**Code for third party API, in this case UniProt**

Generic website - <https://www.uniprot.org/help/api>

The code in section (**A**) below relates to retrieving entries by query (<https://www.uniprot.org/help/api_queries>)

(**A**)

To initiate a search for a specific accession number in UniProt:-

(i)

<https://www.uniprot.org/uniprot/?query=Q64303>

To initiate a search for a specific accession number in UniProt and give parameters for cellular location and family:-

(ii)

https://www.uniprot.org/uniprot/?query=accession:Q9BQI6&sort=score&columns=domain,domains,comment(DOMAIN),comment(SUBCELLULAR LOCATION),feature(INTRAMEMBRANE),feature(TOPOLOGICAL DOMAIN),feature(TRANSMEMBRANE),length&format=html

*Importantly, the final format of html can be modified to txt, tab and other delimiters.*

(**B**)

To obtain the UniProt information, we can use Pandas:-

*### Import relevant modules***import** pandas **as** pd  
  
*### Select the specific accession number to search from*url = **'https://www.uniprot.org/uniprot/?query=Q64303'***### Define a variable, here denoted as dfs (dataframes)*dfs = pd.read\_html(url)  
  
*### Print the results. Here we can select specific data frames which relate to tables from the website.*print(dfs[0:3])

This format however does not work for query with specific fields. e.g. case (ii) above. The UniProt website has some on how do this and it (<https://www.uniprot.org/help/api_idmapping>)

import urllib,urllib2

url = 'https://www.uniprot.org/uploadlists/'

params = {

'from':'ACC',

'to':'P\_REFSEQ\_AC',

'format':'tab',

'query':'P13368 P20806 Q9UM73 P97793 Q17192'

}

data = urllib.urlencode(params)

request = urllib2.Request(url, data)

contact = "" # Please set a contact email address here to help us debug in case of problems (see https://www.uniprot.org/help/privacy).

request.add\_header('User-Agent', 'Python %s' % contact)

response = urllib2.urlopen(request)

page = response.read(200000)

I am currently working on this as urllib and urllib2 have changed and it is a little bit more tricky!