In Power BI's **Power Query Editor**, the **Split Column**, **Format**, and **Extract** options are essential tools for transforming and manipulating data to fit analysis needs. Here's a deep dive into each of these, explaining every detail and functionality.

---

1. Split Column

The **Split Column** option in Power Query Editor allows you to divide a column into multiple columns based on specific delimiters or positions within the text. This is especially useful when working with concatenated data or data that includes structured information like dates, addresses, or names.

Options for Split Column:

1. **By Delimiter**:
   - **Functionality**: Splits the column based on a specific character or string, such as a comma, space, hyphen, or any custom delimiter.
   - **Example**: If you have a column with full names like "John Doe," splitting by space will separate "John" and "Doe" into two columns.

   Steps:
   - Select the column to split.
   - Go to **Transform** > **Split Column** > **By Delimiter**.
   - Choose a delimiter (e.g., comma, space) or specify a custom one.
   - Choose where to apply the split (e.g., each occurrence, left-most, or right-most delimiter).
2. **By Number of Characters**:
   - **Functionality**: Divides the column into segments based on a specified number of characters.
   - **Example**: For a column with phone numbers like "1234567890," splitting by 3 characters could yield columns "123," "456," "789," and "0."

   Steps:
   - Select the column.
   - Go to **Transform** > **Split Column** > **By Number of Characters**.
   - Enter the number of characters for each split segment.
   - Decide whether to split once or repeatedly by that number of characters.
3. **By Positions**:
   - **Functionality**: Splits the column at specific positions within the text.
   - **Example**: If you have product codes like "ABC12345," you could split after the first 3 characters to separate the "ABC" and "12345" parts.

   Steps:
   - Select the column.
   - Go to **Transform** > **Split Column** > **By Positions**.
   - Specify the positions at which you want the splits to occur.

4. **By Uppercase to Lowercase and Vice Versa**:

   - **Functionality**: Splits text based on changes between uppercase and lowercase characters.
   - **Example**: For a column like "JohnDoe," this split would yield "John" and "Doe."

---

2. Format

The **Format** option allows you to adjust the appearance of data in a column, mainly text-based, by applying consistent styling transformations. Formatting options help clean up inconsistent data, such as names with mixed case or spaces, making it easier to analyze and maintain data uniformity.

Available Format Options:

1. **Lowercase**:

   - **Functionality**: Converts all characters in a column to lowercase.
   - **Example**: "John DOE" becomes "john doe."
   - **Use Case**: Useful for normalizing text data to avoid case-sensitive discrepancies.
2. **Uppercase**:

   - **Functionality**: Converts all characters to uppercase.
   - **Example**: "john doe" becomes "JOHN DOE."
   - **Use Case**: Commonly used for codes or IDs that need to be uniformly capitalized.
3. **Capitalize Each Word**:

   - **Functionality**: Capitalizes the first letter of each word.
   - **Example**: "john doe" becomes "John Doe."
   - **Use Case**: Ideal for formatting names, titles, or addresses where capitalization is required for readability.
4. **Trim**:

   - **Functionality**: Removes any leading or trailing spaces in the text.
   - **Example**: " John Doe " becomes "John Doe."
   - **Use Case**: Useful for eliminating accidental spaces that can cause matching issues.
5. **Clean**:

   - **Functionality**: Removes non-printable characters from text.
   - **Example**: "John\u0000Doe" becomes "JohnDoe."
   - **Use Case**: Ensures the text is readable and free from hidden characters, often imported from external sources.
6. **Add Prefix** and **Add Suffix**:

   - **Functionality**: Adds specified text before (prefix) or after (suffix) each value in a column.

- **Example**: Adding "Mr." as a prefix to "John Doe" changes it to "Mr. John Doe."
- **Use Case**: Great for adding labels or clarifying information, like appending "USD" to currency amounts.

---

3. **Extract**

The **Extract** function allows you to pull specific parts of text from a column based on patterns or positions. It's useful for isolating relevant parts of data, such as extracting area codes from phone numbers or initials from names.

Extract Options:

1. **Extract First Characters**:
   - **Functionality**: Extracts the first n characters from each entry in a column.
   - **Example**: Extracting the first 3 characters from "ABC123" results in "ABC."
   - **Use Case**: Commonly used for codes or identifiers where only the initial part is relevant.

   Steps:
   - Select the column.
   - Go to **Transform** > **Extract** > **First Characters**.
   - Specify the number of characters to extract.
2. **Extract Last Characters**:
   - **Functionality**: Extracts the last n characters.
   - **Example**: From "ABC123," extracting the last 3 characters results in "123."
   - **Use Case**: Useful for suffix-based data or when identifying specific elements at the end of strings, such as file extensions.
3. **Extract Range**:
   - **Functionality**: Extracts a specified range of characters by defining a starting point and the number of characters.
   - **Example**: From "ABC12345," extracting a range from position 3 with a length of 4 results in "C123."
   - **Use Case**: Ideal for pulling substrings from data with a known structure.
4. **Extract Text Before/After/Between Delimiters**:
   - **Text Before Delimiter**:
     - **Functionality**: Extracts text up to a specified delimiter.
     - **Example**: From "John.Doe@example.com," extracting text before the "." results in "John."
   - **Text After Delimiter**:
     - **Functionality**: Extracts text after a specified delimiter.
     - **Example**: From "John.Doe@example.com," extracting text after "@" results in "example.com."
   - **Text Between Delimiters**:
     - **Functionality**: Extracts text between two specified delimiters.

- **Example**: From "John.Doe@example.com," extracting text between "." and "@" results in "Doe."

Use Case: Extracting parts of email addresses, URLs, or product codes, often used for refining data and simplifying analysis.

Example Using All Three (Split, Format, Extract)

Suppose you have a dataset with a column, **Full Name**, containing values like "Mr. John Doe."

Steps:

1. **Split Column** by space to separate "Mr.", "John," and "Doe."
   - Result: Three columns—**Prefix**, **First Name**, and **Last Name**.
2. Use **Format**:
   - **Capitalize Each Word** for **First Name** and **Last Name** columns if names are inconsistently capitalized.
3. Use **Extract**:
   - Extract only the initials from **First Name** and **Last Name** to create an **Initials** column.
   - **Extract First Characters** with n=1 for each name column and then concatenate to get initials like "JD."

Summary Table

Transformation	Description	Example
Split Column	Divides data in a column into multiple columns based on delimiters or positions	Split "John Doe" into "John" and "Doe"
Format	Adjusts text formatting for consistency	Convert "john doe" to "John Doe"
Extract	Pulls specific parts of text based on position or delimiters	Extract domain from "example.com" in "name@example.com"

Understanding **Split Column**, **Format**, and **Extract** empowers you to transform data flexibly, making it suitable for various analytical tasks in Power Query.

In [ ]: