**Assignment 2 - Pandas Introduction**

All questions are weighted the same in this assignment.

**Part 1**

The following code loads the olympics dataset (olympics.csv), which was derrived from the Wikipedia entry on [All Time Olympic Games Medals](https://en.wikipedia.org/wiki/All-time_Olympic_Games_medal_table), and does some basic data cleaning.

The columns are organized as # of Summer games, Summer medals, # of Winter games, Winter medals, total # number of games, total # of medals. Use this dataset to answer the questions below.

import pandas as pd

df = pd.read\_csv('olympics.csv', index\_col=0, skiprows=1)

for col in df.columns:

if col[:2]=='01':

df.rename(columns={col:'Gold'+col[4:]}, inplace=True)

if col[:2]=='02':

df.rename(columns={col:'Silver'+col[4:]}, inplace=True)

if col[:2]=='03':

df.rename(columns={col:'Bronze'+col[4:]}, inplace=True)

if col[:1]=='№':

df.rename(columns={col:'#'+col[1:]}, inplace=True)

names\_ids = df.index.str.split('\s\(') # split the index by '('

df.index = names\_ids.str[0] # the [0] element is the country name (new index)

df['ID'] = names\_ids.str[1].str[:3] # the [1] element is the abbreviation or ID (take first 3 characters from that)

df = df.drop('Totals')

df.head()

**Question 1**

Which country has won the most gold medals in summer games?

*This function should return a single string value.*

**Question 2**

Which country had the biggest difference between their summer and winter gold medal counts?

*This function should return a single string value.*

**Question 3**

Which country has the biggest difference between their summer gold medal counts and winter gold medal counts relative to their total gold medal count?

𝑆𝑢𝑚𝑚𝑒𝑟 𝐺𝑜𝑙𝑑−𝑊𝑖𝑛𝑡𝑒𝑟 𝐺𝑜𝑙𝑑𝑇𝑜𝑡𝑎𝑙 𝐺𝑜𝑙𝑑Summer Gold−Winter GoldTotal Gold

Only include countries that have won at least 1 gold in both summer and winter.

*This function should return a single string value.*

**Question 4**

Write a function that creates a Series called "Points" which is a weighted value where each gold medal (Gold.2) counts for 3 points, silver medals (Silver.2) for 2 points, and bronze medals (Bronze.2) for 1 point. The function should return only the column (a Series object) which you created, with the country names as indices.

*This function should return a Series named Points of length 146*

**Part 2**

For the next set of questions, we will be using census data from the [United States Census Bureau](http://www.census.gov/). Counties are political and geographic subdivisions of states in the United States. This dataset contains population data for counties and states in the US from 2010 to 2015. [See this document](https://www2.census.gov/programs-surveys/popest/technical-documentation/file-layouts/2010-2015/co-est2015-alldata.pdf) for a description of the variable names.

The census dataset (census.csv) should be loaded as census\_df. Answer questions using this as appropriate.

**Question 5**

Which state has the most counties in it? (hint: consider the sumlevel key carefully! You'll need this for future questions too...)

census\_df **=** pd.read\_csv('census.csv')

**Question 6**

**Only looking at the three most populous counties for each state**, what are the three most populous states (in order of highest population to lowest population)? Use CENSUS2010POP.

*This function should return a list of string values.*

**Question 7**

Which county has had the largest absolute change in population within the period 2010-2015? (Hint: population values are stored in columns POPESTIMATE2010 through POPESTIMATE2015, you need to consider all six columns.)

e.g. If County Population in the 5 year period is 100, 120, 80, 105, 100, 130, then its largest change in the period would be |130-80| = 50.

*This function should return a single string value.*

**Question 8**

In this datafile, the United States is broken up into four regions using the "REGION" column.

Create a query that finds the counties that belong to regions 1 or 2, whose name starts with 'Washington', and whose POPESTIMATE2015 was greater than their POPESTIMATE 2014.

*This function should return a 5x2 DataFrame with the columns = ['STNAME', 'CTYNAME'] and the same index ID as the census\_df (sorted ascending by index).*