My solution:

1. create a class to represent one website.

2. create getSoup method, which take an url as input and output a beautifulsoup object.

3. create a getTargeturl method. In this method we first scraping root url to find tag that contain internal links and external links

second, I filter out tags that do not contain keywords (annual reports,press releases, about, investors)

third, recursively access all the internal links.

In the second step, we store all the internal and external target tags(contain keywords) in separate instance variable(interanltargettag, externaltargettag).

4 create getTargetdata method. access url that stored in instance variables. If this is a normal url, find the table and specific data. if this is a pdf, extract pdf with pdfminer3k to string and then find specific data. regular expression: premium[\s\S]\*?(\d+(,\d+)+)

5, Then we may have several premiums and revenues, then we just pick the maxium

Explanation:

1.403 error, this means the server thinks this is a bot, so it denied its access. So I need to use phantomjs browser to access the url to fool the server.

2. why pick the largest premium? Because when I scraping the data, in most circumstance, the data was in a table(whether it is in a pdf or htlml) when I convert table into string, the keywords premium will occur and in this row, will be its values year by year. only the biggest number that contain this keyword will be the data that we want. If there will be any smaller data, it cannot be an annual revenue or it doesn't contain the keywords but bigger than the value that contain keywords.

view the code at :<https://github.com/kyao4/web_crawler>

So far this version of web crawler can get 3 premiums for first 20 websites in the list. I didn’t test the rest of them but I think this is enough for now