**Code:**

import streamlit as st

from selenium import webdriver

from selenium.webdriver.common.by import By

from selenium.webdriver.chrome.options import Options

import time

# Function to fetch and count hyperlinks using Selenium

def fetch\_hyperlinks\_selenium(url):

# Configure Selenium WebDriver

options = Options()

options.add\_argument('--headless') # Run browser in headless mode (no UI)

options.add\_argument('--disable-gpu') # Disable GPU (for compatibility)

options.add\_argument('--no-sandbox') # Disable sandbox mode (for Linux)

driver = webdriver.Chrome(options=options)

try:

driver.get(url) # Open the webpage

time.sleep(3) # Wait for page to load

# Find all <a> tags on the page

links = driver.find\_elements(By.TAG\_NAME, 'a')

# Extract valid href attributes

hyperlinks = [link.get\_attribute('href') for link in links if link.get\_attribute('href')]

return len(hyperlinks), hyperlinks # Return count and list of hyperlinks

except Exception as e:

return None, f"Error: {e}"

finally:

driver.quit() # Close the browser

# Streamlit App

def main():

st.title("Hyperlink Counter (Selenium)")

# Input for URL

url = st.text\_input("Enter the URL of the webpage", placeholder="https://example.com")

# Fetch and display hyperlinks

if st.button("Count Hyperlinks"):

if url:

with st.spinner("Fetching hyperlinks..."):

count, result = fetch\_hyperlinks\_selenium(url)

if count is not None:

st.success(f"Total hyperlinks found: {count}")

# Display the hyperlinks if available

if st.checkbox("Show hyperlinks"):

st.write("### Hyperlinks:")

for idx, link in enumerate(result, start=1):

# Display hyperlinks with numbers

st.markdown(f"{idx}. [\*\*{link}\*\*]({link})")

else:

st.error(result)

else:

st.error("Please enter a valid URL.")

if \_\_name\_\_ == "\_\_main\_\_":

main()

### **What Does the Code Do?**

This app lets you enter a website link (URL), then counts all the clickable links (hyperlinks) on that page. It even shows you a list of the links if you want to see them.

### **How the App Works:**

1. You type a webpage link like https://example.com into the box.
2. The app opens the website in the background (using a tool called **Selenium** that acts like a robot browser).
3. It looks through the page to find all the links (those blue, underlined clickable things you see on websites).
4. The app counts these links and shows the total number.
5. If you want, it also shows a full list of all the links.

### **Parts of the Code**

Let’s go step by step.

#### **1. Importing the Tools**

import streamlit as st

from selenium import webdriver

from selenium.webdriver.common.by import By

from selenium.webdriver.chrome.options import Options

import time

* **streamlit**: Helps create the app’s user interface (the webpage you see when you use the app).
* **selenium**: A tool that can open websites, find things on the page (like links), and interact with them.
* **time**: Used to pause for a few seconds, giving the website time to load before searching for links.

#### **2. Function to Find Links**

def fetch\_hyperlinks\_selenium(url):

options = Options()

options.add\_argument('--headless') # Browser runs in the background

options.add\_argument('--disable-gpu')

options.add\_argument('--no-sandbox')

driver = webdriver.Chrome(options=options) # Start a browser (Chrome)

* **Options for Selenium**: These lines make the browser run invisibly in the background so it doesn’t open a window every time.
* **webdriver.Chrome**: Starts a Chrome browser (invisible)

try:

driver.get(url) # Open the webpage

time.sleep(3) # Wait for the page to fully load

links = driver.find\_elements(By.TAG\_NAME, 'a') # Find all <a> tags (links)

hyperlinks = [link.get\_attribute('href') for link in links if link.get\_attribute('href')]

return len(hyperlinks), hyperlinks # Return number of links and the list

**driver.get(url)**: Opens the URL you type into the box.

**time.sleep(3)**: Pauses for 3 seconds to let the page load completely.

**find\_elements(By.TAG\_NAME, 'a')**: Searches for all <a> tags (these tags are used for links in webpages).

**get\_attribute('href')**: Grabs the actual link (e.g., https://google.com) from each <a> tag.

**Returns**:

* **How many links are there**.
* **The list of all the links**.

#### **Objective**

To develop a Python-based Streamlit app that uses Selenium to fetch a webpage, count the total hyperlinks, and display them interactively.

#### 

#### **Requirements**

1. **Software/Tools**:
   * Python 3.x
   * Google Chrome Browser
   * ChromeDriver
   * Streamlit
   * Internet connection
2. **Libraries**:
   * selenium
   * streamlit

#### **Theory**

1. **Selenium**:  
   Selenium is a browser automation tool that allows dynamic webpage content (like JavaScript-rendered pages) to be interacted with and scraped.
2. **Streamlit**:  
   A Python library for creating interactive web apps with minimal effort.
3. **Hyperlink**:  
   Hyperlinks are references to web resources, typically identified by the <a> tag in HTML.

#### **Step-by-Step Procedure**

1. **Install Required Packages**: Run the following commands to install necessary libraries:

***pip install selenium streamlit***

2.**Save the Code**: Save the code in a file, e.g., selenium\_hyperlink\_counter.py.

3.**Run the App**:

* Open a terminal and navigate to the file's location.

Run the following command

streamlit run selenium\_hyperlink\_counter.py

4.**Provide Input**:

* Enter a webpage URL (e.g., https://www.wikipedia.org) into the app input box.

**Output**:

* The app will display the total number of hyperlinks found.
* You can check the box to view the list of hyperlinks.



