python library that enables automation of GUI (Graphical User Interface) interactions. It allows you to control the mouse and keyboard to automate tasks on your computer. Here are the basics of using PyAutoGUI:

1. **Installation**: You can install PyAutoGUI using pip, the Python package installer. Open your terminal or command prompt and type:

Copy code

pip install pyautogui

1. **Importing the Library**: Once installed, you need to import the PyAutoGUI module into your Python script:

python

Copy code

import pyautogui

1. **Mouse Control**:
   * **Moving the Mouse**: You can move the mouse cursor to specific coordinates on the screen using the **moveTo()** function:

python

Copy code

pyautogui.moveTo(x, y, duration=seconds)

* + **Clicking**: You can perform mouse clicks using the **click()** function:

python

Copy code

pyautogui.click(x, y)

* + **Dragging**: To perform mouse drag operations, you can use the **dragTo()** function:

python

Copy code

pyautogui.dragTo(x, y, duration=seconds)

1. **Keyboard Control**:
   * **Typing**: You can simulate keyboard key presses using the **typewrite()** function:

python

Copy code

pyautogui.typewrite('Hello, world!')

* + **Pressing and Releasing Keys**: You can press and release specific keys using the **keyDown()** and **keyUp()** functions:

python

Copy code

pyautogui.keyDown('ctrl') pyautogui.press('c') pyautogui.keyUp('ctrl')

1. **Screen Information**:
   * **Screen Size**: You can get the size of the screen using the **size()** function:

python

Copy code

screenWidth, screenHeight = pyautogui.size()

* + **Mouse Position**: You can get the current position of the mouse cursor using the **position()** function:

python

Copy code

mouseX, mouseY = pyautogui.position()

1. **Screenshot**:
   * You can take a screenshot of the entire screen or a specific region using the **screenshot()** function:

python

Copy code

screenshot = pyautogui.screenshot() screenshot.save('screenshot.png')

These are just some of the basic functionalities provided by PyAutoGUI. It's a powerful tool for automating repetitive tasks or testing GUI applications. However, be cautious when using it, especially for actions that could have unintended consequences. Always test your scripts thoroughly before running them on critical systems.

**To get the current location of the mouse cursor on the screen using PyAutoGUI, you can use the** **position()** function. Here's how you can do it:

python

Copy code

import pyautogui

# Get the current position of the mouse cursor

mouse\_x, mouse\_y = pyautogui.position()

# Print the coordinates

print("Mouse Cursor Location - X:", mouse\_x, "Y:", mouse\_y)

This code will print the x and y coordinates of the mouse cursor's current position on the screen. You can then use these coordinates in your script for various purposes, such as moving the cursor or performing actions at specific locations on the screen.

**Project**

**(Automatic Post liker of Instagram)**

**Overview:** An "Automatic Post Liker" for Instagram is a program or script that automatically likes posts on Instagram without requiring manual interaction from the user.



Code:

#intagram liker  
time.sleep**(**3**)  
for** i **in** range**(**20**):** pyautogui.moveTo**(**1875, 534**) #location as per my PC** pyautogui.leftClick**()** time.sleep**(**1**)** pyautogui.moveTo**(**934,819**)** pyautogui.doubleClick**()**print**('50 like done on your page!')**

Note:

Cursor location is different for different device so you have to find your proper cursor location using below code. We Have to found the location of Like button and next post button as per image red circles.

Copy code

import pyautogui

# Get the current position of the mouse cursor

time.sleep(5)

mouse\_x, mouse\_y = pyautogui.position()

# Print the coordinates

print("Mouse Cursor Location - X:", mouse\_x, "Y:", mouse\_y)



