Project Documentation

web scarpping: Web scraping is used to collect large information from websites. But why does someone have to collect such large data from websites? To know about this, let's look at the applications of web scraping

Use Library in this project:

Selenium:Selenium Python bindings provides a simple API to write functional/acceptance tests using Selenium WebDriver. Through Selenium Python API you can access all functionalities of Selenium WebDriver in an intuitive way.Selenium Python bindings provide a convenient API to access Selenium WebDrivers like Firefox, Ie, Chrome, Remote etc. The current supported Python versions are 2.7, 3.5 and above.

pytesseract: Python-tesseract is an optical character recognition (OCR) tool for python. That is, it will recognize and "read" the text embedded in images.Python-tesseract is a wrapper for Google's Tesseract-OCR Engine. It is also useful as a stand-alone invocation script to tesseract, as it can read all image types supported by the Pillow and Leptonica imaging libraries, including jpeg, png, gif, bmp, tiff, and others. Additionally, if used as a script, Python-tesseract will print the recognized text instead of writing it to a file.
Time: for some interval in webdriver

json: for store the dict data in json formate

problem Statement

Develope a Python CLI Application with the following utilities

Project Docomantation

1. Make a scraper which fills the form at https://parivahan.gov.in/rcdlstatus/?pur_cd=101 and scrapes the resultant data.
2. Use the Python library requests, lxml along with xpath to do so.
3. The page contains a Captcha so write a dummy function get_captcha() which outputs the text of captcha based on the input image of the captcha, assume that the captcha can be wrong sometimes, so handle retries accordingly. For testing purposes, you can use the Python input function to enter captchas manually while scrapping, but you will be judged after we replace it with our get_captcha() and run tests on thousands of samples, so make sure to make this scrapper fault tolerant and output useful error messages.

Here is a sample page of the result after filling the form - https://imagebin.ca/v/4eE1iM6REVNM

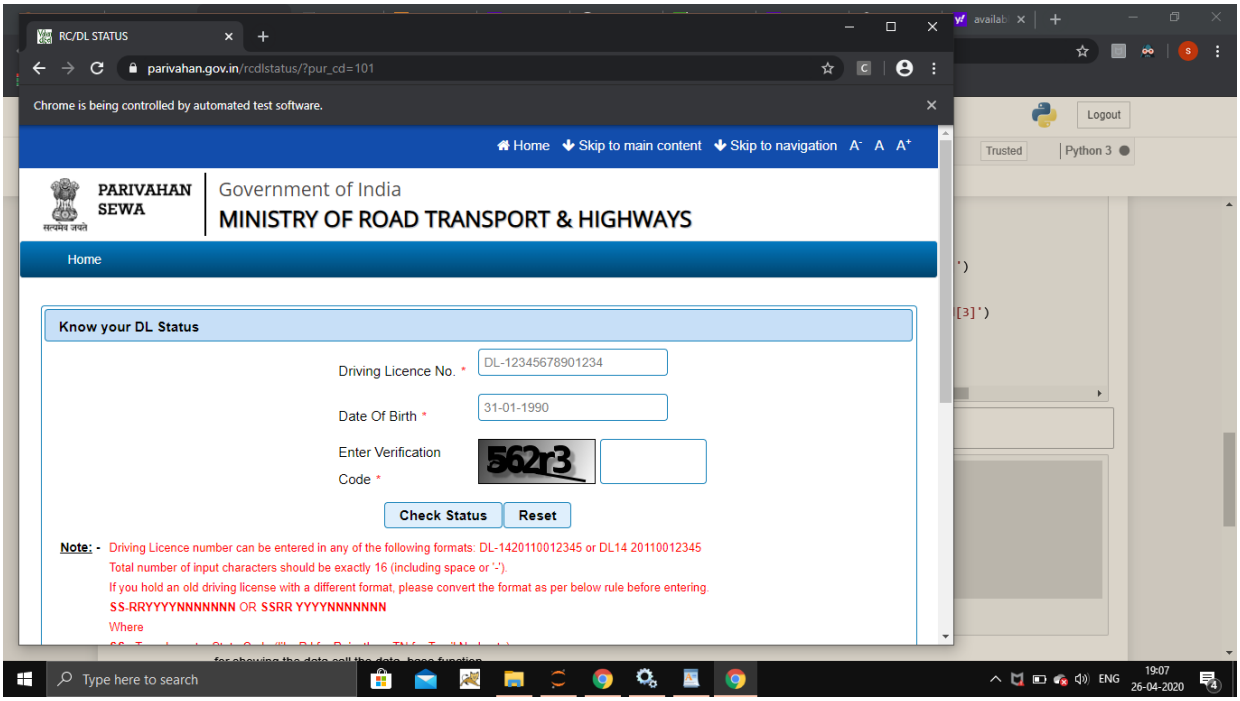
Here is a sample Driving License to try out - https://5.imimg.com/data5/UD/GT/MY-35587652/driving-license-service-500x500.jpg

The result should have all the fields like Name, Date of Issue, Date of expiry, Class of Vehicles etc.

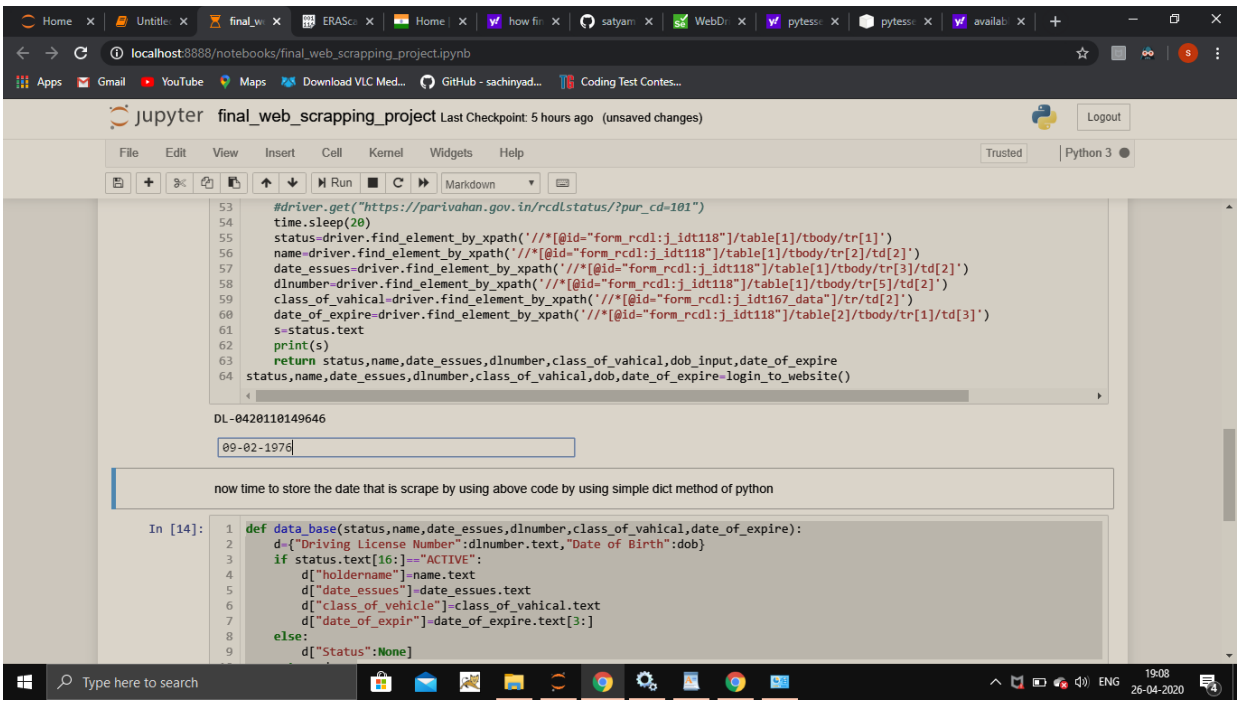
The final application should demand a Driving License Number and Date of Birth, and should shell out the results in JSON format.

solution

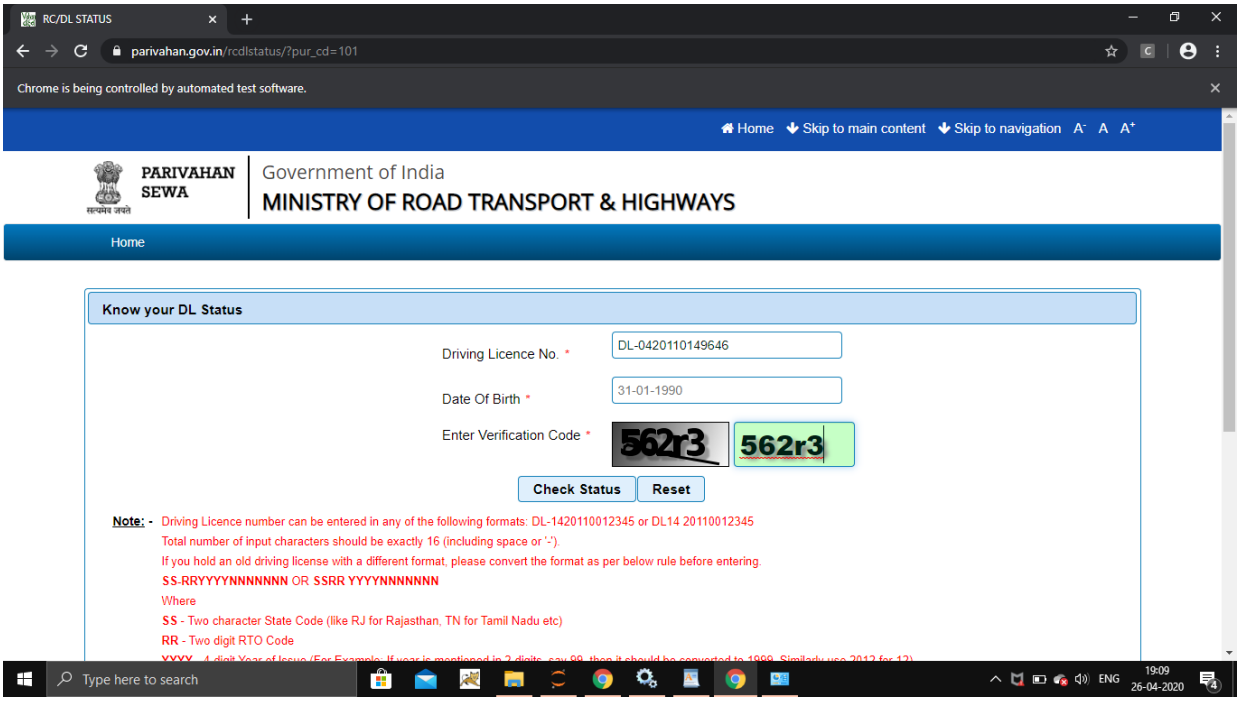
Automation: for solution of this project we use some method that helpful to this .
Step 1: by using automation we fill the form of given url.



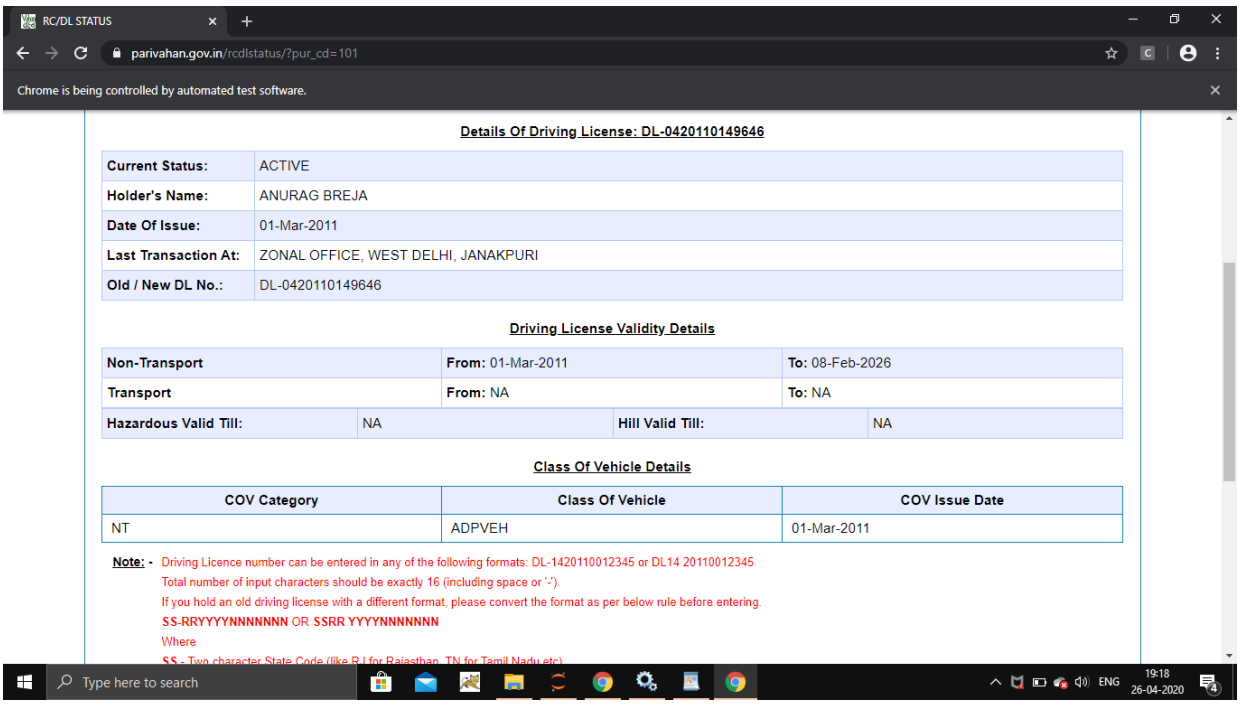
Step 2: for filling the form we have three section dl no ,dob and chaptcha code for solution of this we use selenium library .



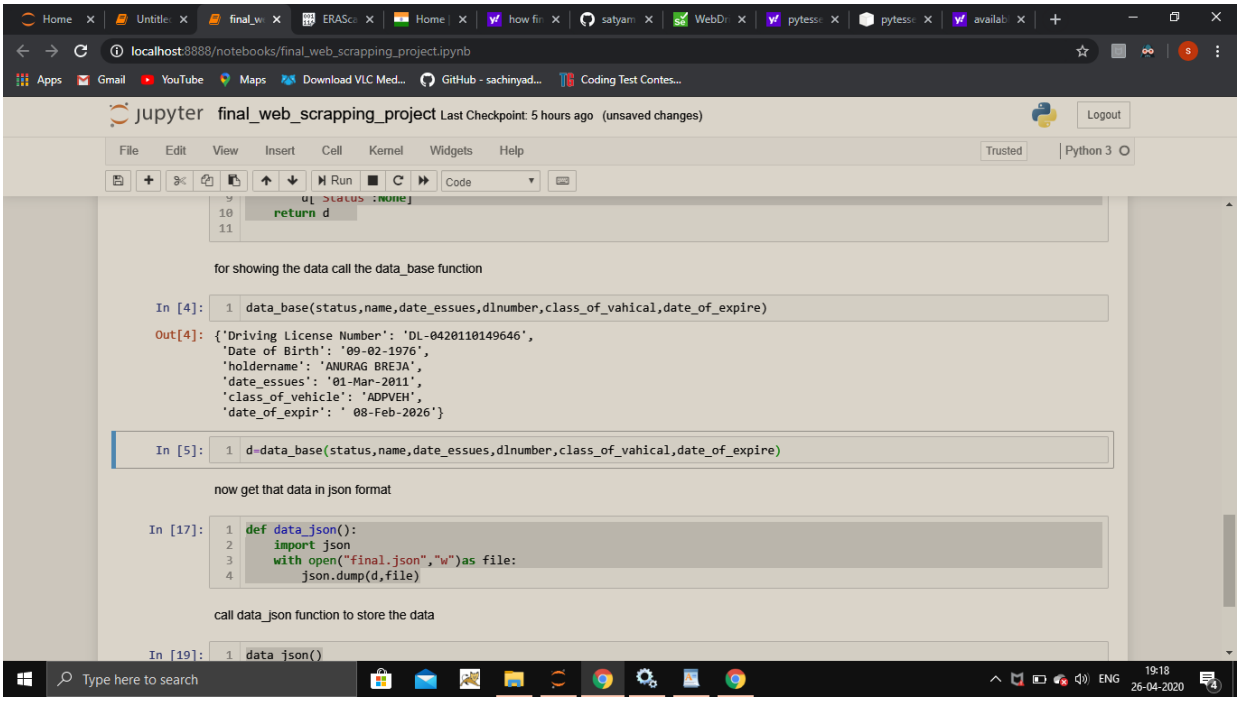
Step 3: for captcha we download image of captcha and then crop and remove noise from that and then fill but here i fill the captcha by user.



Data scrapping: for scrapping the data also use selenium that is scrap data by using some method that is available in that then store that data in dict.



Step 4: josn format : after the srapping store the all data into json format by using json.


localhost:8888/notebooks/final_web_scrapping_project.ipynb
jupyter final_web_scrapping_project Last Checkpoint: 5 hours ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted Python 3
Run Code
d['status']=None
return d
for showing the data call the data_base function
In [4]: data_base(status,name,date_essues,dlnumber,class_of_vahical,date_of_expire)
Out[4]: {'Driving License Number': 'DL-0420110149646',
'Date of Birth': '09-02-1976',
'holdername': 'ANURAG BREJA',
'date_essues': '01-Mar-2011',
'class_of_vehicle': 'ADPVEH',
'date_of_expir': ' 08-Feb-2026'}
In [5]: d=data_base(status,name,date_essues,dlnumber,class_of_vahical,date_of_expire)
now get that data in json format
In [17]: def data_json():
import json
with open("final.json","w")as file:
json.dump(d,file)
call data_json function to store the data
In [19]: data_json()
Type here to search
ENG
19:18
26-04-2020