Given the following python function parse shown in textbox1

Textbox 1

|  |
| --- |
| **def** parse(self, response):  *'''  Required function for spider to crawl  Run two different type of parsing depending on the json keyword type of parse  if type of parse == google\_search --> get list of links from google results  if type of parse == general --> get the meta information for each site  '''* **if** self.setting\_data[**'type\_of\_parse'**] == **'google\_search'**:  **print  print 'For google search parsing'** *## Get the selector for xpath parsing* sel = Selector(response)  google\_search\_links\_list = sel.xpath(**'//h3/a/@href'**).extract()  google\_search\_links\_list = [re.search(**'q=(.\*)&sa'**,n).group(1) **for** n **in** google\_search\_links\_list **if** re.search(**'q=(.\*)&sa'**,n)]   *## Display a list of the result link* **for** n **in** google\_search\_links\_list:  **print** n   *## Dump all results to file* self.combine\_all\_url\_link\_for\_multiple\_search(google\_search\_links\_list)   **if** self.setting\_data[**'type\_of\_parse'**] == **'general'**:   **print   print 'general website processing'** sel = Selector(response)   *## Get meta info from website* title = sel.xpath(**'//title/text()'**).extract()  **if** len(title)>0:  title = title[0].encode(errors=**'replace'**) *#replace any unknown character with ?* contents = sel.xpath(**'/html/head/meta[@name="description"]/@content'**).extract()  **if** len(contents)>0:  contents = contents[0].encode(errors=**'replace'**) *#replace any unknown character with ?* **if** ENABLE\_PARAGRAPH\_STORED:  paragraph\_list = sel.xpath(**'//p/text()'**).extract()  para\_str = self.join\_list\_of\_str(paragraph\_list, joined\_chars= **'..'**)  para\_str = para\_str.encode(errors=**'replace'**)  para\_str = self.remove\_whitespace\_fr\_raw(para\_str)  *## Dump results to text file* **with** open(RESULT\_FILE,**'a'**) **as** f:  f.write(**'\n'**)  f.write(**'#'**\*20)  f.write(**'\n'**)  f.write(title + **'\n'**)  f.write(response.url)  **for** n **in** range(2): f.write(**'\n'**)  f.write(str(contents))  **for** n **in** range(2): f.write(**'\n'**)  **if** ENABLE\_PARAGRAPH\_STORED:  f.write(para\_str)  f.write(**'\n'**)  f.write(**'#'**\*20)  **for** n **in** range(2): f.write(**'\n'**) |

It dumps the scraped results to a file in the following format as shown in textbox2

Textbox 2

|  |
| --- |
| #################### Healthcare Analytics: Healthcare and big data: Hype or unevenly distributed future? http://www.analytics-magazine.org/november-december-2015/1465-healthcare-analytics-healthcare-and-big-data-hype-or-unevenly-distributed-future  Mathematics, operations research, and statistics to drive better business decisions.  We had a quiet few months in the healthcare analytics industry news wise. Healthcare organizations, however, continued to build and deploy analytics solutions to optimize care, track outcomes and measure cost. Next month, Healthcare Information and Management Systems Society (HIMSS), a not-for-profit organization dedicated to improving healthcare quality, safety, cost and access through the use of information systems, will kick off a big data and healthcare analytics forum in Boston. #################### |

Q. I want the text format to be of the following type

|  |
| --- |
| Unique ID, Keyword, Creation Date, Date of the Article, Author, Category, URL, Subject, Text (Delimiter: |) |

So if I write a code as shown in textbox 3

Textbox 3

|  |
| --- |
| count=0 **with** open(MYFILE,**'a+'**) **as** j:  *#j.write('\n')  #j.write('UniqueID')  #j.write('\n')* j.write(str(count))  j.write(**'|'**)  j.write(**'keyword'**)  j.write(**'|'**)  *#j.write(search\_words)* j.write(title)  j.write(**'|'**)  j.write(response.url)  j.write(**'|'**)  j.write(str(contents))  j.write(**'\n'**) |

The output of this code is shown in textbox 4

Textbox 4

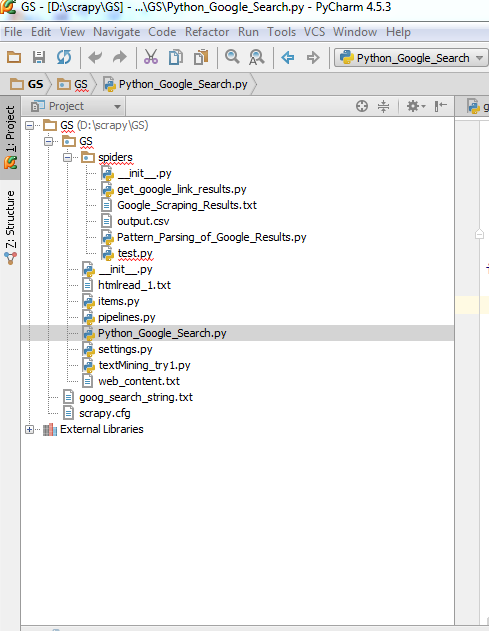
|  |
| --- |
| 0|keyword|Advancing Healthcare through Connected Devices, Big Data and Smart Cities |http://www.firstlinesoftware.com/outsourcery/141-advancing-healthcare|Healthcare and Connected Devices, Big Data and Smart Cities  0|keyword|Healthcare and big data: Drive insights, optimize costs, innovate care | The Big Data Hub|http://www.ibmbigdatahub.com/blog/healthcare-and-big-data-drive-insights-optimize-costs-innovate-care|While healthcare reform and its new laws have ignited a number of important changes, some core issues are not resolved. Applying a preventative big data solution can help avoid these growing pains. |

Problem with the code in textbox 3 are the following;

1. UniqueID is not auto-incrementing?
2. The keyword present in blog post is not being written?
3. Do you know how to extract attributes like post creation date, author, date of article, category, subject from the blog post using scrappy?
4. I am executing this code (Python-google\_Search.py) file on my computer using PyCharm IDE. My question is how to write the text file as shown in textbox4 to a remote hdfs location?

For your reference I have attached the complete code as a separate file.

Place get\_google\_link\_results.py file in the spider folder of scrappy and Python\_Google\_Search.py file outside the spider folder. Please see the attached screenshot for the same. You will need to execute Python\_Google\_Search.py



Thank you for your help.