**Step-by-Step Guide: Using the ILLiad Automation Script For requesting Journal Articles on Mac**

### **1. Prerequisites**

Before using the script, ensure the following are in place:

1. **Python 3 Installed**:

Verify Python 3 is installed by running:  
bash  
Copy code  
python3 --version

* + If not installed, download and install it from [python.org](https://www.python.org/downloads/).

**Install Required Python Libraries**: Open a terminal and install the necessary libraries:  
bash  
Copy code  
python3 -m pip install selenium webdriver-manager pandas openpyxl

1. **Chrome Browser Installed**:
   * Ensure Google Chrome is installed on your Mac.
   * Verify the version by navigating to Chrome > About Google Chrome.
2. **ChromeDriver Installed Automatically**: The script uses webdriver-manager to install the correct ChromeDriver version automatically.

### **2. Prepare Your Environment**

1. **Set Up the Excel File**:
   * Use the provided Excel file format with columns like Journal, Article Title, Author, etc.
   * Save the file to a convenient location, e.g., ~/Documents/illiad\_requests.xlsx.
2. **Download the Script**:
   * Save the illiadroboloaner1\_0.py script to a folder, e.g., ~/Documents.

**Open Chrome in Debug Mode**: Run the following command in your terminal to start Chrome with remote debugging:  
bash  
Copy code  
/Applications/Google\ Chrome.app/Contents/MacOS/Google\ Chrome --remote-debugging-port=9222 --user-data-dir=/tmp/selenium

* + A new Chrome window will open.
  + Log in to your ILLiad account in this browser.
  + Navigate to New Request > Journal. Deal with the cookie pop up by clicking “got it” or whatever it was it says. Now the pop up won't interfere with the script later

### **3. Run the Script**

**Navigate to the Script Directory**: In the terminal, go to the folder where you saved the script:  
bash  
Copy code  
cd ~/Documents

**Execute the Script**: Run the script using Python 3:  
bash  
Copy code  
python3 illiadroboloaner1\_0.py

1. **Monitor Execution**:
   * Watch the browser window to ensure the script interacts with the ILLiad form.
   * Check the terminal for any errors or confirmation messages.

### **4. Handling Issues**

1. **Errors in the Terminal**:
   * Review the error messages to identify what went wrong (e.g., missing data, incorrect element names).
   * Fix the issue in the script or Excel file and rerun.
2. **Row Failures**:
   * If some rows fail, edit the Excel file to include only the failed rows and rerun the script.
3. **Cookie Pop-Ups**:
   * The script dismisses cookie pop-ups automatically, but if it fails, ensure the pop-up button ("Got It") is correctly identified.

### **5. Post-Run Validation**

1. Log into your ILLiad account and verify that all requests were submitted.
2. Check the terminal for any skipped or failed rows and handle them as needed.

### **6. Automate the Workflow (Optional)**

For frequent use, you can automate the process:

1. **Create a Dedicated Chrome Profile**:
   * This ensures your ILLiad credentials persist across sessions.

Use a command like this to start Chrome:  
bash  
Copy code  
/Applications/Google\ Chrome.app/Contents/MacOS/Google\ Chrome --remote-debugging-port=9222 --user-data-dir=/path/to/custom/profile

**Create a Shell Script to Simplify Execution**: Create a shell script (run\_illiad.sh) to automate starting Chrome and running the Python script:  
bash  
Copy code  
#!/bin/bash

/Applications/Google\ Chrome.app/Contents/MacOS/Google\ Chrome --remote-debugging-port=9222 --user-data-dir=/tmp/selenium &

sleep 5

python3 /path/to/illiadroboloaner1\_0.py

Save this file, make it executable:  
bash  
Copy code  
chmod +x run\_illiad.sh

Run it using:  
bash  
Copy code  
./run\_illiad.sh

### **Sharing the Guide**

* **Share the Script**: Provide the illiadroboloaner1\_0.py file and this guide to your friends.
* **Dependencies**: Remind them to install Python 3 and the required libraries as listed above.

### **Summary Workflow for Users**

1. Set up the Excel file.
2. Open Chrome with debugging enabled.
3. Run the script using Python 3.
4. Verify requests in ILLiad after the script finishes.

Let me know if you'd like me to tweak or expand this guide further! 🚀

[Click here to try this new GPT!](https://f614.short.gy/Code)