Your goal for this week is to produce an appropriately formatted catalogue index file with listings for every ebook in the Gutenberg collection. This will involve querying the meta data using RDFLib, in order to extrapolate relevant fields. While it is up to you which fields you include in your index file, you must include as minimum: the id of the ebook in the catalogue, the title of the ebook, the name of the author, and the language of the book. The format of the file is up to you, but it is recommended you structure the data so that it is compatible with the JSON, tsv or csv file formats.

You may find it helpful to refer to the example pgcat.py script that is provided in this week's resources folder. While you will ideally write your own script(s) to produce a catalogue index, as a fall-back, you may use and modify the example.

As well as being an opportunity to practice your new Spark programming skills, this is also a data wrangling exercise, which is something you're likely to do a lot of when working with big data.

**Review criteria**

* Is the file appropriately formatted? (i.e. JSON/tsv/csv compatible structure)
* Is the number of entries in the region one would expect?
* Have they successfully extrapolated the required fields from the meta data? (id, title, author, language)
* Is there evidence that the application was run on a Spark cluster?

SAMPLE OUTPUT:

Here is an example of what you should be aiming for in this assignment (note that this example has been truncated to the first 10 lines!)

{"book": "http://www.gutenberg.org/ebooks/103", "id": 103, "title": "Around the World in Eighty Days", "author": "Verne, Jules", "lang": "en"}

{"book": "http://www.gutenberg.org/ebooks/1038", "id": 1038, "title": "Style", "author": "Raleigh, Walter Alexander, Sir", "lang": "en"}

{"book": "http://www.gutenberg.org/ebooks/10462", "id": 10462, "title": "Clarissa Harlowe; or the history of a young lady \u2014 Volume 4", "author": "Richardson, Samuel", "lang": "en"}

{"book": "http://www.gutenberg.org/ebooks/10586", "id": 10586, "title": "Mike and Psmith", "author": "Wodehouse, P. G. (Pelham Grenville)", "lang": "en"}

{"book": "http://www.gutenberg.org/ebooks/10597", "id": 10597, "title": "Four Early Pamphlets", "author": "Godwin, William", "lang": "en"}

{"book": "http://www.gutenberg.org/ebooks/10688", "id": 10688, "title": "The Camp Fire Girls at Camp Keewaydin; Or, Paddles Down", "author": "Frey, Hildegard G.", "lang": "en"}

{"book": "http://www.gutenberg.org/ebooks/10780", "id": 10780, "title": "Only an Incident", "author": "Litchfield, Grace Denio", "lang": "en"}

{"book": "http://www.gutenberg.org/ebooks/10848", "id": 10848, "title": "Natalie\r Or, A Gem Among the Sea-Weeds", "author": "Vale, Ferna", "lang": "en"}

{"book": "http://www.gutenberg.org/ebooks/10850", "id": 10850, "title": "Philaster; Or, Love Lies a Bleeding", "author": "Fletcher, John", "lang": "en"}

{"book": "http://www.gutenberg.org/ebooks/10880", "id": 10880, "title": "Teddy's Button", "author": "Le Feuvre, Amy", "lang": "en"}

Notes:

 it's not the number of files, it's a memory limitation. I set up swap space (virtual memory) on the master and my job worked beautifully.

# the master will run out of memory for assignment 3,

# so enable some swap space (also known as virtual

# memory)

# this has to be done only once on a started cluster

# before enabling swap will see all 0 values under Swap

free -m

# these commands need to be run only once

# create the swap file of size 512MB

dd if=/dev/zero of=/root/myswapfile bs=1M count=512

# make it only accessible by root (that's us anyway)

chmod 600 /root/myswapfile

# create the swap layout on the file

mkswap /root/myswapfile

# turn it on for swapping

# if you restart the cluster the following

# command is all you need

swapon /root/myswapfile

# after enabling swap will see some non-zero values

# under Swap

free -m