Problem Set 2 - Advanced Pandas Manipulations

1. Advanced Pivoting and Aggregation

Medium

You are provided with a sales dataset:

```
df = pd.DataFrame({
    'Region': ['North', 'North', 'South', 'South', '
      ↪ East', 'East'],
    'Product': ['A', 'B', 'A', 'B', 'A', 'B'],
    'Year': [2023, 2023, 2023, 2023, 2024, 2024],
    'Quarter': ['Q1', 'Q2', 'Q1', 'Q2', 'Q1', 'Q2'],
    'Sales': [100, 150, 200, 250, 300, 350]
})
```

Pivot this dataset such that: - Rows are `Region` and `Year`. - Columns are `Quarter`. - Values are the sum of `Sales` for each combination.

Hard

Using the same dataset, create a pivot table where: - Rows are `Region`. - Columns are `Year` and `Quarter`. - Values are the sum of `Sales`. - Add margins to display the total sales for each `Region` and overall.

Advanced

You are tasked with analyzing the performance of each product. Create a pivot table that: - Rows are `Product`. - Columns are a multi-level index of `Region` and `Year`. - Values are the percentage contribution of each product's sales to the region's total sales for the year.

—

2. Advanced Melting and Reshaping

Medium

You are given the following DataFrame:

```
df = pd.DataFrame({
    'Country': ['USA', 'USA', 'India', 'India'],
    'Year': [2022, 2023, 2022, 2023],
    'Population': [330, 335, 1400, 1420],
    'GDP': [21.5, 22.0, 2.9, 3.2]
})
```

Melt this DataFrame so that: - It has columns: `Country`, `Year`, `Metric`, and `Value`. - `Metric` should differentiate between `Population` and `GDP`.

Hard

Using the melted DataFrame from the Medium question, reshape it back into a wide format where: - Rows are `Country` and `Year`. - Columns are `Population` and `GDP`.

Advanced

Extend the melted DataFrame from the Medium question by: 1. Creating a new column `Metric Type` that categorizes `Population` as `Demographic` and `GDP` as `Economic`. 2. Reshape the DataFrame into a format where: - Rows are `Country` and `Year`. - Columns are multi-indexed with `Metric Type` and `Metric`. - Values are the corresponding `Value`.

—

3. Multi-Level Index Manipulation

Medium

You have a multi-indexed DataFrame:

```
df = pd.DataFrame({
    'Sales': [100, 150, 200, 250],
    'Profit': [20, 40, 50, 70]
}, index=pd.MultiIndex.from_tuples(
    [('North', 'Q1'), ('North', 'Q2'), ('South', 'Q1
    ↪ '), ('South', 'Q2')],
```

```
    names=['Region', 'Quarter']
))
```

Unstack the `Quarter` level so that it becomes columns, and calculate the `Profit Margin` (`Profit / Sales`) for each `Region` and `Quarter`.

Hard

Using the same DataFrame, normalize the `Sales` values for each region such that they sum to 1, while keeping the multi-index intact.

Advanced

You are tasked with reshaping and analyzing the data further: 1. Stack the DataFrame back to long format with columns: `Region`, `Quarter`, `Metric`, and `Value`. 2. Filter the DataFrame to only include rows where the `Metric` is `Profit` and the `Value` is above 50.

—

4. Method Chaining and Advanced Transformations

Medium

You have a dataset of orders:

```
df = pd.DataFrame({
    'OrderID': [1, 2, 3, 4],
    'Customer': ['Alice', 'Bob', 'Alice', 'Bob'],
    'Amount': [100, 200, 300, 400],
    'Date': ['2024-01-01', '2024-01-02', '2024-01-03
      ↪ ', '2024-01-04']
})
```

Using method chaining: 1. Group the data by `Customer`. 2. Calculate the total `Amount` spent by each customer. 3. Sort the results in descending order of `Amount`.

Hard

You are analyzing sales trends. Using the same dataset: 1. Extract the `Year` and `Month` from the `Date`. 2. Group by `Year` and `Month` to calculate the

total `Amount`. 3. Add a column that shows the percentage contribution of each month's `Amount` to the total sales for the year.

Advanced

Extend the analysis from the Hard question: 1. Identify the month with the highest `Amount` for each `Year`. 2. Return a DataFrame with `Year`, `Month`, and `Total Amount` for the identified months.

—