Assignment 3
Introduction to Data Science

Poonam Adhikari *

September 2023

This is an individual assignment. You may not consult your peers or AI tools to do these tasks except where explicitly asked for. Any misconduct will result in a 0 score in this entire assignment and will be noted and reported to the Academic Integrity Committee.

1. In this assignment you are allowed to use numpy and pandas packages.
2. Use the exact same function names as specified in the assignment statement, otherwise the auto grader will fail and you will get a 0 score for that task.
3. Organize your script with a check for main at the bottom (after defining all the functions described below) like this:

```
#if __name__ == '__main__':
    ## Here you may call your functions for testing
    # print(studentinfo('students.csv'))
    pass
```

*

1. Krishna is a Teaching Assistant for the data science course and maintains a CSV file named **"students.csv"**, which contains information about students, including fields like Name, Age, and City. An example of the file structure is displayed in **Table 1**. The task is to write a Python function, **studentinfo(filename)**, that returns the mean age of students from different cities in dictionary format. This function should perform the following checks:

   - Verify the existence of the specified file, raising an error if the file is not found.
   - Ensure that the file contains at least one row; otherwise, raise an error.
   - Remove leading and trailing white spaces from student names and city names.
   - Replace blank (Null) 'city' and 'name' with the most frequently specified 'city' and 'name'.
   - Replace blank (Null) 'Age' with the average.

   .

	Name	Age	City
0	John	28	Chandigarh
1	Alice	24	Mohali
2	Bob	22	Chandigarh
3	Eve	30	Delhi

Table 1: Data frame for question 1

   a. Input: Path of a csv file (filename) with students' detail (one student's information in one row).

   b. Output: A dictionary with city as key and average age as value.

   c. **Example**: Input file: students.csv with information as shown in Table 2 and returns {'Mohali': 25.5, 'Pune': 24.0}

	Name	Age	City
0	Ansh	24	Pune
1	Ankit	26	Mohali
2	Anjana	25	Mohali

Table 2: Example data frame

2. Ankit, an eager and passionate data scientist, is determined to assess whether the field of data science aligns with his expectations. In his quest to evaluate this, he found a dataset on Kaggle, accessible at the following URL: `https://shorturl.at/nyFO5`. Your task is to create a function called **dssalary(filename,expected_salary)** that will assess whether Ankit will find satisfaction in this job role. Ankit has outlined specific requirements, and he anticipates that all of these requirements need to be fulfilled for him to have a happy life ahead.

   - He can work only full-time ('employment_type': FT)
   - He chose his job role as a Data Scientist
   - He is currently located in Canada (CA) however, willing to relocate to the US.
   - He wanted to work only in a large (L) company.

   Your function needs to compute the average salary meeting all the requirements as specified above and check if salary_in_usd is greater than or equal to expected_salary then Ankit will be satisfied with the job otherwise he is unsatisfied. *Note: In case of null values are present in any row remove that row before proceeding further. Remove leading and trailing white spaces if any.*

   a. Input: Path of a text file (filename) and expected_salary.
   b. Output: satisfied or unsatisfied (be careful with case).
   d. Example 1: Input: (path of "ds_salaries.csv",expected_salary = 100000). Output: satisfied
   c. Example 2: Input: (path of "ds_salaries.csv",expected_salary = 155555. Output: unsatisfied

3. Design a function named increment which accepts two parameters: filename set to "ds_salaries.csv" and "salary_raise", a dictionary featuring percentage-based salary increments. Your objective is to modify the columns "salary" and "salary_in_usd" in the specified CSV file, applying the specified increments as outlined in the "salary_raise" dictionary. If a job role is not found in the dictionary, please assume that there will be no salary increase associated with that role.

   a. Input: File path and name, dictionary with salary increment values.
   b. Output: Return the updated dataframe having incremented salary and 'salary_in_usd' column.
   c. Example 1: Path of a csv file (filename) and dictionary having information about salary_raise.

      increment('ds_salaries.csv','AI Developer':10,'Data Analyst':15,'Data Scientist':20
      Output: Return updated dataframe.

4. Create a function called **preprocess(filename)** to convert the "work_year" column into a DateTime format. In this dataset, years like 2023 are interpreted as January 1, 2023, and 2022 as January 1, 2022. Then, select rows with DateTime values between January 1, 2020, and January 1, 2021 (both dates inclusive), and permanently remove the columns "remote ratio" and "experience level," returning the updated DataFrame.

   a. Input: File path and name

   b. Output: Updated dataframe, dropped specified columns and containing "work_year" ranging from January 1, 2020, to January 1, 2021, inclusive.

5. Imagine that, unlike Ankit, you and a group of your friends share a common interest in the realm of data science. Your task create a csv file named buddies.csv which comprises of data shown in Table 3. This CSV file includes information such as the individual's "job_profile", which can be chosen from the following options: Statistics, Machine Learning, Computer Vision, or Natural Language Processing (NLP). Each individual can choose only one "job_profile". Additionally, the CSV has information about the "expected_salary" and "location" as shown in the table below. Secondly, you have to create another file named "salary.csv" **(different from ds_salaries.csv)**, which consists of information about "job_profile", "location", and "company_name" as shown in table 4.

	job_*profile*	expected_salary	location
0	Computer Vision	1800000	Pune
1	Statistics	2000000	Mohali
2	Machine Learning	31000000	Chandigarh
3	Computer Vision	2400000	Chandigarh
4	Natural Language Processing (NLP)	2800000	Pune
5	Natural Language Processing (NLP)	1800000	Mohali

Table 3: Example of buddies.csv file structure

   Your task is to write a function named jobsearch(filepath1,filepath2) that returns a merged dataframe which comprises attributes having "job_profile", "expected_salary", "location", and "company_name" having common "job_profile" and "location" in both the dataframe.
   Hint: You can apply the pandas merge function on these two data frames.

	job_profile	location	company_name
0	Computer Vision	Pune	Microsoft
1	Statistics	Mohali	Zomato
2	Machine Learning	Chandigarh	Zomato
3	Computer Vision	Mohali	Bridging
4	Natural Language Processing (NLP)	Mohali	Bridging
5	Natural Language Processing (NLP)	Pune	Microsoft
6	Computer Vision	Mohali	Zomato

Table 4: Example of salary.csv file structure