Homework 2 – 130 points

Because students have a wide variety of programming skills in this class, the assignment is based fairly closely on the content. If this is too easy for you, feel free to do some extra exploration with web services or SQLite so you have a chance to learn something new.

For this homework, you'll be **writing 3 Perl scripts (Perl 5)**. The requirements for each are given below.

If you have any questions please email me or post in the Homework 2 discussion topic in the Questions forum.

Functional Requirements

Script 1 (50 pts) will:

1. Read in the file HW2_1.txt and get the UniProt identifiers.
2. Use a regular expression to trim the whitespace from the identifier. (Reuse from Homework 1 if you wish).
3. Check that the UniProt identifier is one letter (upper or lower) followed by 5 numbers. If the UniProt identifier doesn't fit this format, skip it and output an error message. (Note: legal UniProt identifiers do follow other formats, but we'll validate for this one pattern only.) Continue reading in the rest of the file.
4. Use LWP to get the UniProt's XML for each identifier.
5. Parse the UniProt XML to get the name, the recommendedName fullName, and the sequence (the amino acid residue letters).
6. Write the sequence to a text file in fasta format. Put the fullName and name in the first line after the >. Then put the sequence letters after that line. Name the file with the UniProt identifier and .fasta. (For example P11111.fasta).

Script 2 (50 pts) will:

1. Read in the file HW2_2.txt and get the UniProt identifiers. Use the same rules for validating the format as above.
2. Parse the UniProt XML for these identifiers to get the name, recommendedName fullName, and scientific name for the organism.
3. Create an SQLite table with columns: UniProtID, name, recommendedName, organism. If the table or database already exists, keep going.
4. Populate the table with the information from the UniProt records.
5. At the same time, get each UniProt's PDBs but ONLY if the method is X-ray.
6. Create a second table with columns: PDBID, UniProtID and populate it.

7. At the same time, get each UniProt's GO dbReferences. For the GO's get the id, and the term's value.
8. Create a third table with columns: GOID, UniProtID, term.

Script 3 (30 pts) will:

1. Prompt the user to enter a PDBID. You may assume the identifier is well formed.
2. Lookup the PDBID. If it is not in the 2nd table, output an error message.
3. If the PDBID is in the 2nd table, print its UniProt ID, and name.
4. Get all the GOIDs for that UniProtID and print the GOIDs and terms.
5. In your screenshot, show an example where the PDBID is not in the 2nd table.
6. In your screenshot, show an example where the PDBID is in the 2nd table.

Other Requirements

1. The scripts need to be in Perl 5 and use SQLite. No exceptions.
2. They need to run and you must submit evidence (screen shots) of execution. Add a few print statements to log what your code is doing. Also, I will likely also try running your code.
3. Good program structure and comments. I will be reading the code. I must be able to understand it without undue mental effort.
4. You must zip up your scripts, database file, and all files created by your scripts into a zip file with the following naming convention: **your-lastname_HW2.zip**. I will return any submissions with the wrong file name/format and will not grade your homework until it is submitted correctly. Late penalties will apply.

The Programming Assignment is worth **130** points. See the Calendar for the due date. Please plan accordingly, and start early if you think you will have questions or have not yet installed Perl on your system. Also let me know right away if you have questions about what to do regarding the assignment.

You will find the link to upload the assignment in **Module 5**, when the assignment is due. If you need to submit earlier than this, please let me know.

You may use any resource (textbook or web site, not person) you like to do this homework, but please credit in your comments the sources you use.