MapReduce

Overview

- Create a Microsoft Azure Notebooks account
  - https://notebooks.azure.com/
- Or, use a local Python distribution
  - Anaconda (if you are using your own machine) https://www.anaconda.com/
- Install MRJob Python library
  - https://github.com/Yelp/mrjob
- Create and test word count locally and deploy script to AWS

Microsoft Azure Notebooks

- Go to: **Sign In**
- Login with your Microsoft account (or create one if you don’t have one)
- Go to **Projects**
- Click on: **+ New Project**
- In Project name put: **Cloud Computing**
- Project ID: **cloudcomputing**
- Press **CREATE**

Great, you now have your library where you can create Jupyter Notebooks and execute Python code.

Local Anaconda Distribution

- Press the START button, type: **Anaconda Prompt** and run it.
- Go to the desired folder, where you would like to have your project files, using **cd**. In my case: **cd Documents\cloud_computing**
- Type **jupyter-notebook** and press enter.

Great, you now have your library where you can create Jupyter Notebooksand execute Python code.

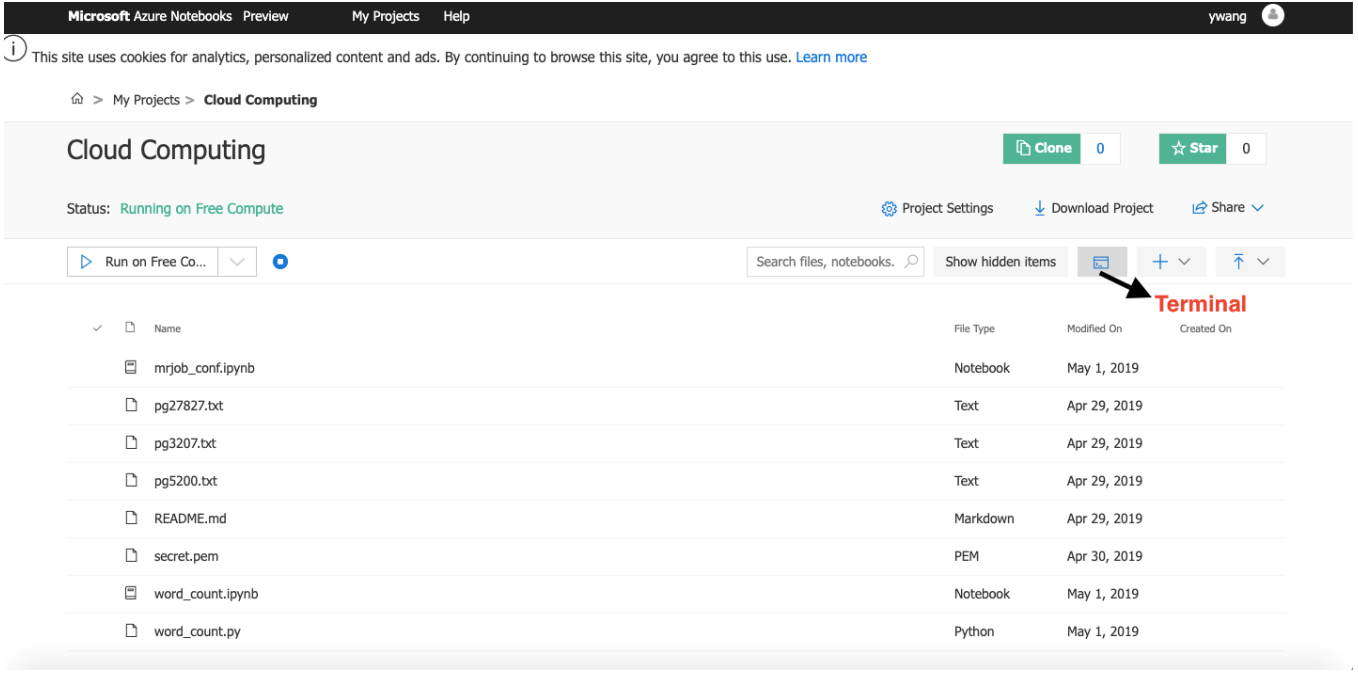
Create your first Jupyter Notebook file

- Azure:
  - Press: **+ NEW >** word_count.ipynb and select Python 3.6 Notebook. Now press **NEW**
  - **You will see the new file below.**

- Jupyter:
  - Go to **NEW (**top right corner) > select **Python 3.** A new tab will open with the file.
  - In the top left file of the newly opened tab, next to the Jupyter logo you will see **Untitled,** click on it and type in **word_count.**
  - You now have your first jupyter notebook. This will be visible in the first tab, from where you have created it.

Installing MRJob on Azure

- In your newly created Project go to **Terminal**, press it and new tab will open. In that new tab, at the command line, type: **pip install mrjob --user** and press enter.



Microsoft Azure Notebooks Preview My Projects Help ywang

Powered by jupyter Cloud Computing

```
nbuser@nbserver:~$ pip install mrjob --user
Collecting mrjob
  Downloading https://files.pythonhosted.org/packages/40/e3/53ee0f4a5791e856065878751fa1959b0a5ea0b20d458c8b6bf28c59020d/mrjob-0.6.8-py2.py3-
none-any.whl (428kB)
    100% |████████████████████████████████| 430kB 2.6MB/s
Collecting google-cloud-storage>=1.13.1 (from mrjob)
  Downloading https://files.pythonhosted.org/packages/9c/aa/048f5b3950f78c9e6afdb05e3667abb7a7ca4463bfde002257acd1874c3f/google_cloud_storage
-1.15.0-py2.py3-none-any.whl (64kB)
    100% |████████████████████████████████| 71kB 6.6MB/s
Collecting google-cloud-logging>=1.9.0 (from mrjob)
  Downloading https://files.pythonhosted.org/packages/a1/56/ea970a90679ff2bd065fb455a0a1b6c810cfc21e3ed674aec68f4f3cd37a/google_cloud_logging
-1.10.0-py2.py3-none-any.whl (112kB)
    100% |████████████████████████████████| 112kB 6.6MB/s
Collecting google-cloud-dataproc>=0.3.0 (from mrjob)
  Downloading https://files.pythonhosted.org/packages/86/9b/30f1e5f55515334b2d897afd19234da53113910ac9fb2d9b2ec128dd60d5/google_cloud_datapro
c-0.3.1-py2.py3-none-any.whl (211kB)
    100% |████████████████████████████████| 215kB 4.9MB/s
Collecting botocore>=1.6.0 (from mrjob)
  Downloading https://files.pythonhosted.org/packages/9b/fc/71ecdeb891a45cce2f873eb1f98324aac82e58d8b81544d46dce936ff6a3/botocore-1.12.139-py
2.py3-none-any.whl (5.4MB)
    100% |████████████████████████████████| 5.4MB 67kB/s
Collecting boto3>=1.4.6 (from mrjob)
  Downloading https://files.pythonhosted.org/packages/1b/6f/36b51dfcc87d8eb7fae7abb5f69c65ac0c062657fd4a4a1e68c3bbe3ea8c/boto3-1.9.139-py2.py
3-none-any.whl (128kB)
    100% |████████████████████████████████| 133kB 7.2MB/s
Requirement already satisfied: PyYAML>=3.10 in /usr/local/lib/python3.5/dist-packages (from mrjob)
Collecting google-cloud-core<0.30dev,>=0.29.0 (from google-cloud-storage>=1.13.1->mrjob)
  Downloading https://files.pythonhosted.org/packages/0c/f2/3c225e7a69cb27d283b68bff867722bd066bc1858611180197f711815ea5/google_cloud_core-0.
29.1-py2.py3-none-any.whl
Collecting google-resumable-media>=0.3.1 (from google-cloud-storage>=1.13.1->mrjob)
  Downloading https://files.pythonhosted.org/packages/e2/5d/4bc5c28c252a62efe69ed1a1561da92bd5af8eca0cdcdf8e60354fae9b29/google_resumable_med
ia-0.3.2-py2.py3-none-any.whl
Collecting google-api-core<2.0.0dev,>=1.6.0 (from google-cloud-storage>=1.13.1->mrjob)
  Downloading https://files.pythonhosted.org/packages/3d/3d/328de10db1b3ec788faa65419727b223b720e9812c9c8660a390b3d56ee9/google_api_core-1.10
.0-py2.py3-none-any.whl (65kB)
```

Installing MRJob locally on Anaconda

- Press the START button, type: **Anaconda Prompt** and run it. Now at the command line prompt type: **pip install mrjob**

Administrator: Anaconda Prompt

```
(base) C:\>pip install mrjob
Requirement already satisfied: mrjob in c:\users\ywang\appdata\roaming\python\python37\site-packages (0.6.8)
Requirement already satisfied: google-cloud-storage>=1.13.1 in c:\users\ywang\appdata\roaming\python\python37\site-packa
ges (from mrjob) (1.15.0)
Requirement already satisfied: PyYAML>=3.10 in c:\users\ywang\appdata\roaming\python\python37\site-packages (from mrjob)
 (3.13)
Requirement already satisfied: google-cloud-dataproc>=0.3.0 in c:\users\ywang\appdata\roaming\python\python37\site-packa
ges (from mrjob) (0.3.1)
Requirement already satisfied: google-cloud-logging>=1.9.0 in c:\users\ywang\appdata\roaming\python\python37\site-packag
es (from mrjob) (1.10.0)
Requirement already satisfied: boto3>=1.4.6 in c:\users\ywang\appdata\roaming\python\python37\site-packages (from mrjob)
 (1.9.138)
Requirement already satisfied: botocore>=1.6.0 in c:\users\ywang\appdata\roaming\python\python37\site-packages (from mrj
ob) (1.12.130)
Requirement already satisfied: google-cloud-core<0.30dev,>=0.29.0 in c:\users\ywang\appdata\roaming\python\python37\site
-packages (from google-cloud-storage>=1.13.1->mrjob) (0.29.1)
Requirement already satisfied: google-resumable-media>=0.3.1 in c:\users\ywang\appdata\roaming\python\python37\site-pack
ages (from google-cloud-storage>=1.13.1->mrjob) (0.3.2)
Requirement already satisfied: google-api-core<2.0.0dev,>=1.6.0 in c:\users\ywang\appdata\roaming\python\python37\site-p
ackages (from google-cloud-storage>=1.13.1->mrjob) (1.10.0)
Requirement already satisfied: s3transfer<0.3.0,>=0.2.0 in c:\users\ywang\appdata\roaming\python\python37\site-packages
(from boto3>=1.4.6->mrjob) (0.2.0)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in c:\users\ywang\appdata\roaming\python\python37\site-packages (f
```

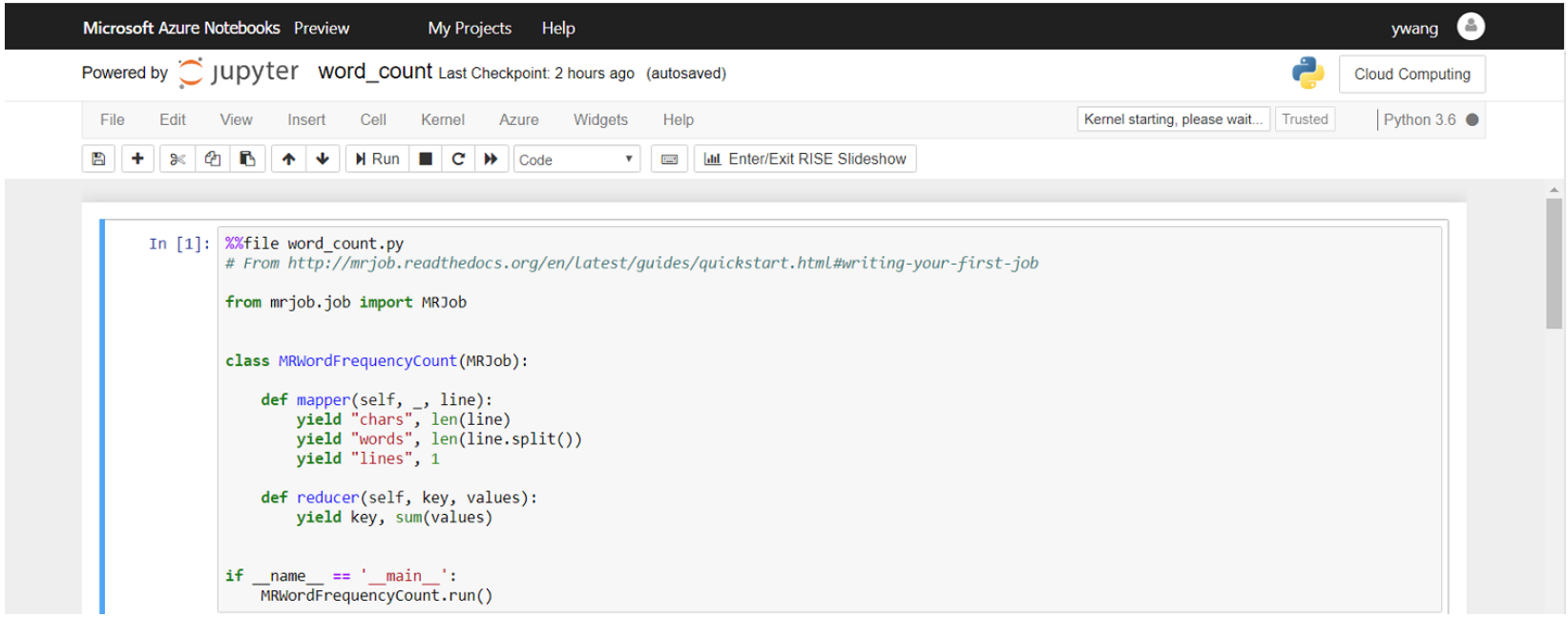
Good job, you can run MapReduce jobs now!

Congrats, this is all you need to run mapreduce jobs locally or in Azure Jupyter notebooks!

Now let’s create and run our first local mapreduce word count program

MRJob hello world

Go to your newly created notebooks (word_count.ipynb) and type the code from this link: http://paws.kettering.edu/~ywang/cs351//word_count.ipynb Go to **Cell > Run Cells.**



Download data to run mapreduce on

Please download the following books in plain text format, which have been sourced from the Gutenberg Project

http://paws.kettering.edu/~ywang/cs351/pg27827.txt

http://paws.kettering.edu/~ywang/cs351/pg3207.txt

http://paws.kettering.edu/~ywang/cs351/pg5200.txt

For Azure you will need to upload them to your project.

Make sure you know the path where you save them as you will need to pass them to your mapreduce program.

Run the MapReduce job

Go back to your word_count notebook, click on the first cell (the one that has all the code inside) and go to: **Insert > Cell Below**. A new cell will be visible below.

Go inside the new cell and type the following:

!python word_count.py -r local *.txt --output-dir=word_count_out --no-output

If you get an error, delete the line and type:

!pip install mrjob

After the installation finishes please try the **!python** line again

That’s it!

If we go back to our notebook dashboard (project in Azure), refresh the page, we will see a new **word_count_out** folder.

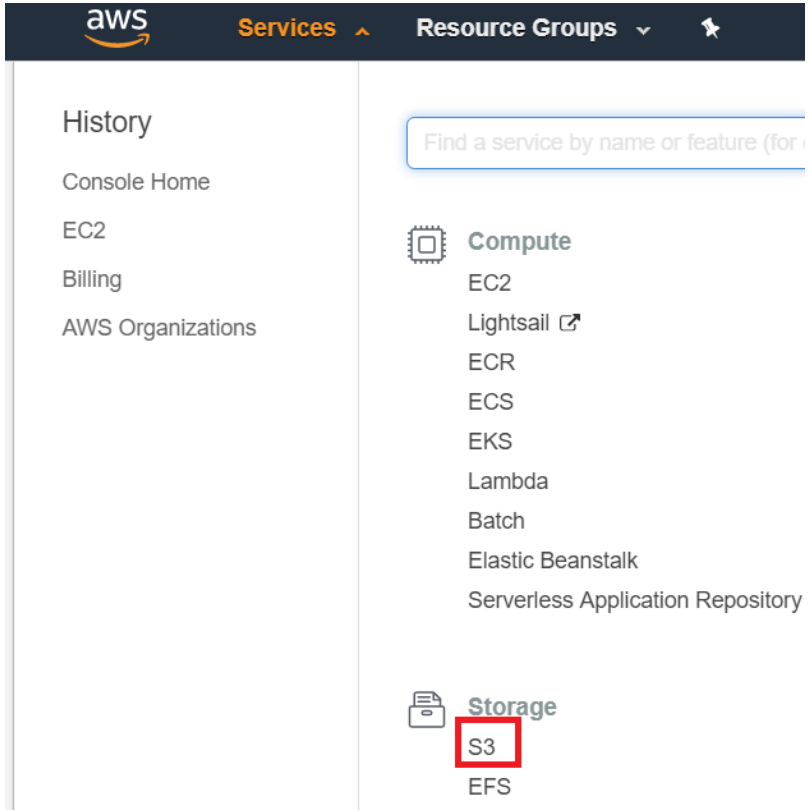
Inside you have the results of your MapReduce script.

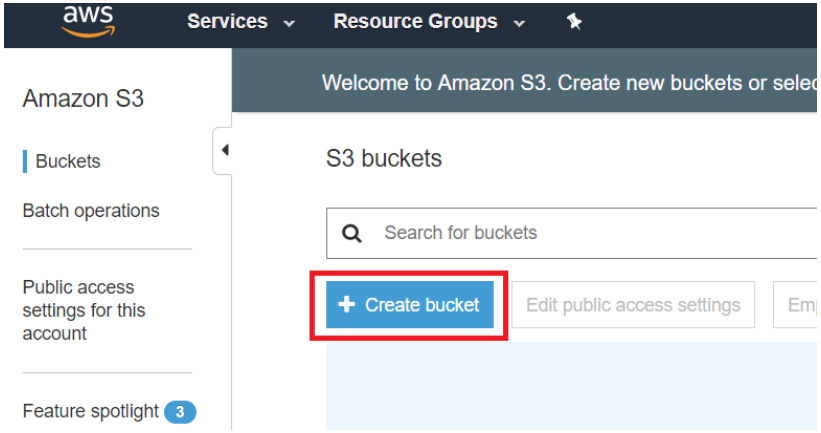
Running the file on AWS

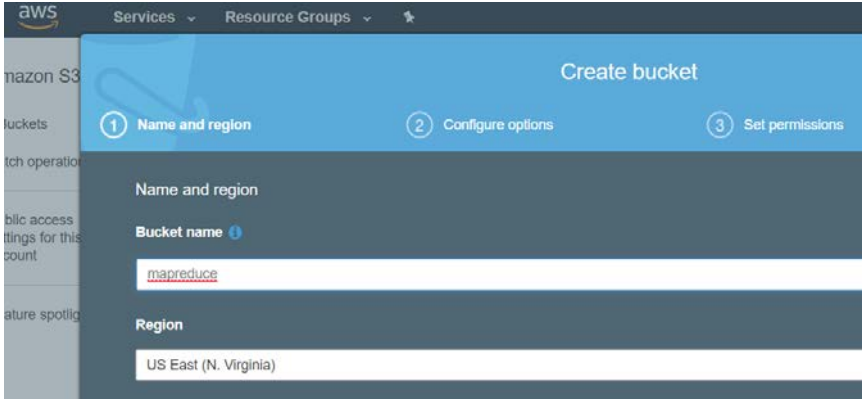
We will need the following:

- Create a aws account
- Create an EC2 Key Pair
- Create S3 storage
  - Upload files to this storage
- Create an MRJob conf file that will automatically create the MapReduce job, execute it and terminate the instances.

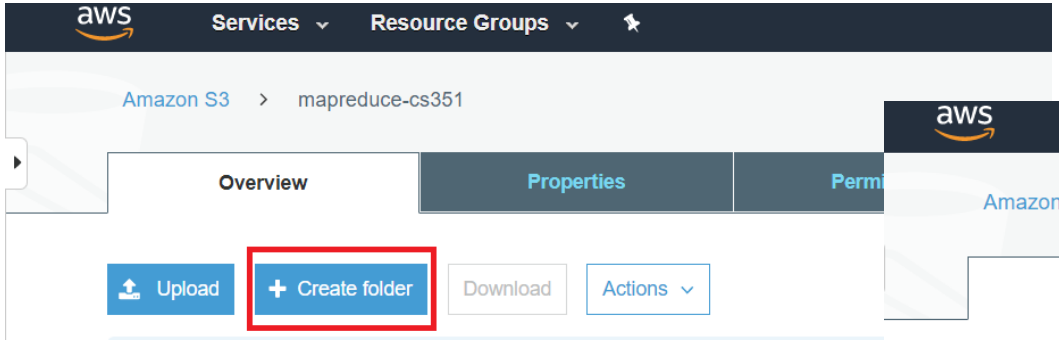
Create a **bucket** under **Storage > S3**

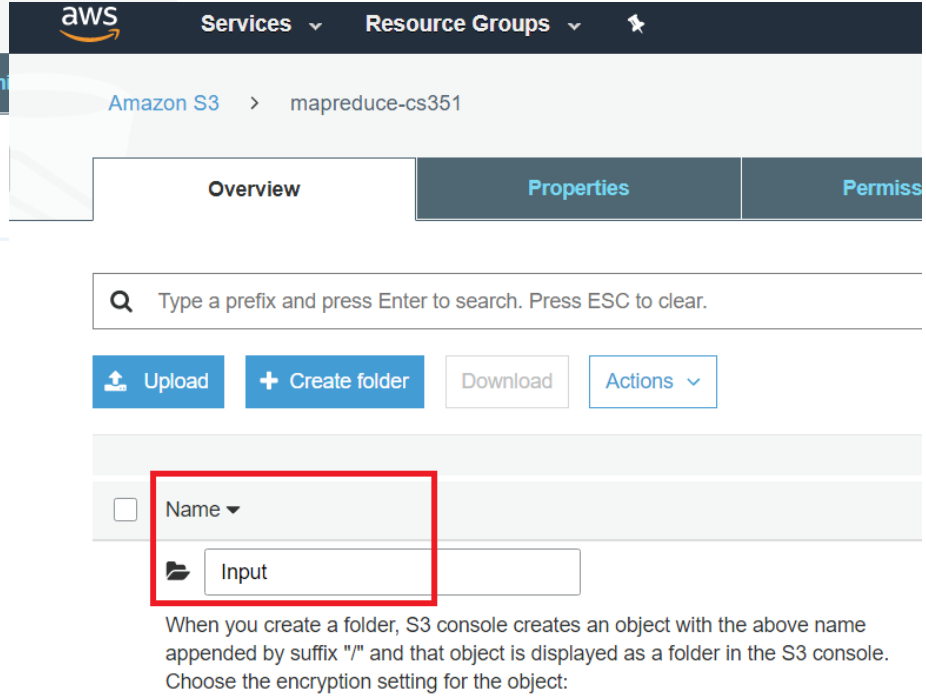






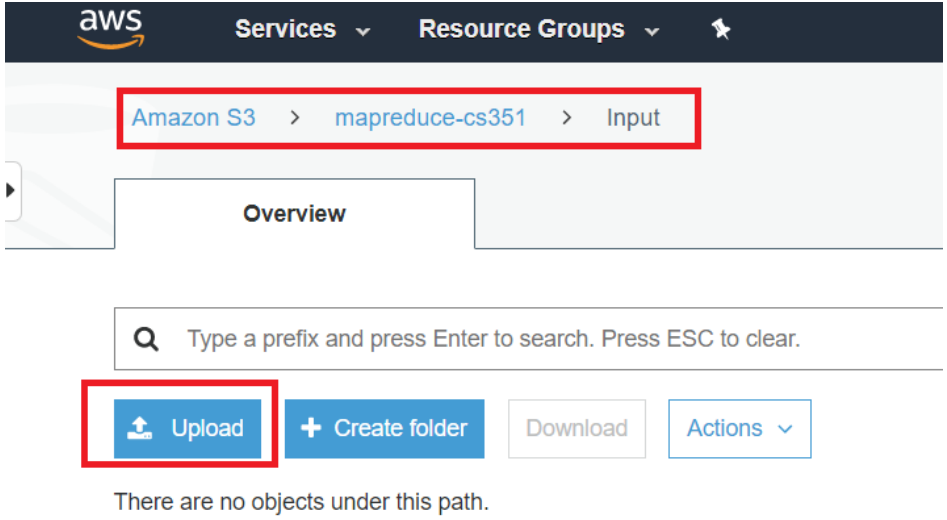
Add folders to the newly created **S3 bucket**



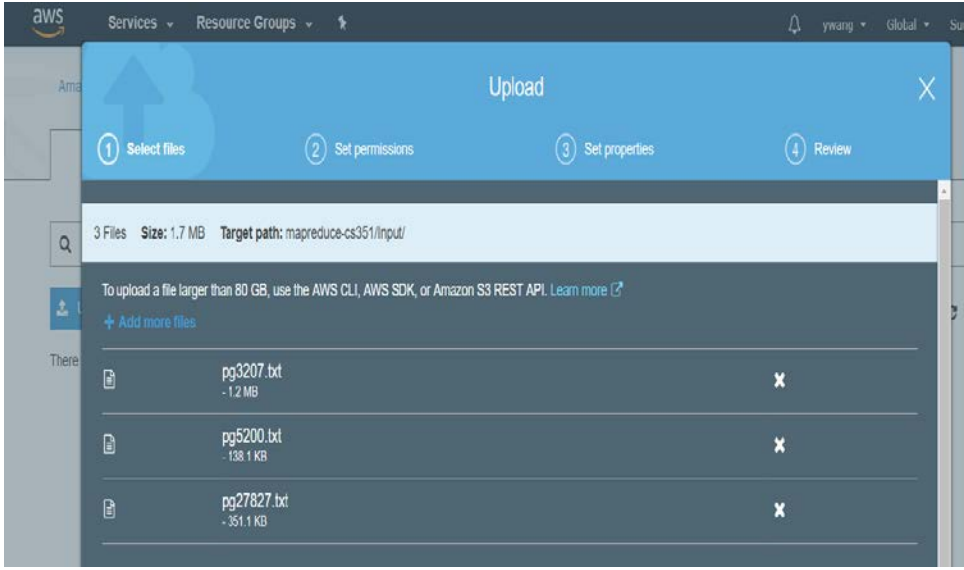


Upload word count books into their **S3 bucket** folder

Select the input folder and press Upload



Add all your input files and upload



Create a new notebook file for the config

In the same way you created the word_count notebook, create a new one with the name **mrjob_conf.ipynb**

It should have the contents from the following link, in its first cell:

http://paws.kettering.edu/~ywang/cs351/mrjob_conf.ipynb

In [1]:
```
%%file ~/.mrjob.conf

# http://mrjob.readthedocs.io/en/stable/guides/emr-opts.html

runners:
  emr:
    aws_access_key_id: AKIAJOB5UIV753BXTQLA
    aws_secret_access_key: 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
    ec2_key_pair: secret
    ec2_key_pair_file: /secret.pem
    region: us-east-1 # http://docs.aws.amazon.com/general/latest/gr/rande.html
    master_instance_type: m5.xlarge # https://aws.amazon.com/emr/pricing/
    instance_type: m5.xlarge
    num_core_instances: 1
    ssh_tunnel: true
```

Getting all the keys to place in your conf

aws_access_key_id and **aws_secret_access_key**

Go to the AWS Console, click on your name (top right corner) and select My Security Credentials. Click on Continue to Security Credentials if it asks.

Go to Access keys (access key ID and secret access key) and click on Create New Access Key. Press on Show Access Key and Copy and Paste your individual access key in it’s appropriate place inside the new mrjob_conf.ipynb tab.

Access Key ID > **aws_access_key_id**

Secret Access Key > **aws_secret_access_key**

Create an EC2 KeyPair

Go to: **Services > EC2**

Select **Key Pairs**, which is under **NETWORK SECURITY** (Left hand side bar).

Press **Create Key Pair** and use any **Key pair name.** Press **Create** after typing name.

Select **Save File,** and save it in **Downloads.**

Get the Key Pair in place

For Azure go to **+ NEW > From Computer > Choose files** and select the downloaded key, it should end in **.pem**

For local Jupyter users make sure you know the location of that file as you will need to update it in the mrjob_conf.ipynb

- **ec2_key_pair**: secure
- **ec2_key_pair_file**: /home/nbuser/library/secure.pem

In my case these are the updates that I need to make.

RUN THE CELL AGAIN!!!

Configuration done!

All done with the MRJob conf, you can now run the cell.

Please have a look at all the configurations that you add to this file:

http://mrjob.readthedocs.io/en/stable/guides/emr-opts.html

You will need to read this if you want to understand what everything does there.

You can control the number of reducers, what types of instances, etc.

Running the word_count on AWS EMR

Go back to your word_count.ipynb tab, click on the **!python** cell and select **Insert > Cell below.**

In the newly created cell, type the following:

**!python word_count.py -r emr s3://mapreduce-cs351/input/*.txt **

**--output-dir=s3://mapreduce-cs351/word_count_out **

--no-output

Make sure that you use your S3 bucket name instead of **mapreduce1-cs351.**

Go to **Cell > Run cells.**

In [6]:
```
!python word_count.py -r emr s3://mapreduce-cs351/input/pg27827.txt \
--output-dir=s3://mapreduce-cs351/word_count_out \
--no-output
```

```
Using configs in /home/nbuser/.mrjob.conf
Auto-created temp S3 bucket mrjob-cd7a34ebf580009c
Using s3://mrjob-cd7a34ebf580009c/tmp/ as our temp dir on S3
Creating temp directory /tmp/word_count.nbuser.20190501.180801.402514
writing master bootstrap script to /tmp/word_count.nbuser.20190501.180801.402514/b.sh
uploading working dir files to s3://mrjob-cd7a34ebf580009c/tmp/word_count.nbuser.20190501.180801.402514/files/wd...
Copying other local files to s3://mrjob-cd7a34ebf580009c/tmp/word_count.nbuser.20190501.180801.402514/files/
Created new cluster j-1M5RZE1QWOLKB
Added EMR tags to cluster j-1M5RZE1QWOLKB: __mrjob_label=word_count, __mrjob_owner=nbuser, __mrjob_version=0.6.8
Waiting for Step 1 of 1 (s-3BSH8WEAIDK5D) to complete...
  PENDING (cluster is STARTING)
  PENDING (cluster is STARTING)
  PENDING (cluster is STARTING)
  PENDING (cluster is STARTING)
  PENDING (cluster is STARTING)
  PENDING (cluster is STARTING)
```

Congratulations!

All done, you have successfully ran your first mapreduce program on AWS.

Jupyter notebook tutorial:

https://www.lynda.com/NumPy-tutorials/Introduction-Jupyter-Notebook/508873/543336-4.html

MRJob:

https://pythonhosted.org/mrjob/

Multistep MRJob:

https://www.youtube.com/watch?v=l_wH6cdcRGQ