**DAILY ASSESSMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **03/06/2020** | **Name:** | **Dhavala** |
| **Course:** | **DIGITAL DESIGN USING HDL** | **USN:** | **4AL17EC027** |
| **Topic:** | * **EDA Playground Tutorial Demo Video** * **How to Download And Install Xilinx Vivado Design Suite** * **Vivado Design Suite for**   **implementation of HDL code** | **Semester & Section:** | **6TH SEM & A Section** |
| **Github Repository:** | **Dhavala27** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
|  |

|  |
| --- |
| **Report** |
|  |

**DAILY ASSESSMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **03/06/2020** | **Name:** | **Dhavala** |
| **Course:** | **PYTHON** | **USN:** | **4AL17EC027** |
| **Topic:** | * **Application 7: Scrape Real Estate Property Data from the Web** | **Semester & Section:** | **6TH SEM & A Section** |
| **Github Repository:** | **Dhavala27** |  |  |

|  |
| --- |
| **AFTERNOON SESSION DETAILS** |
|  |

|  |
| --- |
| **Report**  Scrape Real Estate Property Data from the Web  Web Scraping in real estate to the rescue  Web scraping is the process of sorting through overwhelming amounts of data, refine the user’s searches and provide a list of relevant information. In a realtor’s case, it is the go-to tool for organized property listings. Scraping the web provides parameters  which the realtor can further study to determine sales and prospective buyers. Parameters extracted by web scraping are:  •Size  •Property type  •Location  •Sale price  •Size  •Amenities  •Monthly rental price  •Parking spaces  •Agent contact  This information is displayed in form of a spreadsheet, allowing the realtor to make comparisons of relevant parameters.  1.Property value tracking  Let’s assume you decide to sell your property. Scraping the web for the value of similar properties can aid you in setting a good value on your own. This allows users searching for such properties to get fair deals, and on the other, you getting a profitable one.  2. Making the right investment  Obtaining real estate data is hard, as result of which most investors make financial investments blindly. ith web scraping, an investor can make decisions based on qualitative and relevant empirical data, rather than outdated or incomplete information. Aggregating property data from real estate listing websites is essential for investment analysis.  3. Rental Yield  Rental yield is the most important factor to be considered before investing in property. By scraping data from real estate websites, you can determine which properties have the best rental yield for any suburb. Moreover, scraping answers which property types  (house, apartment, 1 bedroom, 2 bedrooms) are more preferred in a particular area and yield the best return on investment.  4. Track vacancy rates  A vacant investment property is risky. To minimize this risk, it is imperative to analyze  property data and suburbs which have higher rental listings.The above parameters are the most relevant decoded by web scraping through numerous websites online. Having the above details at your fingertips improves a realtor’s efficacy at decision making, better communication and faster and profitable sales. The role of web scraping in retail is just getting started, its potential is however limitless. Build a Web-based Financial Graph Firstly, web scraping is performed to scrape the required data. We are using real-time stock market data for scraping and then store it into a CSV file format using Python libraries. Data is extracted from the web using Python's Beautiful Soup module. Beautiful Soup is an inbuilt package in Python that is used for parsing HTML and XML documents (including having distorted or abnormal mark- up, i.e. non-closed tags, so named after tag soup). It develops a parse tree for already parsed  documents that can further be used to extract data from HTML file or document, which is useful in case of web scraping. Secondly, the graph is a plot on our designed website from the CSV files containing the scraped data from the web. This task is done by using c3.js and Papa Parse libraries and their respective dependencies. C3 provides an easy way to construct D3-based charts by encapsulating the code that is required to generate the entire chart/graph. C3 library of JavaScript provides a wide range of APIs and Callbacks (Callback is a function in JavaScript that is executed after  another program has finished its execution) to access the status of the chart at a particular time. By using this C3 library, we can update the chart/graph even after it is accomplished.  Another library that we have used in graph making is Papa Parse which is the fastest in-browser CSV (or delimited text) parser for JavaScript. Papa Parse is the world's first multi- threaded CSV parser used for the browser. It is reliable and easy to use. Papa-parse is an effective and convenient CSV parser that can handle files having size in gigabytes without crashing. It is capable of manipulating your CSV files in many ways. First off, the input. This component can read your data from anywhere, via a URL, from a raw string or even from your local storage. The output will be an array of rows, where each row is an array of table data, and it will be returned if the header flag is not set. Otherwise, an array of objects will be the product, where each object is a map comprising of the column name and its corresponding value or the row  (e.g., {col1: value1, col2: value2}). The recent format is in the vicinity of a JSON file  Request Headers  Note  Whenever I use this code in the next videos:  r = requests.get("http://www.pythonhow.com/real-estate/rock-springs-wy/LCWYROCKSPRINGS/")  please use this instead:  r = requests.get("http://www.pyclass.com/real-estate/rock-springs-wy/LCWYROCKSPRINGS/", headers={'User-agent': 'Mozilla/5.0 (X11; Ubuntu; Linux x86\_64; rv:61.0) Gecko/20100101 Firefox/61.0'})  The rest of the code stays the same.  So, we're just changing the domain name from *pythonhow* to *pyclass* and we're adding a *header* argument. Some webpages don't like scripts sometimes, so adding a header allows the script to impersonate a web browser. |