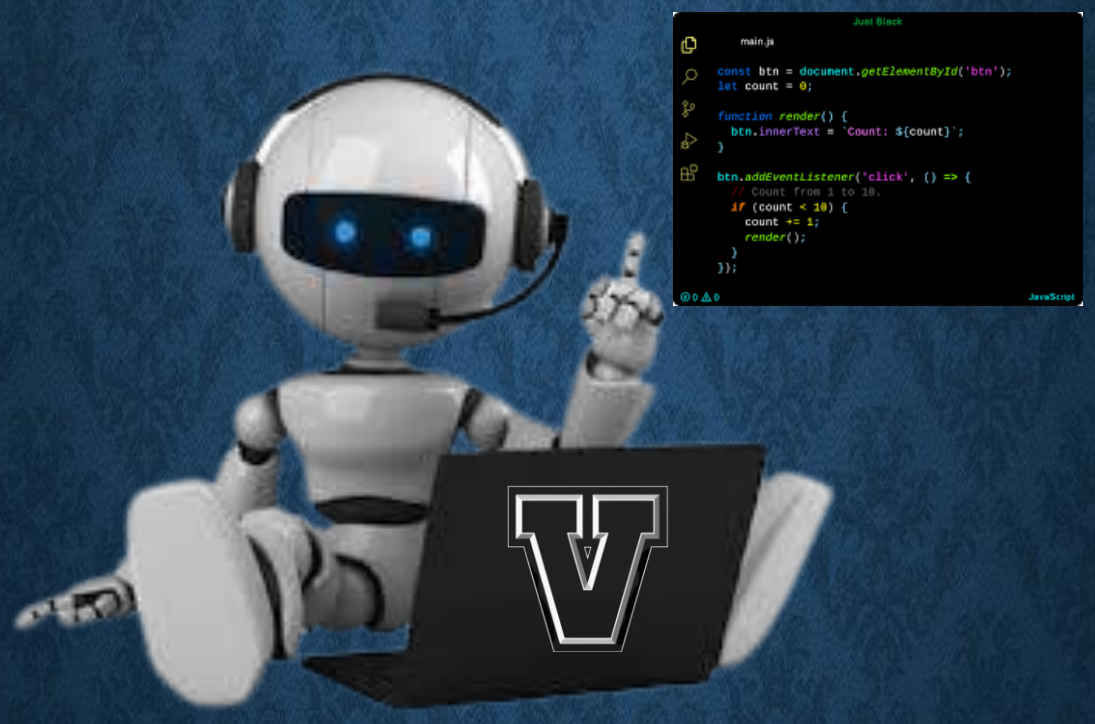
HOW TO MAKE YOUR OWN ROBOT ?



VICTOR LEMOS YOSHIDA

1. PYAUTOGUI

code_sample.py

```python
import pyautogui

# Move the mouse
pyautogui.moveTo(100, 100, duration=2)

# Left click
pyautogui.click()

# Type text
pyautogui.write('Hello, world!', interval=0.2)

# Press a key
pyautogui.press('enter')
```

PyAutoGUI is a Python library that enables automation of user interface tasks by controlling the mouse and keyboard. It allows developers to simulate mouse movements, clicks, and drags, as well as keyboard key presses and text entry. PyAutoGUI is particularly useful for automating repetitive tasks that involve interacting with graphical user interfaces (GUIs), such as filling out forms, clicking buttons, and navigating through menus. The library can also take screenshots and locate images on the screen, facilitating automated workflows that depend on visual elements. Its ease of use and ability to work across multiple operating systems make PyAutoGUI a versatile tool for automating routine tasks and improving productivity.

2. SELENIUM

```python
from selenium.webdriver.common.by import By
from selenium import webdriver
from selenium.webdriver.common.action_chains import ActionChains

#Lambdatest setup and opening the desired website
username = "Your LambdaTest Username"
accessToken = "Your LambdaTest Access Key"
gridUrl = "hub.lambdatest.com/wd/hub"

capabilities = {
    'LT:Options' : {
        "user" : "Your LambdaTest Username",
        "accessKey" : "Your LambdaTest Access Key",
        "build" : "your build name",
        "name" : "your test name",
        "platformName" : "Windows 11",
    },
    "browserName" : "Chrome",
    "browserVersion" : "103.0",
}

url = "https://"+username+":"+accessToken+"@"+gridUrl

browser = webdriver.Remote(
    command_executor=url,
    desired_capabilities=capabilities
)

browser.maximize_window()
browser.get("https://www.lambdatest.com/selenium-playground/table-sort-search-demo")
```

Selenium is an open-source library designed to automate web browsers. It allows for programmatic control of a web browser, enabling the automatic execution of repetitive and complex tasks. Selenium is widely used for testing web applications, ensuring they work correctly across different browsers and scenarios. In addition to testing, Selenium is useful for web scraping, which involves extracting data from web pages, as well as automating interactions on websites, such as form filling, page navigation, and executing clicks and other actions. Its flexibility and support for multiple browsers, including Chrome, Firefox, and Safari, make Selenium a powerful and indispensable tool for developers and testers seeking efficiency and precision in their web-related activities.