Data Validation Checklist: January 2021 to December 2022

# January 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# February 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# March 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# April 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# May 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# June 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# July 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# August 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# September 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# October 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# November 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# December 2021 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# January 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# February 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# March 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# April 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# May 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# June 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# July 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# August 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# September 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# October 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# November 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**

# December 2022 Data Validation Checklist

## 1. File-Level Validation:

**[ ] 1. Confirm file existence and accessibility.**

**[ ] 2. Validate file format.**

**[ ] 3. Check file completeness for the month.**

## 2. Header Validation:

**[ ] 1. Validate column names and order.**

**[ ] 2. Rename `member\_casual` to `user\_type`.**

## 3. Data Type and Format Validation:

**[ ] 1. Ensure data type consistency for each column.**

**[ ] 2. Validate and convert date formats in started\_at and ended\_at.**

**[ ] 3. Validate format of start\_station\_id and end\_station\_id.**

## 4. Content Validation:

**[ ] 1. Validate values in rideable\_type.**

**[ ] 2. Ensure user\_type contains only "member" or "casual".**

**[ ] 3. Check correspondence between station names and IDs.**

**[ ] 4. Validate uniqueness of ride\_id.**

**[ ] 5. Ensure valid geographical coordinates.**

## 5. Consistency and Logical Checks:

**[ ] 1. Verify logical consistency in ride durations.**

**[ ] 2. Check spatial consistency between start and end locations.**

## 6. Missing Value Check:

**[ ] 1. Identify and handle missing values.**

## 7. Outlier Detection:

**[ ] 1. Detect and handle outliers in ride durations and spatial data.**

## 8. Station ID Range:

**[ ] 1. Validate start\_station\_id and end\_station\_id against sensible ranges or a master list.**

## 9. Temporal Consistency:

**[ ] 1. Identify suspicious temporal patterns.**

**[ ] 2. Ensure no future dates are present in the data.**

## 10. Documentation:

**[ ] 1. Log all steps, inconsistencies, and actions taken during validation.**