**TODO:**

1. Description of Data Cleaning Performed (MenuItem)
2. Document data quality changes
   1. Quantify the results of your efforts, e.g., by providing a summary table of changes: Which columns changed? How many cells (per column) have changed, etc.? (10 points)
   2. Demonstrate that data quality has been improved, e.g., by devising IC-violation reports (answers to denial constraints) and showing the difference between the number of IC violations reported before and after cleaning. (10 points)
3. Create a workflow model
   1. A visual representation of your overall (or “outer”) workflow W1, e.g., using a tool such as YesWorkflow. At a minimum, you should identify key inputs, outputs, and steps of the workflow, along with dependencies between these. Key phases and steps of your data cleaning project may include, e.g., data profiling, data loading, data cleaning, IC violation checks, etc. Explain the design of W1 and why you’ve chosen the tools that you have in your overall workflow. (10 points) A detailed (possibly visual) representation of your “inner” data cleaning workflow W2 (e.g., if you’ve used OpenRefine, you can use the OR2YW tool). (10 points)
4. Submission of supplementary materials in a single ZIP file (10 points)
   1. Workflow Model

• A single PDF file with your Phase-II report. This report should include:

* A narrative that ties all steps together and explains the motivation (use case U1), the rationale for the design of the overall workflow W, and the tools used. – Documentation that data quality was improved, e.g., through running “before queries” QU (D) and “after queries” QU (D′ ) on D (original) and D′ (cleaned), respectively.
* A summary of the data changes ∆D resulting from the overall workflow W: D ⇝ D′ .
* A summary of findings, problems encountered, and lessons learned, including possible next steps (e.g., how would you implement the main use case U1).

**Description of Data Cleaning Performed**

**Menu:**

### **Description of Data Cleaning Performed**

The data cleaning workflow consists of a series of transformations applied to multiple columns. These can be grouped into four main high-level steps:

1. **Trimming Whitespace:** The value.trim() operation was applied to 19 different columns to remove any leading or trailing whitespace.
2. **Standardizing Text:** The workflow standardized text in several columns.
   * **Replacing Whitespace:** The grel:value.replace(/\s+/, '\_') operation was used to replace all whitespace with underscores.
   * **Removing Brackets and Punctuation:** The grel:value.replace(/[\[\]]/, '') and grel:value.replace(/[\[\]";?()\*!]/, '\_') operations were used to remove specific characters.
3. **Standardizing Case:** The value.toTitlecase() operation was applied to ensure consistent capitalization.
4. **Type and Format Conversion:**
   * **Converting to Number:** The value.toNumber() operation was applied to convert their data type to a number.
   * **Converting and Formatting Date:** The value.toDate() and grel:toString(toDate(value), "yyyy-MM-dd") operations were used to convert text to a standard date format.

### **Rationale for each high-level data cleaning step**

1. **Trimming Whitespace:** The presence of leading or trailing spaces can cause different entries to be treated as unique, even if the core content is identical. This could lead to inaccurate counts, failed joins, and inconsistent filtering. This step is a prerequisite for any use case that relies on consistent data values for accurate analysis or lookups.
2. **Standardizing Text:** These transformations were performed to make text data more uniform and easier to work with. This is particularly useful for use cases that involve searching, tagging, or grouping records based on these text fields.
3. **Standardizing Case:** Using toTitlecase() ensures that the values in columns like name, sponsor, and location are presented consistently. For example, "new york" and "New York" would be treated as separate entities without this step.
4. **Type and Format Conversion:**
   * **Number Conversion:** The conversion of page\_count, dish\_count, and id to a numeric type is essential. Without it, these values would be treated as text, making it impossible to perform mathematical operations, such as summing counts or calculating averages.
   * **Date Formatting:** Converting the date column to a standardized yyyy-MM-dd format ensures that all date entries are interpreted correctly. Inconsistent date formats can lead to significant errors in chronological sorting, time-series analysis, and filtering by date ranges.

**MenuItem:**

**MenuPage:**

**Description of Data Cleaning Performed**:

1. **Text Transformation (Trimming Whitespace):** The value.trim() operation was applied. This step removed any leading or trailing whitespace from the cell values in these columns.
2. **Text Transformation (Type Conversion):** The value.toNumber() operation was applied. This step converted the data in these columns from text strings to numerical data types.

### **Rationale for each high-level data cleaning step:**

1. **Trimming Whitespace:** The presence of leading or trailing spaces can cause different entries to be treated as unique, even if the core content is identical. This could lead to inaccurate counts, failed joins, and inconsistent filtering. This step is a prerequisite for any use case that relies on consistent data values for accurate analysis or lookups.
2. **Type Conversion:** The conversion to a numeric type is essential. Without it, these values would be treated as text, making it impossible to perform mathematical operations, such as summing counts or calculating averages.

**Dish:**

Due to the size of the Dish Table, it was difficult to use OpenRefine; therefore, we opted to use Python for data cleaning.

### **Description of Data Cleaning Performed**

1. **Normalization of Dish Names:** This process involves a series of string transformations on the name column, including converting to lowercase, removing specific punctuation and non-alphanumeric characters, and standardizing whitespace.
2. **Filtering of Empty/Null Values:** Rows with empty, null, or 'nan' values in the name column are removed from the dataset.
3. **Clustering**: The name column had K-Means clustering applied, clustering like named dishes.
4. **Removal of a Column:** The description column is explicitly dropped from the final DataFrame.

### **Rationale for each high-level data cleaning step**

1. **Normalization of Dish Names:**
   1. **Trimming Whitespace:** The presence of leading or trailing spaces can cause different entries to be treated as unique, even if the core content is identical. This could lead to inaccurate counts, failed joins, and inconsistent filtering. This step is a prerequisite for any use case that relies on consistent data values for accurate analysis or lookups.
   2. **Standardizing Text:** These transformations were performed to make text data more uniform and easier to work with. This is particularly useful for use cases that involve searching, tagging, or grouping records based on these text fields.
   3. **Standardizing Case:** Converting to lowercase ensures that the values are presented consistently. For example, "new york" and "New York" would be treated as separate entities without this step.
2. **Filtering of Empty/Null Values:** This step is a fundamental aspect of data integrity. Rows with empty or null values in the name are unusable for the analysis of dish data. Retaining these rows could lead to errors in calculations, inaccurate cluster assignments, or failed processing in subsequent steps. Therefore, this step is required to ensure that the dataset is of high quality and that downstream operations can execute without errors.
3. **Clustering:** This step is done to group like-named dishes under a single name. This is done so that analysis on a specific dish can be done, regardless of how it was represented in the original dataset.
4. **Removal of a Column:** The description column is removed because all rows are empty and to simplify the dataset

**Document Data Quality Changes**

**Menu:**

**MenuItem:**

**MenuPage:**

**Dish:**

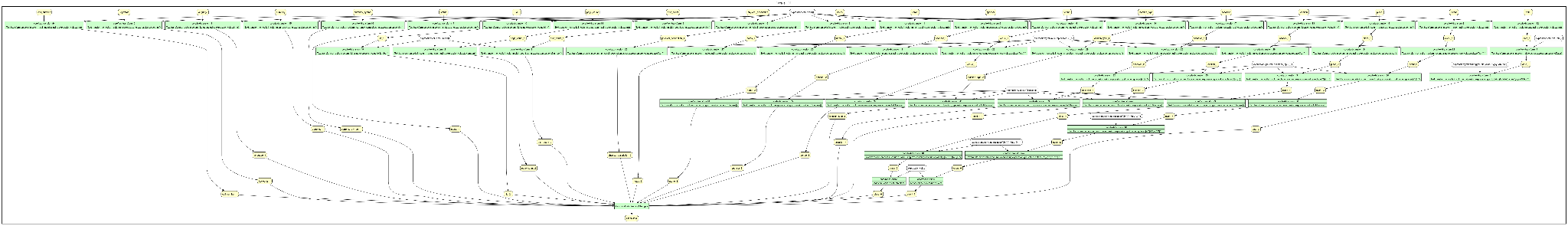
1. Empty Description Column was Removed
2. Names were clustered to match similar dishes
3. Document data quality changes
   1. Quantify the results of your efforts, e.g., by providing a summary table of changes: Which columns changed? How many cells (per column) have changed, etc.? (10 points)
   2. Demonstrate that data quality has been improved, e.g., by devising IC-violation reports (answers to denial constraints) and showing the difference between the number of IC violations reported before and after cleaning. (10 points)

**Workflow Models**

A visual representation of your overall (or “outer”) workflow W1, e.g., using a tool such as YesWorkflow. At a minimum, you should identify key inputs, outputs, and steps of the workflow, along with dependencies between these. Key phases and steps of your data cleaning project may include, e.g., data profiling, data loading, data cleaning, IC violation checks, etc. Explain the design of W1 and why you’ve chosen the tools that you have in your overall workflow. (10 points)

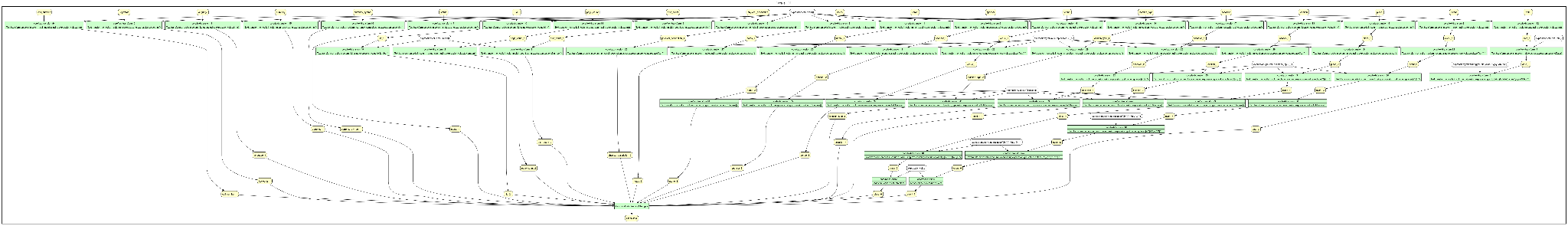
A detailed (possibly visual) representation of your “inner” data cleaning workflow W2 (e.g., if you’ve used OpenRefine, you can use the OR2YW tool). (10 points)

**Menu:**

****

**MenuItem:**

**MenuPage:**



**Dish:**

**Conclusions & Summary**

Please provide a concise summary and conclusions of your project, including lessons learned. Reflect on how work was completed. You should summarize the contributions of each team member here (for teams with >= 2 members).