Module 3 Instructor Exercises

Introduction to Python Programming

The following exercises simulate problems you may see when working with data files.

Read the problems below and examine the figures included. Take time to review the data files and come up with a strategy for addressing the problem.

Format for Submission:

Put all of your code for all of the exercises in a single document formatted using size 10 `Courier New` font.

Fix a busted CSV file

For this exercise, assume you have received the CSV file from a partner. Expecting the file to contain CSV formatted text, you use a simple CSV and find that there is a stray white space character "\n" in the CSV data.

```
0  Case ID, SSN, Birth Date, First Name, Last Name, Address, City, ICD9
1  623047343,390634665,20130309,Andrea                                    ,Rodney
2  ,0632 Colleen Mount                         ,New Amberfurt          ,532
3  285355096,384113106,19741024,Thomas                                   ,Sean
4  ,17362 Rebecca Canyon Apt. 649              ,Port Jeffrey           ,537
5  837915337,044552123,19830527,Tara                                     ,Kelly
6  ,849 Samantha Crossing Apt. 358             ,Port Stephanie         ,553
```
Figure 1 (Indexes added to rows for reference)

You'll notice that each line is broken across two lines. Lines 1 and 2 should be combine to make one full row of data. Similarly, line 3 and 4 are one row though they're broken across two rows in the file we got *from a partner*.

This is a common occurrence with CSV file data, sometimes stray white space causes the file to be non-trivial to process. This exercise is designed to simulate a scenario I've encountered many times. Any beginner Python program should be able to overcome such an issue and produce the desired, more usable CSV file:

```
0  Case ID, SSN, Birth Date, First Name, Last Name, Address, City, ICD9
1  623047343,390634665,20130309,Andrea,Rodney,0632 Colleen Mount,New Amberfurt,532
2  285355096,384113106,19741024,Thomas,Sean,17362 Rebecca Canyon Apt. 649,Port Jeffrey,537
3  837915337,044552123,19830527,Tara,Kelly,849 Samantha Crossing Apt. 358,Port Stephanie,553
```
Figure 2 (Indexes added to rows for reference)

The same data above in a more sane form.

You goal is to take the **digestive_illnesses.csv** file and make it look like Figure 2. Write some python that creates an output file **cleaned_digestive_illnesses.csv.**

Submit your code as instructed.

Build a better data set

For this exercise, assume you have received a CSV file containing a *crosswalk* for various health data codes (HCPCS, ICD10, SNOMED). This file contains records in a format with no header and rows are an indeterminate length.

```
0 SNOMED-CT,38341003,HCPCS,,ICD-10,I10,Hypertension
1 SNOMED-CT,44054006,HCPCS,0403T,ICD-10,E11.9,Diabetes
2 SNOMED-CT,50849002,Emergency room admission
3 SNOMED-CT,308646001,Death Certification
```
Figure 1 (Indexes added to rows for reference)

The general form of each of these rows ends with the description and each row has one or three code types. Really, we want things in more of tabular form. Like this:

```
0 Description, SNOMED-CT, HCPCS, ICD-10
1 Hypertension,38341003,,I10
2 Prediabetes,44054006,0403T,E11.9
3 Emergency room admission,50849002,,
4 Death Certification,308646001,,
```
Figure 2 (Indexes added to rows for reference)

The same data as in Figure 1, in a more sane form.

You goal is to take the **health_codes.csv** file and make it look like Figure 2. Write some python that creates an output file **cleaned_health_codes.csv.**

Submit your code as instructed.