Removing Duplicates

Estimated time needed: **30** minutes

Introduction

In this lab, you will focus on data wrangling, an important step in preparing data for analysis. Data wrangling involves cleaning and organizing data to make it suitable for analysis. One key task in this process is removing duplicate entries, which are repeated entries that can distort analysis and lead to inaccurate conclusions.

Objectives

In this lab you will perform the following:

1. Identify duplicate rows in the dataset.
2. Use suitable techniques to remove duplicate rows and verify the removal.
3. Summarize how to handle missing values appropriately.
4. Use ConvertedCompYearly to normalize compensation data.

Install the Required Libraries

In [1]:
```
!pip install pandas
```

```
Requirement already satisfied: pandas in /opt/conda/lib/python3.12/site-packages (2.3.0)
Requirement already satisfied: numpy>=1.26.0 in /opt/conda/lib/python3.12/site-packages (from panda
s) (2.3.0)
Requirement already satisfied: python-dateutil>=2.8.2 in /opt/conda/lib/python3.12/site-packages (fr
om pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /opt/conda/lib/python3.12/site-packages (from pandas)
(2024.2)
Requirement already satisfied: tzdata>=2022.7 in /opt/conda/lib/python3.12/site-packages (from panda
s) (2025.2)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.12/site-packages (from python-date
util>=2.8.2->pandas) (1.17.0)
```

Step 1: Import Required Libraries

In [2]:
```
import pandas as pd
```

Step 2: Load the Dataset into a DataFrame

load the dataset using pd.read_csv()

In [3]:
```
# Define the URL of the dataset
file_path = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/n01PQ9pSmiRX6520flu

# Load the dataset into a DataFrame
df = pd.read_csv(file_path)

# Display the first few rows to ensure it loaded correctly
print(df.head())
```

```
   ResponseId                      MainBranch                 Age  \
0           1  I am a developer by profession  Under 18 years old
1           2  I am a developer by profession     35-44 years old
2           3  I am a developer by profession     45-54 years old
3           4           I am learning to code     18-24 years old
4           5  I am a developer by profession     18-24 years old

            Employment RemoteWork   Check  \
0  Employed, full-time     Remote  Apples
1  Employed, full-time     Remote  Apples
2  Employed, full-time     Remote  Apples
3   Student, full-time        NaN  Apples
4   Student, full-time        NaN  Apples

                                  CodingActivities  \
0                                            Hobby
1  Hobby;Contribute to open-source projects;Other...
2  Hobby;Contribute to open-source projects;Other...
3                                              NaN
4                                              NaN

                                       EdLevel  \
0                       Primary/elementary school
1      Bachelor's degree (B.A., B.S., B.Eng., etc.)
2    Master's degree (M.A., M.S., M.Eng., MBA, etc.)
3  Some college/university study without earning ...
4  Secondary school (e.g. American high school, G...

                                    LearnCode  \
0                           Books / Physical media
1  Books / Physical media;Colleague;On the job tr...
2  Books / Physical media;Colleague;On the job tr...
3  Other online resources (e.g., videos, blogs, f...
4  Other online resources (e.g., videos, blogs, f...

                                  LearnCodeOnline  ... JobSatPoints_6  \
0                                             NaN  ...            NaN
1  Technical documentation;Blogs;Books;Written Tu...  ...            0.0
2  Technical documentation;Blogs;Books;Written Tu...  ...            NaN
3  Stack Overflow;How-to videos;Interactive tutorial  ...            NaN
4  Technical documentation;Blogs;Written Tutorial...  ...            NaN

  JobSatPoints_7 JobSatPoints_8 JobSatPoints_9 JobSatPoints_10  \
0            NaN            NaN            NaN             NaN
1            0.0            0.0            0.0             0.0
2            NaN            NaN            NaN             NaN
3            NaN            NaN            NaN             NaN
4            NaN            NaN            NaN             NaN

  JobSatPoints_11           SurveyLength SurveyEase ConvertedCompYearly JobSat
0             NaN                    NaN        NaN                 NaN    NaN
1             0.0                    NaN        NaN                 NaN    NaN
2             NaN  Appropriate in length       Easy                 NaN    NaN
3             NaN               Too long       Easy                 NaN    NaN
4             NaN             Too short       Easy                 NaN    NaN

[5 rows x 114 columns]
```

Note: If you are working on a local Jupyter environment, you can use the URL directly in the `pandas.read_csv()` function as shown below:

```
df = pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/n01PQ9pSmiRX6520flujwQ/survey-data.csv")
```

Step 3: Identifying Duplicate Rows

Task 1: Identify Duplicate Rows

1. Count the number of duplicate rows in the dataset.
2. Display the first few duplicate rows to understand their structure.

In [4]:
```
## Write your code here
print("--- Task 1: Identify Duplicate Rows ---")

# 1. Count the number of duplicate rows in the dataset.
# The .duplicated() method returns a boolean Series indicating whether each row is a duplicate.
# By default, it marks subsequent duplicates as True.
# sum() on a boolean Series counts the True values.
num_duplicate_rows = df.duplicated().sum()
print(f"Number of duplicate rows in the dataset: {num_duplicate_rows}")
```

```
--- Task 1: Identify Duplicate Rows ---
Number of duplicate rows in the dataset: 0
```

In [5]:
```
## Write your code here
# We use keep=False to mark ALL occurrences of a duplicate row as True.
# Then, we filter the DataFrame to show only these rows.
if num_duplicate_rows > 0:
    print("\nFirst few duplicate rows:")
    print(df[df.duplicated(keep=False)].head())
else:
    print("\nNo duplicate rows found in the dataset.")
```

```
No duplicate rows found in the dataset.
```

Step 4: Removing Duplicate Rows

Task 2: Remove Duplicates

1. Remove duplicate rows from the dataset using the drop_duplicates() function.
2. Verify the removal by counting the number of duplicate rows after removal .

In [6]:
```
## Write your code here
print("\n--- Step 4: Removing Duplicate Rows ---")

# 1. Remove duplicate rows from the dataset using the drop_duplicates() function.
# By default, drop_duplicates() keeps the first occurrence and removes subsequent duplicates.
df_cleaned = df.drop_duplicates()

# 2. Verify the removal by counting the number of duplicate rows after removal.
num_duplicates_after_removal = df_cleaned.duplicated().sum()

print("\nDataFrame after removing duplicate rows:")
print(df_cleaned)
print(f"\nNumber of rows after duplicate removal: {len(df_cleaned)}")
print(f"Number of duplicate rows remaining (should be 0): {num_duplicates_after_removal}")

if num_duplicates_after_removal == 0:
    print("\nVerification successful: All exact duplicate rows have been removed.")
else:
    print("\nVerification failed: Some duplicate rows still exist.")

# If you want to replace your original DataFrame:
# df = df_cleaned
```

```
--- Step 4: Removing Duplicate Rows ---

DataFrame after removing duplicate rows:
       ResponseId                      MainBranch                 Age  \
0               1  I am a developer by profession  Under 18 years old
1               2  I am a developer by profession     35-44 years old
2               3  I am a developer by profession     45-54 years old
3               4           I am learning to code     18-24 years old
4               5  I am a developer by profession     18-24 years old
...           ...                             ...                 ...
65432       65433  I am a developer by profession     18-24 years old
65433       65434  I am a developer by profession     25-34 years old
65434       65435  I am a developer by profession     25-34 years old
65435       65436  I am a developer by profession     18-24 years old
65436       65437     I code primarily as a hobby     18-24 years old

                Employment                                RemoteWork   Check  \
0      Employed, full-time                                    Remote  Apples
1      Employed, full-time                                    Remote  Apples
2      Employed, full-time                                    Remote  Apples
3       Student, full-time                                       NaN  Apples
4       Student, full-time                                       NaN  Apples
...                    ...                                       ...     ...
65432  Employed, full-time                                    Remote  Apples
65433  Employed, full-time                                    Remote  Apples
65434  Employed, full-time                                 In-person  Apples
65435  Employed, full-time  Hybrid (some remote, some in-person)  Apples
65436   Student, full-time                                       NaN  Apples

                                       CodingActivities  \
0                                                 Hobby
1      Hobby;Contribute to open-source projects;Other...
2      Hobby;Contribute to open-source projects;Other...
3                                                   NaN
4                                                   NaN
...                                                 ...
65432                  Hobby;School or academic work
65433        Hobby;Contribute to open-source projects
65434                                             Hobby
65435  Hobby;Contribute to open-source projects;Profe...
65436                                               NaN

                                           EdLevel  \
0                            Primary/elementary school
1        Bachelor’s degree (B.A., B.S., B.Eng., etc.)
2         Master’s degree (M.A., M.S., M.Eng., MBA, etc.)
3      Some college/university study without earning ...
4      Secondary school (e.g. American high school, G...
...                                                 ...
65432    Bachelor’s degree (B.A., B.S., B.Eng., etc.)
65433                                               NaN
65434    Bachelor’s degree (B.A., B.S., B.Eng., etc.)
65435  Secondary school (e.g. American high school, G...
65436                                               NaN

                                         LearnCode  \
0                              Books / Physical media
1      Books / Physical media;Colleague;On the job tr...
2      Books / Physical media;Colleague;On the job tr...
3      Other online resources (e.g., videos, blogs, f...
4      Other online resources (e.g., videos, blogs, f...
...                                                 ...
65432  On the job training;School (i.e., University, ...
65433                                               NaN
65434  Other online resources (e.g., videos, blogs, f...
65435  On the job training;Other online resources (e....
65436                                               NaN

                                      LearnCodeOnline  ... JobSatPoints_6  \
0                                                 NaN  ...            NaN
1      Technical documentation;Blogs;Books;Written Tu...  ...            0.0
2      Technical documentation;Blogs;Books;Written Tu...  ...            NaN
3      Stack Overflow;How-to videos;Interactive tutorial  ...            NaN
4      Technical documentation;Blogs;Written Tutorial...  ...            NaN
...                                                 ...  ...            ...
```

```
65432                                                 NaN  ...           NaN
65433                                                 NaN  ...           NaN
65434  Technical documentation;Stack Overflow;Social ...  ...           NaN
65435  Technical documentation;Blogs;Written Tutorial...  ...           0.0
65436                                                 NaN  ...           NaN

      JobSatPoints_7 JobSatPoints_8 JobSatPoints_9 JobSatPoints_10  \
0                NaN            NaN            NaN             NaN
1                0.0            0.0            0.0             0.0
2                NaN            NaN            NaN             NaN
3                NaN            NaN            NaN             NaN
4                NaN            NaN            NaN             NaN
...              ...            ...            ...             ...
65432            NaN            NaN            NaN             NaN
65433            NaN            NaN            NaN             NaN
65434            NaN            NaN            NaN             NaN
65435            0.0            0.0            0.0             0.0
65436            NaN            NaN            NaN             NaN

      JobSatPoints_11           SurveyLength SurveyEase ConvertedCompYearly  \
0                 NaN                     NaN        NaN                 NaN
1                 0.0                     NaN        NaN                 NaN
2                 NaN  Appropriate in length       Easy                 NaN
3                 NaN               Too long       Easy                 NaN
4                 NaN              Too short       Easy                 NaN
...               ...                     ...        ...                 ...
65432             NaN                     NaN        NaN                 NaN
65433             NaN                     NaN        NaN                 NaN
65434             NaN                     NaN        NaN                 NaN
65435             0.0                     NaN        NaN                 NaN
65436             NaN                     NaN        NaN                 NaN

      JobSat
0        NaN
1        NaN
2        NaN
3        NaN
4        NaN
...      ...
65432    NaN
65433    NaN
65434    NaN
65435    NaN
65436    NaN

[65437 rows x 114 columns]

Number of rows after duplicate removal: 65437
Number of duplicate rows remaining (should be 0): 0

Verification successful: All exact duplicate rows have been removed.
```

Step 5: Handling Missing Values

Task 3: Identify and Handle Missing Values

1. Identify missing values for all columns in the dataset.
2. Choose a column with significant missing values (e.g., EdLevel) and impute with the most frequent value.

In [8]:
```python
## Write your code here
# --- Identify missing values for all columns ---
print("\n--- Identifying Missing Values for All Columns ---")
missing_values_count = df.isnull().sum()
print("\nNumber of missing values per column:")
print(missing_values_count)

# --- Choose a column with significant missing values (e.g., EdLevel) and impute with the most freq
print("\n--- Imputing Missing Values in 'EdLevel' Column ---")

# 1. Get the most frequent value (mode) of the 'EdLevel' column
# .mode()[0] is used because .mode() can return multiple values if there's a tie,
# and we just want the first one.
edlevel_mode = df['EdLevel'].mode()[0]
```

```
print(f"Most frequent value in 'EdLevel' column: '{edlevel_mode}'")

# 2. Impute missing values in 'EdLevel' with its mode
df['EdLevel'].fillna(edlevel_mode, inplace=True)
print(f"Filled missing values in 'EdLevel' with '{edlevel_mode}'.")

# --- Verify the imputation for EdLevel ---
print("\n--- Verifying Imputation for 'EdLevel' ---")
missing_edlevel_after_imputation = df['EdLevel'].isnull().sum()
print(f"Number of missing values in 'EdLevel' after imputation: {missing_edlevel_after_imputation}"

if missing_edlevel_after_imputation == 0:
    print("'EdLevel' column successfully imputed. No missing values remain in this column.")
else:
    print("Warning: 'EdLevel' column still has missing values. Review the data and imputation logic

print("\nDataFrame Head after 'EdLevel' imputation:")
print(df.head(7))
```

```
--- Identifying Missing Values for All Columns ---

Number of missing values per column:
ResponseId                 0
MainBranch                 0
Age                        0
Employment                 0
RemoteWork             10631
                       ...
JobSatPoints_11        35992
SurveyLength            9255
SurveyEase              9199
ConvertedCompYearly    42002
JobSat                 36311
Length: 114, dtype: int64

--- Imputing Missing Values in 'EdLevel' Column ---
Most frequent value in 'EdLevel' column: 'Bachelor's degree (B.A., B.S., B.Eng., etc.)'
Filled missing values in 'EdLevel' with 'Bachelor's degree (B.A., B.S., B.Eng., etc.)'.

--- Verifying Imputation for 'EdLevel' ---
Number of missing values in 'EdLevel' after imputation: 0
'EdLevel' column successfully imputed. No missing values remain in this column.

DataFrame Head after 'EdLevel' imputation:
   ResponseId                                         MainBranch  \
0           1                     I am a developer by profession
1           2                     I am a developer by profession
2           3                     I am a developer by profession
3           4                              I am learning to code
4           5                     I am a developer by profession
5           6                        I code primarily as a hobby
6           7  I am not primarily a developer, but I write co...

                  Age           Employment RemoteWork   Check  \
0  Under 18 years old  Employed, full-time     Remote  Apples
1     35-44 years old  Employed, full-time     Remote  Apples
2     45-54 years old  Employed, full-time     Remote  Apples
3     18-24 years old   Student, full-time        NaN  Apples
4     18-24 years old   Student, full-time        NaN  Apples
5  Under 18 years old   Student, full-time        NaN  Apples
6     35-44 years old  Employed, full-time     Remote  Apples

                                  CodingActivities  \
0                                            Hobby
1  Hobby;Contribute to open-source projects;Other...
2  Hobby;Contribute to open-source projects;Other...
3                                              NaN
4                                              NaN
5                                              NaN
6                      I don't code outside of work

                                          EdLevel  \
0                          Primary/elementary school
1       Bachelor's degree (B.A., B.S., B.Eng., etc.)
2    Master's degree (M.A., M.S., M.Eng., MBA, etc.)
3  Some college/university study without earning ...
4  Secondary school (e.g. American high school, G...
5                          Primary/elementary school
6     Professional degree (JD, MD, Ph.D, Ed.D, etc.)

                                       LearnCode  \
0                           Books / Physical media
1  Books / Physical media;Colleague;On the job tr...
2  Books / Physical media;Colleague;On the job tr...
3  Other online resources (e.g., videos, blogs, f...
4  Other online resources (e.g., videos, blogs, f...
5  School (i.e., University, College, etc);Online...
6  Other online resources (e.g., videos, blogs, f...

                                 LearnCodeOnline  ... JobSatPoints_6  \
0                                            NaN  ...            NaN
1  Technical documentation;Blogs;Books;Written Tu...  ...            0.0
2  Technical documentation;Blogs;Books;Written Tu...  ...            NaN
3  Stack Overflow;How-to videos;Interactive tutorial  ...            NaN
```

```
4  Technical documentation;Blogs;Written Tutorial...  ...           NaN
5                                                NaN  ...           NaN
6  Technical documentation;Stack Overflow;Written...  ...           NaN

  JobSatPoints_7 JobSatPoints_8 JobSatPoints_9 JobSatPoints_10  \
0            NaN            NaN            NaN             NaN
1            0.0            0.0            0.0             0.0
2            NaN            NaN            NaN             NaN
3            NaN            NaN            NaN             NaN
4            NaN            NaN            NaN             NaN
5            NaN            NaN            NaN             NaN
6            NaN            NaN            NaN             NaN

  JobSatPoints_11           SurveyLength                  SurveyEase  \
0             NaN                    NaN                         NaN
1             0.0                    NaN                         NaN
2             NaN  Appropriate in length                        Easy
3             NaN               Too long                        Easy
4             NaN              Too short                        Easy
5             NaN  Appropriate in length                        Easy
6             NaN               Too long  Neither easy nor difficult

  ConvertedCompYearly JobSat
0                 NaN    NaN
1                 NaN    NaN
2                 NaN    NaN
3                 NaN    NaN
4                 NaN    NaN
5                 NaN    NaN
6                 NaN    NaN

[7 rows x 114 columns]
```

```
/tmp/ipykernel_906/2380446697.py:18: FutureWarning: A value is trying to be set on a copy of a DataF
rame or Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate
object on which we are setting values always behaves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, in
place=True)' or df[col] = df[col].method(value) instead, to perform the operation inplace on the ori
ginal object.


  df['EdLevel'].fillna(edlevel_mode, inplace=True)
```

Step 6: Normalizing Compensation Data

Task 4: Normalize Compensation Data Using ConvertedCompYearly

1. Use the ConvertedCompYearly column for compensation analysis as the normalized annual compensation is already provided.
2. Check for missing values in ConvertedCompYearly and handle them if necessary.

In [9]:

```python
## Write your code here
# --- Task 4: Normalize Compensation Data Using ConvertedCompYearly ---
print("\n--- Task 4: Handling Missing Values in 'ConvertedCompYearly' ---")

# Check for missing values in ConvertedCompYearly
missing_count = df['ConvertedCompYearly'].isnull().sum()

if missing_count > 0:
    print(f"\n{missing_count} missing values found in 'ConvertedCompYearly' column.")

    # Handle missing values: Impute with the median
    # Median is often preferred for compensation data to be robust to outliers.
    median_comp = df['ConvertedCompYearly'].median()
    df['ConvertedCompYearly'].fillna(median_comp, inplace=True)
    print(f"Missing values in 'ConvertedCompYearly' filled with median: {median_comp:.2f}")

else:
    print("\nNo missing values found in 'ConvertedCompYearly' column. No handling needed.")

# Verify the absence of missing values after treatment
```

```
missing_after_handling = df['ConvertedCompYearly'].isnull().sum()
print(f"\nMissing values in 'ConvertedCompYearly' (after handling): {missing_after_handling}")

print("\n'ConvertedCompYearly' column after handling missing values:")
print(df['ConvertedCompYearly'])
print("\nDataFrame Info after handling missing values in 'ConvertedCompYearly':")
df['ConvertedCompYearly'].info()
```

```
--- Task 4: Handling Missing Values in 'ConvertedCompYearly' ---

42002 missing values found in 'ConvertedCompYearly' column.
Missing values in 'ConvertedCompYearly' filled with median: 65000.00

Missing values in 'ConvertedCompYearly' (after handling): 0

'ConvertedCompYearly' column after handling missing values:
0        65000.0
1        65000.0
2        65000.0
3        65000.0
4        65000.0
          ...
65432    65000.0
65433    65000.0
65434    65000.0
65435    65000.0
65436    65000.0
Name: ConvertedCompYearly, Length: 65437, dtype: float64

DataFrame Info after handling missing values in 'ConvertedCompYearly':
<class 'pandas.core.series.Series'>
RangeIndex: 65437 entries, 0 to 65436
Series name: ConvertedCompYearly
Non-Null Count  Dtype
--------------  -----
65437 non-null  float64
dtypes: float64(1)
memory usage: 511.4 KB
/tmp/ipykernel_906/3643183075.py:14: FutureWarning: A value is trying to be set on a copy of a DataF
rame or Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate
object on which we are setting values always behaves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, in
place=True)' or df[col] = df[col].method(value) instead, to perform the operation inplace on the ori
ginal object.


  df['ConvertedCompYearly'].fillna(median_comp, inplace=True)
```

Step 7: Summary and Next Steps

In this lab, you focused on identifying and removing duplicate rows.

- You handled missing values by imputing the most frequent value in a chosen column.
- You used ConvertedCompYearly for compensation normalization and handled missing values.
- For further analysis, consider exploring other columns or visualizing the cleaned dataset.

In [ ]:
```
## Write your code here
```

<!-- ## Change Log |Date (YYYY-MM-DD)|Version|Changed By|Change Description| |-|-|-|-| |2024-11-05|1.2|Madhusudhan Moole|Updated lab| |2024-09-24|1.1|Madhusudhan Moole|Updated lab| |2024-09-23|1.0|Raghul Ramesh|Created lab| --!>