Construction of Genome Sets

Goal

The goal of this tutorial is to create a new **Genome Group** in your PATRIC workspace that contains all the *Klebsiella pneumoniae* genomes. (*Note*: feel free to substitute *Klebsiella pneumoniae* for your favorite organism, or try with *Streptococcus pyogenes*, *Streptococcus pneumoniae*, or *Staphylococcus aureus*).

Login

Once you have the `p3` command line tools installed, you will need to login to PATRIC. You can check if you are already logged in (PATRIC does not normally log you out), using the command `p3-whoami`:

```
p3-whoami
You are currently logged out of PATRIC.
```

Now we need to login. *Note:* be sure to replace *username* with your username.

```
p3-login username
Password: ********
Logged in with username username@patricbrc.org
```

and now the `p3-whoami` command will tell you who you are:

```
p3-whoami
You are logged in as PATRIC user username
```

Approach

Our goal is to create a genome group with all the *Klebsiella pneumoniae* genomes.

The command to find organisms is:

```
p3-all-genomes
```

and as with all `p3` commands, you can append `-h` to pull up a help menu:

```
p3-all-genomes -h
```

```
p3-all-genomes.pl [-aefhKr] [long options...]
    -a STR... --attr STR...               field(s) to return
    -K --count                            if specified, a count of
                                          records returned will be
                                          displayed instead of the
                                          records themselves
```

```
-e STR... --eq STR... --equal STR...  search constraint(s) in the
                                      form field_name,value
--lt STR...                           less-than search constraint(s)
                                      in the form field_name,value
--le STR...                           less-or-equal search
                                      constraint(s) in the form
                                      field_name,value
--gt STR...                           greater-than search
                                      constraint(s) in the form
                                      field_name,value
--ge STR...                           greater-or-equal search
                                      constraint(s) in the form
                                      field_name,value
--ne STR...                           not-equal search constraint(s)
                                      in the form field_name,value
--in STR...                           any-value search constraint(s)
                                      in the form
                                      field_name,value1,value2,...,valueN
--keyword STR                         if specified, a keyword or
                                      phrase that shoould be in at
                                      least one field of every record
-r STR... --required STR...           field(s) required to have values
--delim STR                           delimiter to place between
                                      object names
-f --fields                           show available fields
--public                              only include public genomes
--private                             only include private genomes
-h --help                             display usage information
```

Notice that the `-e` option allows us to search for something with an **e**xact match. We are going to search for *Klebsiella pneumoniae*, but what is the name of the field that we need to search?

We can use the `-f` option to show available fields. *Note* that you probably want to pipe this output to less:

```
p3-all-genomes -f | less
```

Alternatively, since we are looking for a field that describes a *name*, we can search for fields that might be appropriate:

```
p3-all-genomes -f | grep name
common_name
genome_name
host_name
organism_name
taxon_lineage_names (multi)
```

Without any other information, I would guess that either `genome_name` or

`organism_name` are the fields that we want. Lets try searching `genome_name` first and seeing what we find:

```
p3-all-genomes -e genome_name,"Klebsiella pneumoniae"
```

There are a couple of things to note here: *First*, the genome name is in quotes because there is a space between *Klebsiella* and *pneumoniae*. You can also replace the space with an underscore like this

```
p3-all-genomes -e genome_name,Klebsiella_pneumoniae
```

and get the same result!

Second, note that the field name (`genome_name`) and the thing we are searching for ("*Klebsiella pneumoniae*") are separated by a comma. This allows us to specify multiple things on the command line.

When I run this, I get a lot of output, but it is (mostly) meaningless genome IDs. Lets add the genome name to this output so we can see what genomes we have. Looking at the help menu above, we see that `-a` is used for the output fields. So lets try:

```
p3-all-genomes -e genome_name,"Klebsiella pneumoniae" -a genome_name | less
```

This lists all the genomes, but if I want to just see a few so I can see if my search is more-or-less correct, I can pipe this output to `p3-head`:

```
p3-all-genomes -e genome_name,"Klebsiella pneumoniae" -a genome_name | p3-head
```

By default, this gives me the first 10 genomes:

genome.genome_id	genome.genome_name
573.18698	Klebsiella pneumoniae strain Klebsiella pneumoniae 1074
573.18697	Klebsiella pneumoniae strain Klebsiella pneumoniae 1041
573.18699	Klebsiella pneumoniae strain Klebsiella pneumoniae 1000
1416011.11	Klebsiella phage F19 strain Klebsiella pneumoniae
1162297.3	Klebsiella pneumoniae subsp. pneumoniae LCT-KP214
1185420.3	Klebsiella pneumoniae subsp. pneumoniae ST258-K28BO
1193292.6	Klebsiella pneumoniae subsp. pneumoniae 1084
1203544.3	Klebsiella pneumoniae subsp. pneumoniae WGLW1
1203545.3	Klebsiella pneumoniae subsp. pneumoniae WGLW2
1203546.5	Klebsiella pneumoniae subsp. pneumoniae WGLW3

This is looking good, and looks like the set of genomes that I want to use to make my **Genome Group** from.

The `p3` command to create a genome group is `p3-put-genome-group`, and this is looking for two inputs: the name of the genome group and a list of IDs to add to that group. Note that in this case we don't need to add the genome name since PATRIC already knows that.

So we pipe the output from our search to `p3-put-genome-group`:

```
p3-all-genomes -e genome_name,"Klebsiella pneumoniae" | p3-put-genome-group "Klebsiella geno
```

How do we know if it worked?

There are two ways that we can test whether we have created a genome group:

First, from the command line, we can retrieve the genome group. If we can do a round-robin, we have successfully created our group:

```
p3-get-genome-group "Klebsiella genomes"
```

This should give you the same list of genomes as your search did!

Second, we can head to the PATRIC website, and specifically head to our *Workspaces*. There is a special workspace called **genome groups**, and if you look in there you should see one called *Klebsiella genomes*. If you click on that, and then click view, you should be able to see all your genomes.