Python-Pandas Assignment

1. Create a Pandas Dataframe with these columns
   1. City
   2. State
   3. Country
   4. Weather {Sunny, Windy, Cloudy, Raining, Foggy, Snowing, Thundershowers}
   5. GeoPosition {Plain land, Desert land, Near the shore, Mountain region, Hill station}
   6. Season {Summer, Winter, Spring, Autumn}
   7. Create other columns as you deem fit for this exercise

* Populate the dataframe with actual cities, in which state they occur and the country
* Populate Weather and GeoPosition columns also accordingly

We would like to see code that does the following:

1. Select all cities that have similar weather characteristics
2. Select all cities of a particular country and display their weather characteristics

Send your code as a Jupyter notebook, and also the result dataframe

1. Open any PDF document that perform the following operations on the data by creating unique columns in the pandas dataframe.
   1. Read the characters from the PDF to a dataframe. Use any standard pdf reader library to do this conversion
   2. Create a new row for each word in the document
   3. Length of each word
   4. Page number
   5. Linenumber where the word appears
   6. Classify each word/text using Part of Speech tagging
   7. Assume a list of strings are given as input in a dictionary and tag/mark all the rows where the strings are found in the document
      1. Key Tags= [‘string1’, ‘string2’, ‘string3’ etc]
      2. Arrive at a list of about 10 strings that are not commonly found in the document to check your marking logic
   8. Mark two rows above where the text is found and 3 rows below
      1. For eg: If the search string ABCD is found in row with index 150, the expectation is to have a column that marks rows 149, 148, 151, 152 and 153
   9. Mark all rows where you find All caps words [Eg: THIS, WORD, EXAMPLE etc]
   10. Create a line-gram which is a combination of all words in a visual-line in the document

Send your code as a Jupyter notebook, and also the result dataframe

1. Extend example 2 by performing these operations
   1. Create columns that calculate aggregate statistics of the document for key columns like
      1. Pages
      2. Commonly occurring words
      3. Number of NNP POS tags
      4. Number of CD POS tags
   2. Divide the document based on these clusters
      1. Page cluster
         1. Pages < X 🡪 Small
         2. Pages >X and < Y 🡪 Midsize
         3. Pages > Y 🡪 Large
      2. Character cluster
         1. Total chars < X
         2. Chars between X and Y
         3. Total chars > Y
      3. Key characters/Tag cluster
         1. Identify pages that have maximum number of tags