

```
import pandas as pd

# Load the dataset
data = pd.read_csv('task1.csv', delimiter=',', on_bad_lines='skip',
engine='python')

# Display column names to identify any hidden characters
print(data.columns.tolist())

['Data Source', 'World Development Indicators', 'Unnamed: 2']

# Try skipping the first few rows to locate actual data
data = pd.read_csv('task1.csv', delimiter=',', skiprows=5,
on_bad_lines='skip', engine='python') # Adjust 'skiprows' as needed
print(data.head()) # Check if this loads the actual data
```

	Aruba	ABW	Population, total	SP.POP.TOTL	\
0	Africa Eastern and Southern	AFE	Population, total	SP.POP.TOTL	
1	Afghanistan	AFG	Population, total	SP.POP.TOTL	
2	Africa Western and Central	AFW	Population, total	SP.POP.TOTL	
3	Angola	AGO	Population, total	SP.POP.TOTL	
4	Albania	ALB	Population, total	SP.POP.TOTL	

	54608	55811	56682	57475	58178	\
0	130692579.0	134169237.0	137835590.0	141630546.0	145605995.0	
1	8622466.0	8790140.0	8969047.0	9157465.0	9355514.0	
2	97256290.0	99314028.0	101445032.0	103667517.0	105959979.0	
3	5357195.0	5441333.0	5521400.0	5599827.0	5673199.0	
4	1608800.0	1659800.0	1711319.0	1762621.0	1814135.0	

	58782	...	104257	104874	105439	
105962	\					
0	149742351.0	...	600008424.0	616377605.0	632746570.0	
649757148.0						
1	9565147.0	...	33753499.0	34636207.0	35643418.0	
36686784.0						
2	108336203.0	...	408690375.0	419778384.0	431138704.0	
442646825.0						
3	5736582.0	...	28127721.0	29154746.0	30208628.0	
31273533.0						
4	1864791.0	...	2880703.0	2876101.0	2873457.0	
2866376.0						

	106442	106585	106537	106445	106277	\
0	667242986.0	685112979.0	702977106.0	720859132.0	739108306.0	
1	37769499.0	38972230.0	40099462.0	41128771.0	42239854.0	
2	454306063.0	466189102.0	478185907.0	490330870.0	502789511.0	
3	32353588.0	33428486.0	34503774.0	35588987.0	36684202.0	
4	2854191.0	2837849.0	2811666.0	2777689.0	2745972.0	

Unnamed: 68

0	NaN
1	NaN
2	NaN
3	NaN
4	NaN

[5 rows x 69 columns]

```
import pandas as pd
import matplotlib.pyplot as plt
```

```
# Display column names to verify the available columns
print(data.columns)
```

```
# Replace 'year_column' with an actual year column name from the
output of the previous line.
# Check for any additional whitespaces or unexpected characters if
'54608' does not appear as expected.
```

```
year_column = '54608' # Replace with the exact column name if
different
```

```
# Verify if the year_column exists after inspecting the column names
if year_column in data.columns:
```

```
    # Drop NaN values and select relevant data for plotting
    filtered_data = data[['Data Source', year_column]].dropna()
    filtered_data.columns = ['Region', 'Population'] # Rename for
clarity
```

```
    # Create a bar chart for population distribution for the chosen
year
```

```
    plt.figure(figsize=(12, 8))
    plt.bar(filtered_data['Region'], filtered_data['Population'],
color='skyblue')
    plt.title(f'Population Distribution in {year_column}')
    plt.xlabel('Region/Country')
    plt.ylabel('Population')
    plt.xticks(rotation=90)
    plt.tight_layout()
    plt.show()
```

```
else:
    print(f"Column '{year_column}' not found in data columns.")
```

```
Index(['Data Source', 'World Development Indicators', 'Unnamed: 2'],
dtype='object')
Column '54608' not found in data columns.
```

```
import pandas as pd
import matplotlib.pyplot as plt
```

```

# Assuming 'data' is your DataFrame, if it's not, load your dataset
first
# Example: data = pd.read_csv('your_file.csv')

# Check the columns in your dataset
print(data.columns)

# Replace 'Gender' with the actual column name for gender in your
dataset
gender_column = 'Gender' # Replace with the exact column name for
gender

# Verify if the 'Gender' column exists after inspecting the column
names
if gender_column in data.columns:
    # Drop NaN values and count the occurrences of each category
    gender_counts = data[gender_column].dropna().value_counts()

    # Create a bar chart for gender distribution
    plt.figure(figsize=(10, 6))
    gender_counts.plot(kind='bar', color='skyblue', edgecolor='black')
    plt.title('Gender Distribution')
    plt.xlabel('Gender')
    plt.ylabel('Frequency')
    plt.xticks(rotation=0)
    plt.tight_layout()
    plt.show()
else:
    print(f"Column '{gender_column}' not found in data columns.")

```

```

Index(['Data Source', 'World Development Indicators', 'Unnamed: 2'],
      dtype='object')
Column 'Gender' not found in data columns.

```

```

import pandas as pd
import matplotlib.pyplot as plt

# Assuming 'data' is your DataFrame
# Example: data = pd.read_csv('your_file.csv')

# Display the first few rows to inspect the data
print(data.head())

# Replace 'World Development Indicators' with the correct column name
for numeric data
column_to_plot = 'World Development Indicators' # Replace with actual
numeric column

# Verify if the column exists and it's numeric

```

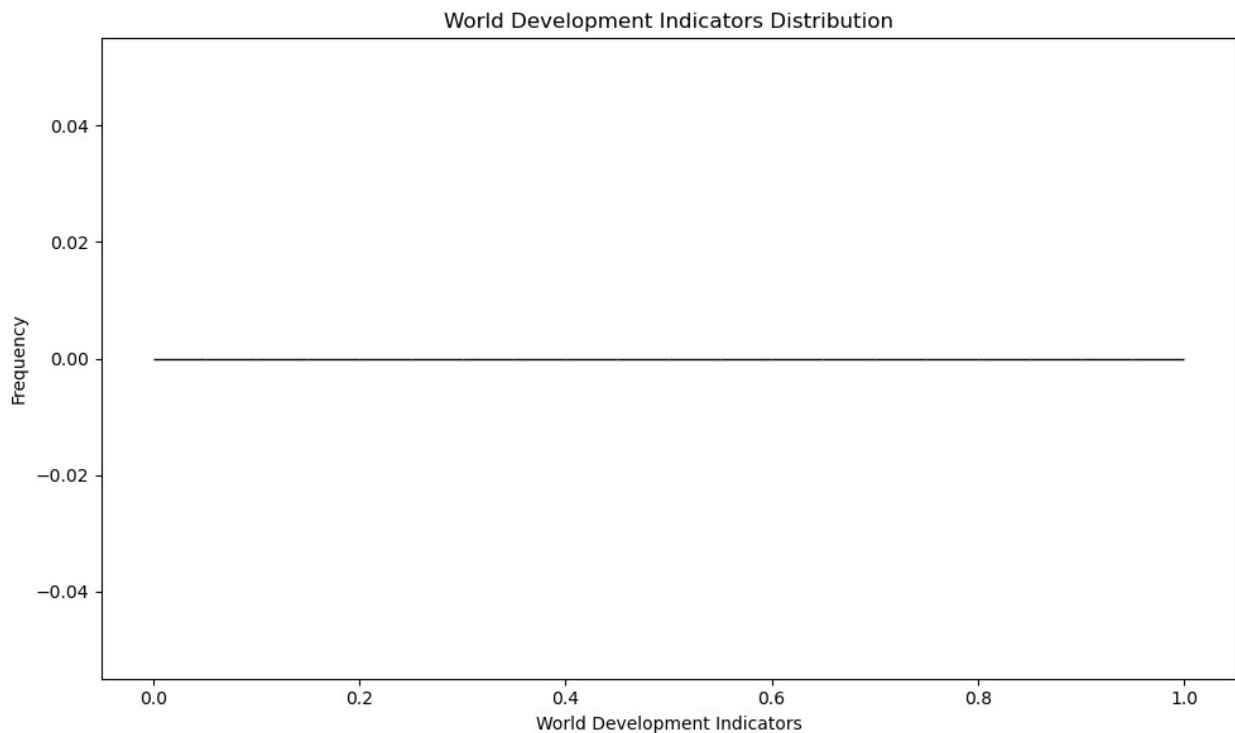
```

if column_to_plot in data.columns:
    # Ensure the column has numeric data (if not, convert it)
    filtered_data = pd.to_numeric(data[column_to_plot],
errors='coerce').dropna() # Convert to numeric and drop NaNs

    # Create a histogram
    plt.figure(figsize=(10, 6))
    plt.hist(filtered_data, bins=20, color='skyblue',
edgecolor='black')
    plt.title(f'{column_to_plot} Distribution')
    plt.xlabel(column_to_plot)
    plt.ylabel('Frequency')
    plt.tight_layout()
    plt.show()
else:
    print(f"Column '{column_to_plot}' not found in data columns.")

```

	Data Source	World Development Indicators	Unnamed: 2
0	Last Updated Date	2024-11-13	NaN



```

import pandas as pd
import matplotlib.pyplot as plt

print(data.head())

# Replace 'Data Source' with the actual column name for categories

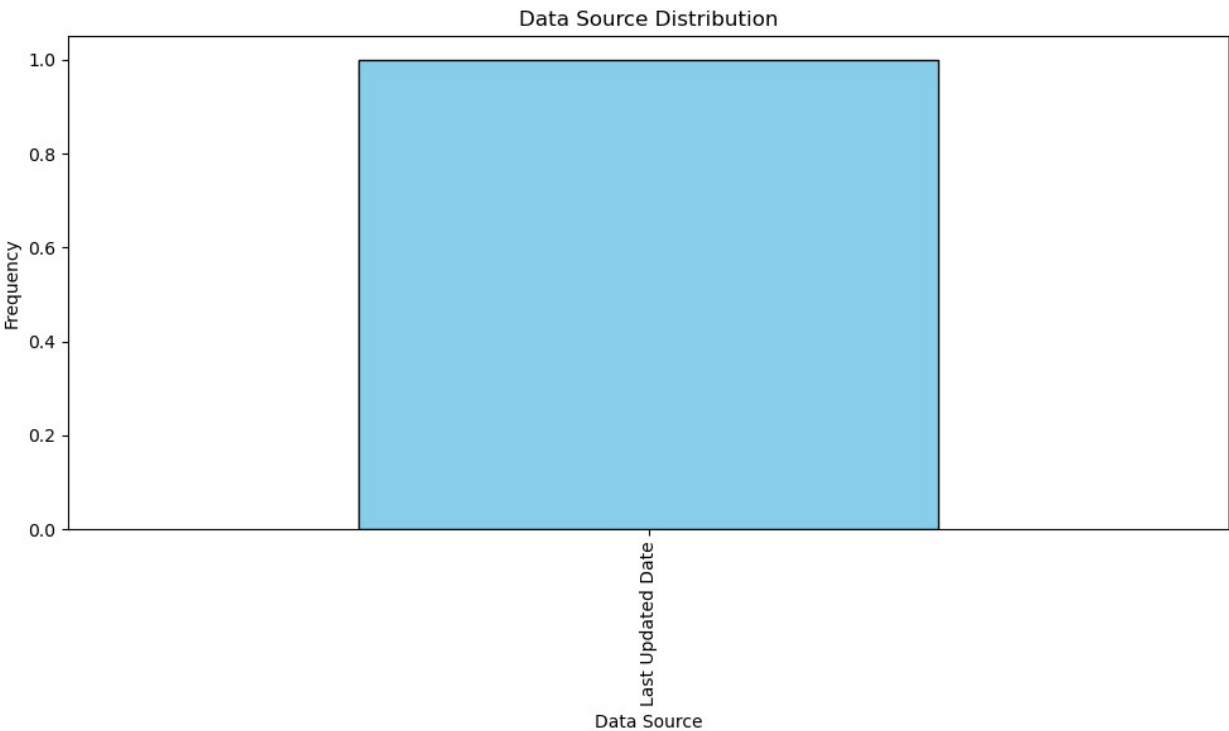
```

```
column_to_plot = 'Data Source' # Replace with the actual categorical
column name

# Verify if the column exists
if column_to_plot in data.columns:
    # Count the occurrences of each category
    category_counts = data[column_to_plot].dropna().value_counts()

    # Create a bar chart for category distribution
    plt.figure(figsize=(10, 6))
    category_counts.plot(kind='bar', color='skyblue',
edgecolor='black')
    plt.title(f'{column_to_plot} Distribution')
    plt.xlabel(column_to_plot)
    plt.ylabel('Frequency')
    plt.xticks(rotation=90) # Rotate the x-axis labels if needed
    plt.tight_layout()
    plt.show()
else:
    print(f"Column '{column_to_plot}' not found in data columns.")
```

	Data Source	World Development Indicators	Unnamed: 2
0	Last Updated Date	2024-11-13	NaN



--	--	--	--