

# March 26 2017

Right now I'm working on a script that will take CSV data that I download from IEDB, and add it to a Python pickle. This pickle will have an object of the IEDBData class, which consist of a dictionary mapping the HLA allele name to a list of tuples, as well as a string representing the IEDB filters. Each tuple maps a peptide to its Kd. I'm calling this script "iedb.py".

Here's the text of the help function:

```
[jforce@jforce TrainingSystems]$ python iedb.py -h
usage: iedb.py [-h] [--listHLA] [--addHLA hla_name csv_file]
               [--showIEDBFilters] [--setIEDBFilters IEDBFilters]
               dataPickle
```

Manipulate the storage of HLA allele data

positional arguments:

dataPickle	Give us the name of the pickle we are working with. If it doesn't exist, then I will create it.
------------	---

optional arguments:

-h, --help	show this help message and exit
--listHLA	Lists the HLA alleles that are already in the pickle (default: False)
--addHLA hla_name csv_file	Adds the HLA with NAME and CSV data to the pickle
--showIEDBFilters	Prints out the IEDB filters that were used to collect data (other than the HLA allele) (default: False)
--setIEDBFilters IEDBFilters	Sets the IEDB filters that were used to collect all of the data in the pickle. You probably shouldn't need to call this more than once. Simply pass in a string with quotes around it. Do not pass the HLA allele name to this

Unfortunately, running Netchop on thousands of proteins takes a long time. One of the things I would do is run 4 at a time, but I kept having issues with the subprocess pipe filling up. So, I made a temporary file to write the output to, but now the output from the call to netchop using the second to last protein seems to get passed to the get\_pieces function. I'm trying to figure out why, but I'm not having much luck.

Start	End	Time	Activity
10:15 a.m.	12:15 a.m.	2:00	Worked on IEDB parse and storage
1:15 p.m.	2:30 p.m.	1:15	Worked on IEDB parse and storage. Looks like the script works.
4:30 p.m.	8:30 p.m.	4:00	Fixed an issue with the netchop script. Now, it will run up to 4 jobs in parallel. Sometimes, when netchop would run, the pipe allocated to it would fill up with the output. Now, I have it writing to a named temporary file