## **What is does:**

Scrapes the Derby City Council Planning Application portal for usable data that Netgraph can ingest. There are four output files created throughout the process; there are samples of each at the end of this document.

## **How is works:**

As a JavaScript application

The overarching function is

doScrapeAlphabet();

the function accepts two params.

* **"https://eplanning.derby.gov.uk/online-applications/search.do?action=property&type=atoz&letter=U"**
* The starting URL is ***required*** and is included to ensure the scraper works even if they change where the data is served from.
* **"U"**
* The endLetter is ***optional*** and signifies the end of the process, if omitted it will scrape all the was to Z (upper and lower lette accepted!)

doScrapeAlphabet("https://eplanning.derby.gov.uk/online-applications/search.do?action=property&type=atoz&letter=D", "M");

As a Node application

node dcc-planning.js "https://eplanning.derby.gov.uk/online-applications/search.do?action=property&type=atoz&letter=C" "G"

As an application Executable

WiP 👀

## **What happens:**

1. First the **delta** between **start** and **end letters** is worked out.
2. Each letter has a URL check carried out on it to ensure the page-set is valid.
3. These are chunked into an array of (max) 10 to allow for faster scraping (multiple browser instances).
4. Any pages found to be void are omitted from the alphabet array that is returned and saved to file under *alphebet.json*.
5. The data saved to file is an array of alphabet URLs.
6. There is user feedback (console log) to give information on start/next/finish and total count.
7. The **streets** are **scraped**.
8. This step collects usable street URLs. ie, all the streets under A, B, C...
9. Some street names have 10's of pages, so they are paginated.
10. All paginated results are stored to file (*streets.json*) as the process goes on to the next letter in the array.
11. The data saved to file is an array of street objects including the street name and a URL to that street names properties.
12. The **properties** are **scraped**.
13. The saved streets array is then chunked into an array of (max) 10 and processed async.
14. Each property on a street is saved to (*properties.json*) file.
15. This step is also heavily paginated due to the amount of properties available per street. 822 is a larger number seen in the U streets.
16. The data saved to file is an array of property objects including the property *name/number*, *street*, *city* and *postcode* and a URL to that property.
17. **Property processing** is the final step.
18. The json file of properties is chunked into an array of 10 and processed up to 10 instances at a time.
19. From each property URL an object is returned with an address string and an item that includes an application URL, status of the application and a unique reference number.
20. When the unique reference number is added to **this URL** **https://docs.derby.gov.uk/padocumentserver/index.html?caseref=**[ADD-REF-HERE] all related documentation can be read/scraped as required

## **Returned Information Structures:**

Alphabet:

|  |
| --- |
| [  "https://eplanning.derby.gov.uk/online-applications/search.do?action=property&type=atoz&letter=S",  "https://eplanning.derby.gov.uk/online-applications/search.do?action=property&type=atoz&letter=T",  "https://eplanning.derby.gov.uk/online-applications/search.do?action=property&type=atoz&letter=U" ] |

Streets:

|  |
| --- |
| [  [  {  "street": "Uffa Magna , Derby",  "link": "https://eplanning.derby.gov.uk/online-applications/alphabeticalSearchResults.do?streetKeyValue=OX1JS900DT03S&action=firstPage&streetLetter=U"  }   ...  ] ] |

Properties:

|  |
| --- |
| [  [  {  "property": "1 Uffa Magna Derby DE3 0SN",  "link": "https://eplanning.derby.gov.uk/online-applications/propertyDetails.do?keyVal=OX20MJ00DT003&activeTab=summary"  },    ...  ]  ... ] |

PropertyData:

|  |
| --- |
| [  [  {  "address": "1 Uffa Magna Derby DE3 0SN",  "item": {  "link": "https://eplanning.derby.gov.uk/online-applications/applicationDetails.do?previousCaseType=Property&keyVal=ZZZZSBFSXE142&previousCaseNumber=OX20MJ00DT000&previousCaseUprn=010010625482&activeTab=summary&previousKeyVal=OX20MJ00DT003",  "status": "Granted",  "ref": "06/99/00715"  }  }  ]   ... ] |