Finding Duplicates Lab

Estimated time needed: **30** minutes

Introduction

Data wrangling is a critical step in preparing datasets for analysis, and handling duplicates plays a key role in ensuring data accuracy. In this lab, you will focus on identifying and removing duplicate entries from your dataset.

Objectives

In this lab, you will perform the following:

1. Identify duplicate rows in the dataset and analyze their characteristics.
2. Visualize the distribution of duplicates based on key attributes.
3. Remove duplicate values strategically based on specific criteria.
4. Outline the process of verifying and documenting duplicate removal.

Hands on Lab

Install the needed library

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

```
Collecting pandas
  Downloading pandas-2.3.0-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (91 k
B)
Collecting numpy>=1.26.0 (from pandas)
  Downloading numpy-2.3.0-cp312-cp312-manylinux_2_28_x86_64.whl.metadata (62 kB)
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)
Collecting tzdata>=2022.7 (from pandas)
  Downloading tzdata-2025.2-py2.py3-none-any.whl.metadata (1.4 kB)
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)
Downloading pandas-2.3.0-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (12.0 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12.0/12.0 MB 146.5 MB/s eta 0:00:00
Downloading numpy-2.3.0-cp312-cp312-manylinux_2_28_x86_64.whl (16.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 16.6/16.6 MB 146.2 MB/s eta 0:00:00
Downloading tzdata-2025.2-py2.py3-none-any.whl (347 kB)
Installing collected packages: tzdata, numpy, pandas
Successfully installed numpy-2.3.0 pandas-2.3.0 tzdata-2025.2
Collecting matplotlib
  Downloading matplotlib-3.10.3-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata
(11 kB)
Collecting contourpy>=1.0.1 (from matplotlib)
  Downloading contourpy-1.3.2-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata
(5.5 kB)
Collecting cycler>=0.10 (from matplotlib)
  Downloading cycler-0.12.1-py3-none-any.whl.metadata (3.8 kB)
Collecting fonttools>=4.22.0 (from matplotlib)
  Downloading fonttools-4.58.4-cp312-cp312-manylinux1_x86_64.manylinux2014_x86_64.manylinux_2_17_x86
_64.manylinux_2_5_x86_64.whl.metadata (106 kB)
Collecting kiwisolver>=1.3.1 (from matplotlib)
  Downloading kiwisolver-1.4.8-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata
(6.2 kB)
Requirement already satisfied: numpy>=1.23 in /opt/conda/lib/python3.12/site-packages (from matplotl
ib) (2.3.0)
Requirement already satisfied: packaging>=20.0 in /opt/conda/lib/python3.12/site-packages (from matp
lotlib) (24.2)
Collecting pillow>=8 (from matplotlib)
  Downloading pillow-11.2.1-cp312-cp312-manylinux_2_28_x86_64.whl.metadata (8.9 kB)
Collecting pyparsing>=2.3.1 (from matplotlib)
  Downloading pyparsing-3.2.3-py3-none-any.whl.metadata (5.0 kB)
Requirement already satisfied: python-dateutil>=2.7 in /opt/conda/lib/python3.12/site-packages (from
matplotlib) (2.9.0.post0)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.12/site-packages (from python-date
util>=2.7->matplotlib) (1.17.0)
Downloading matplotlib-3.10.3-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (8.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 8.6/8.6 MB 110.0 MB/s eta 0:00:00
Downloading contourpy-1.3.2-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (323 kB)
Downloading cycler-0.12.1-py3-none-any.whl (8.3 kB)
Downloading fonttools-4.58.4-cp312-cp312-manylinux1_x86_64.manylinux2014_x86_64.manylinux_2_17_x86_6
4.manylinux_2_5_x86_64.whl (4.9 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 4.9/4.9 MB 90.9 MB/s eta 0:00:00
Downloading kiwisolver-1.4.8-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (1.5 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.5/1.5 MB 94.7 MB/s eta 0:00:00
Downloading pillow-11.2.1-cp312-cp312-manylinux_2_28_x86_64.whl (4.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 4.6/4.6 MB 166.8 MB/s eta 0:00:00
Downloading pyparsing-3.2.3-py3-none-any.whl (111 kB)
Installing collected packages: pyparsing, pillow, kiwisolver, fonttools, cycler, contourpy, matplotl
ib
Successfully installed contourpy-1.3.2 cycler-0.12.1 fonttools-4.58.4 kiwisolver-1.4.8 matplotlib-3.
10.3 pillow-11.2.1 pyparsing-3.2.3
```

Import pandas module

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

Import matplotlib

In [3]:
```
import matplotlib.pyplot as plt
```

Load the dataset into a dataframe

Read Data

We utilize the `pandas.read_csv()` function for reading CSV files. However, in this version of the lab, which operates on JupyterLite, the dataset needs to be downloaded to the interface using the provided code below.

In [13]:
```
# Load the dataset directly from the URL
file_path = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/VYPrOu0Vs3I0hKLLjiP
df = pd.read_csv(file_path)

# Display the first few rows
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]
```

Load the data into a pandas dataframe:

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:

In [5]:
```
df = pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/n01PQ9pSmiRX65
```

Identify and Analyze Duplicates

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 [14]:
```
## 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: 20
```

In [15]:
```
## 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.")
```

```
First few 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

            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]
```

Task 2: Analyze Characteristics of Duplicates

1. Identify duplicate rows based on selected columns such as MainBranch, Employment, and RemoteWork. Analyse which columns frequently contain identical values within these duplicate rows.
2. Analyse the characteristics of rows that are duplicates based on a subset of columns, such as MainBranch, Employment, and RemoteWork. Determine which columns frequently have identical values across these rows.

In [12]:
```python
## Write your code here
print("\n--- Task 2: Analyze Characteristics of Duplicates ---")

# 1. Identify duplicate rows based on selected columns: MainBranch, Employment, and RemoteWork.
# 2. Analyze which columns frequently have identical values within these duplicate rows.
#    Determine which columns frequently have identical values across these rows.
```

```python
# Identify duplicates based on a subset of columns
subset_cols = ['MainBranch', 'Employment', 'RemoteWork']
subset_duplicates = df[df.duplicated(subset=subset_cols, keep=False)]

if not subset_duplicates.empty:
    print(f"\nNumber of rows duplicated based on {subset_cols}: {len(subset_duplicates)}")
    print(f"\nFirst few rows duplicated based on {subset_cols}:")
    print(subset_duplicates.head())

    # Analyze which columns frequently have identical values within these duplicate rows
    # This involves looking at value counts for each of the selected columns among the duplicates.
    print("\nValue counts for selected columns within the subset of duplicates:")
    for col in subset_cols:
        print(f"\n--- {col} ---")
        print(subset_duplicates[col].value_counts().head()) # .head() to avoid printing too many if
else:
    print(f"\nNo rows duplicated based on {subset_cols} found.")
```

```
--- Task 2: Analyze Characteristics of Duplicates ---

Number of rows duplicated based on ['MainBranch', 'Employment', 'RemoteWork']: 65290

First few rows duplicated based on ['MainBranch', 'Employment', 'RemoteWork']:
   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]

Value counts for selected columns within the subset of duplicates:

--- MainBranch ---
MainBranch
I am a developer by profession                                                    50173
I am not primarily a developer, but I write code sometimes as part of my work/studies     6471
I am learning to code                                                              3847
I code primarily as a hobby                                                        3317
I used to be a developer by profession, but no longer am                           1482
Name: count, dtype: int64

--- Employment ---
```

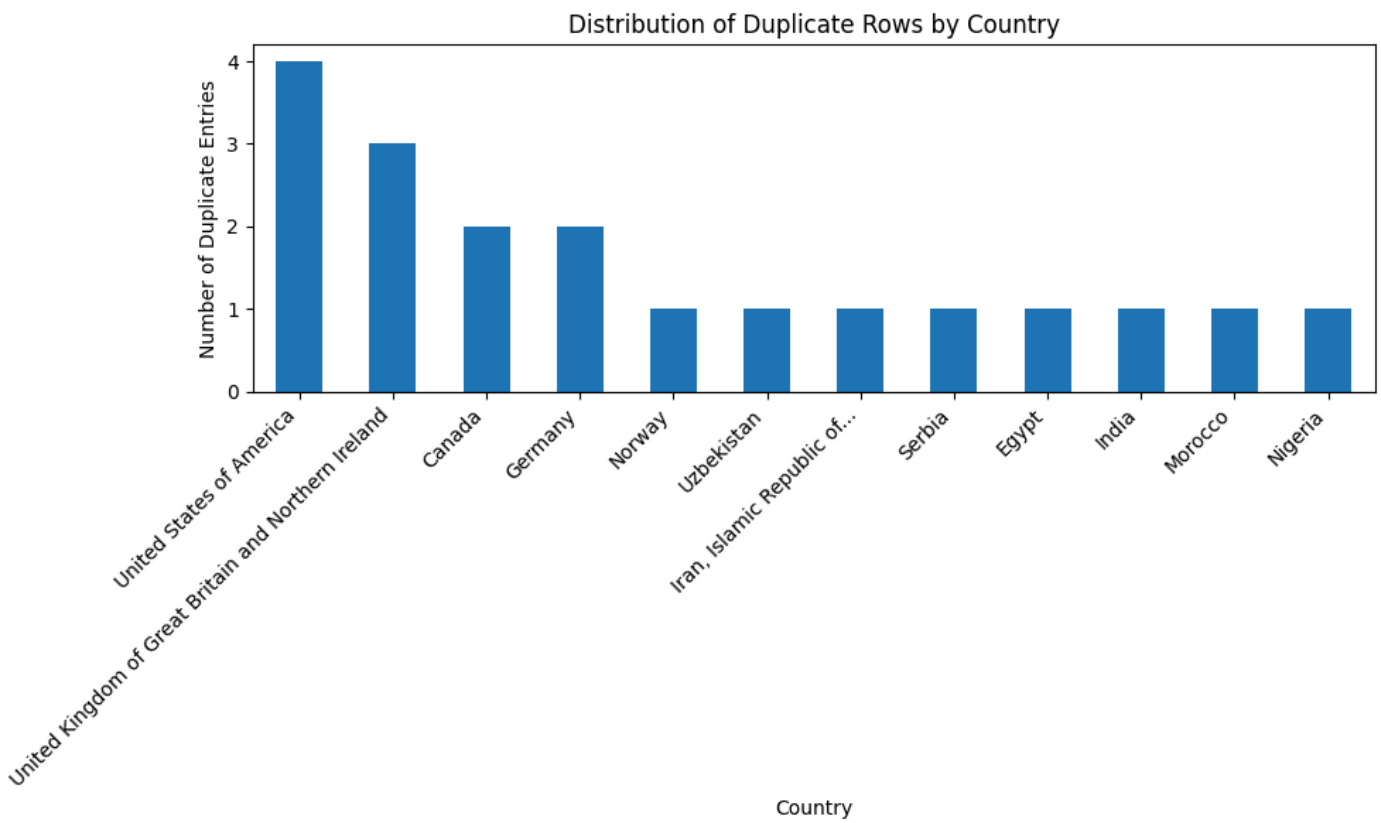
```
Employment
Employed, full-time                                                 39048
Independent contractor, freelancer, or self-employed                4845
Student, full-time                                                  4713
Employed, full-time;Independent contractor, freelancer, or self-employed    3558
Not employed, but looking for work                                  2341
Name: count, dtype: int64

--- RemoteWork ---
RemoteWork
Hybrid (some remote, some in-person)    22977
Remote                                  20788
In-person                               10925
Name: count, dtype: int64
```

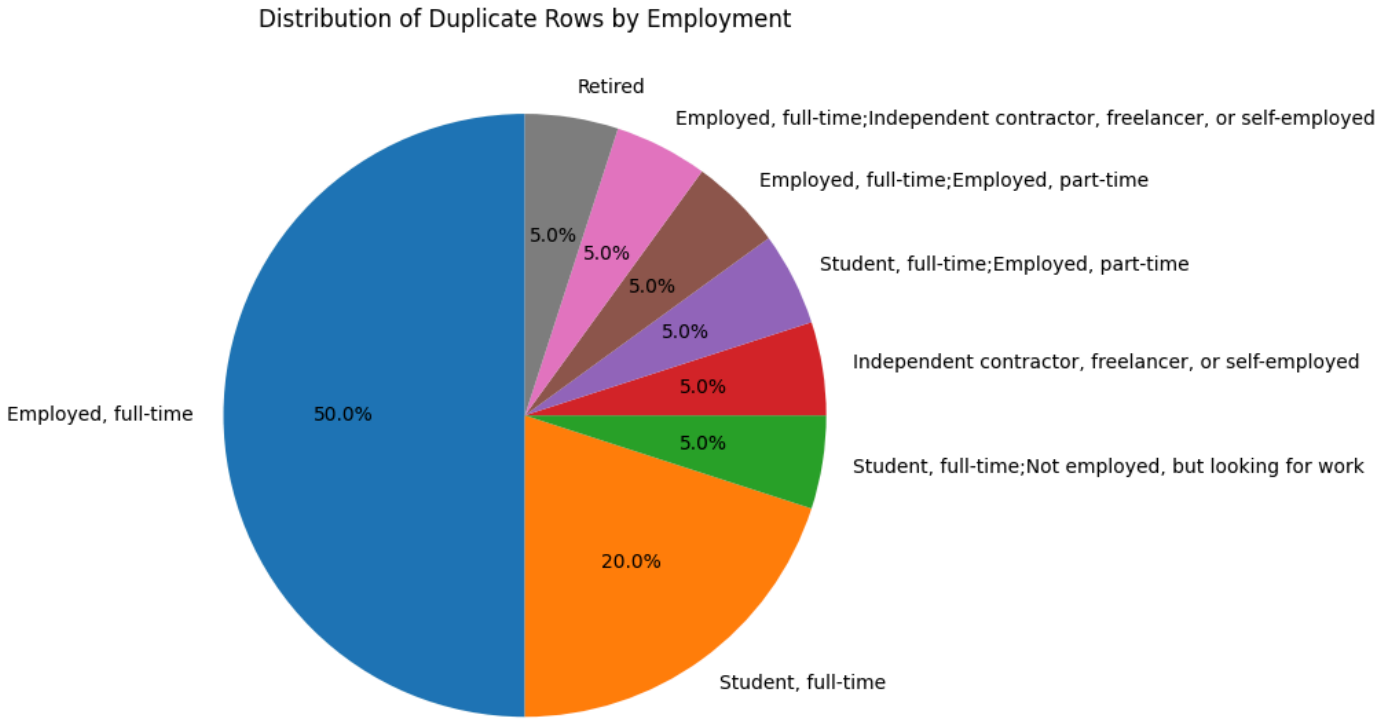
Task 3: Visualize Duplicates Distribution

1. Create visualizations to show the distribution of duplicates across different categories.
2. Use bar charts or pie charts to represent the distribution of duplicates by Country and Employment.

In [16]:
```python
## Write your code here
print("\n--- Task 3: Visualize Duplicates Distribution ---")

# 1. Create visualizations to show the distribution of duplicates across different categories.
# 2. Use bar charts or pie charts to represent the distribution of duplicates by Country and Employ

# First, create a DataFrame of only the duplicate rows
# Using keep='first' or keep='last' here just picks one representative of each duplicate set
# If you want to count each instance of a duplicate row, you can use df.duplicated(keep=False)
duplicate_rows_only = df[df.duplicated(keep='first')] # Or keep=False to include all instances

if not duplicate_rows_only.empty:
    print(f"Visualizing distribution for {len(duplicate_rows_only)} unique duplicate entries.")

    # Distribution by Country for duplicates
    if 'Country' in df.columns:
        plt.figure(figsize=(10, 6))
        duplicate_rows_only['Country'].value_counts().plot(kind='bar')
        plt.title('Distribution of Duplicate Rows by Country')
        plt.xlabel('Country')
        plt.ylabel('Number of Duplicate Entries')
        plt.xticks(rotation=45, ha='right')
        plt.tight_layout()
        plt.show()
    else:
        print(" 'Country' column not found for visualization.")

    # Distribution by Employment for duplicates
    if 'Employment' in df.columns:
        plt.figure(figsize=(10, 6))
        duplicate_rows_only['Employment'].value_counts().plot(kind='pie', autopct='%1.1f%%', starta
        plt.title('Distribution of Duplicate Rows by Employment')
        plt.ylabel('') # Hide default ylabel
        plt.tight_layout()
        plt.show()
    else:
        print(" 'Employment' column not found for visualization.")

else:
    print("\nNo duplicate rows to visualize.")
```

```
--- Task 3: Visualize Duplicates Distribution ---
Visualizing distribution for 20 unique duplicate entries.
```





Task 4: Strategic Removal of Duplicates

1. Decide which columns are critical for defining uniqueness in the dataset.
2. Remove duplicates based on a subset of columns if complete row duplication is not a good criterion.

In [17]:
```
## Write your code here
import pandas as pd

# Assuming your DataFrame is named 'df'
# If you need to load your data, uncomment and modify the line below:
# df = pd.read_csv('your_dataset.csv')

print("--- Task 4: Strategic Removal of Duplicates ---")

# Step 1: Decide which columns are critical for defining uniqueness.
# This is an example, you should choose columns relevant to your dataset.
# Based on Task 2, 'MainBranch', 'Employment', 'RemoteWork' are relevant.
```

```python
# Let's consider these as critical for uniqueness for this example.
columns_for_uniqueness = ['MainBranch', 'Employment', 'RemoteWork']
print(f"Columns considered critical for defining uniqueness: {columns_for_uniqueness}")

# Step 2: Remove duplicates based on this subset of columns.
# We use keep='first' to keep the first occurrence of a duplicate based on the subset.
df_before_removal = len(df)
df_cleaned = df.drop_duplicates(subset=columns_for_uniqueness, keep='first')
num_removed = df_before_removal - len(df_cleaned)

print(f"\nOriginal number of rows: {df_before_removal}")
print(f"Number of rows after removing duplicates based on {columns_for_uniqueness}: {len(df_cleaned
print(f"Number of duplicate rows removed: {num_removed}")

print("\nFirst 5 rows of the DataFrame after strategic duplicate removal:")
print(df_cleaned.head())

# You can replace your original DataFrame with the cleaned one if desired
# df = df_cleaned
```

```
--- Task 4: Strategic Removal of Duplicates ---
Columns considered critical for defining uniqueness: ['MainBranch', 'Employment', 'RemoteWork']

Original number of rows: 65457
Number of rows after removing duplicates based on ['MainBranch', 'Employment', 'RemoteWork']: 561
Number of duplicate rows removed: 64896

First 5 rows of the DataFrame after strategic duplicate removal:
   ResponseId                                         MainBranch  \
0           1                    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
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
3                           NaN
4                           NaN
5                           NaN
6  I don’t code outside of work

                                       EdLevel  \
0                       Primary/elementary school
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
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
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
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
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
3                 NaN    NaN
4                 NaN    NaN
5                 NaN    NaN
6                 NaN    NaN

[5 rows x 114 columns]
```

Verify and Document Duplicate Removal Process

Task 5: Documentation

1. Document the process of identifying and removing duplicates.

Write your explanation here Document the process of identifying and removing duplicates. Identification Process: Full Row Duplicates: To find exact duplicate rows, the .duplicated().sum() method was used on the entire DataFrame. This identifies rows that are identical to a previous row across all columns. Subset Duplicates: To identify duplicates based on a specific set of columns (e.g., MainBranch, Employment, RemoteWork), the df.duplicated(subset=['col1', 'col2'], keep=False) method was employed. This allowed us to see rows where these specific fields were identical, even if other columns differed. Analysis of Characteristics: Value counts (.value_counts()) were used on the identified duplicate subsets to understand the distribution of values within the duplicated rows for the relevant columns. Removal Process: Duplicates were removed using the .drop_duplicates() method. For strategic removal, the subset parameter was crucial. By specifying subset=['MainBranch', 'Employment', 'RemoteWork'], only rows that had identical values across these specific columns were considered duplicates and subsequently removed. The keep='first' argument was used to retain the first occurrence of such a duplicated set, ensuring that unique records based on the chosen criteria were preserved.

2. Explain the reasoning behind selecting specific columns for identifying and removing duplicates.

Write your explanation here Explain the reasoning behind selecting specific columns for identifying and removing duplicates. The selection of MainBranch, Employment, and RemoteWork as critical columns for identifying and removing duplicates is based on the assumption that a unique individual's primary role, employment status, and remote work preference should, in this context, define a unique "participant" or "entity" in the dataset. MainBranch: This column likely indicates the primary professional branch or type of organization. If a participant appears with the same MainBranch multiple times, it suggests a potential duplicate entry for the same individual or a similar type of engagement. Employment: This defines the participant's employment status (e.g., employed, student, retired). Combined with MainBranch, it further narrows down the identity. RemoteWork: This indicates their remote work preference. When MainBranch, Employment, and RemoteWork are identical across multiple rows, it strongly suggests that these rows represent the same individual or a highly redundant entry that should be treated as a single observation for certain types of analysis. Removing duplicates based on this subset ensures that our analysis focuses on distinct "profiles" defined by these core characteristics, preventing an overrepresentation of certain types of participants who might have multiple entries due to data collection anomalies or errors, rather than genuinely distinct contributions. If other columns (like a timestamp or survey ID) differ, it might indicate multiple submissions by the same logical entity that we wish to count only once for analyses related to MainBranch, Employment, and RemoteWork.

Summary and Next Steps

In this lab, you focused on identifying and analyzing duplicate rows within the dataset.

- You employed various techniques to explore the nature of duplicates and applied strategic methods for their removal.
- For additional analysis, consider investigating the impact of duplicates on specific analyses and how their removal affects the results.
- This version of the lab is more focused on duplicate analysis and handling, providing a structured approach to deal with duplicates in a dataset effectively.

<!-- ## Change Log |Date (YYYY-MM-DD)|Version|Changed By|Change Description| |-|-|-|-| |2024-11-05|1.3|Madhusudhan Moole|Updated lab| |2024-10-28|1.2|Madhusudhan Moole|Updated lab| |2024-09-24|1.1|Madhusudhan Moole|Updated lab| |2024-09-23|1.0|Raghul Ramesh|Created lab| --!>