IMDb Data Cleaning Report

MINAL NARANJE – 002087516 | **ADITI DESHMUKH –** 002055568 | **SAMEER MALVE –** 002316478

1. title.principals

1. Data Overview

The title.principals dataset contains key information about the main cast and crew members associated with various movie and TV show titles. It helps establish relationships between people (actors, directors, writers, etc.) and specific titles. This dataset is crucial for analyzing the contribution of individuals in the entertainment industry.

- **Total Rows Before Cleaning**: 90,984,102
- **Total Columns**: 6
- **Key Columns**:
    - tconst: Unique identifier for a title.
    - ordering: Position of the person in the credits.
    - nconst: Unique identifier for a person.
    - category: Role of the person (e.g., actor, director, producer).
    - job: Specific job title (if applicable).
    - characters: Character name played (if applicable).

2. Handling Missing Values

During the data profiling stage, we identified missing values in the following columns:

- **job**: Many records had NULL values, particularly for actors and actresses.
- **characters**: Missing in cases where the role is behind the scenes (e.g., directors, producers).

Cleaning Actions:

- For **job**, we replaced missing values with "Unknown" for better consistency.
- For **characters**, missing values were replaced with "N/A" since not all roles have a character name.

3. Handling Duplicates

- Checked for duplicate records based on (tconst, nconst, category, ordering) to ensure that the same person was not listed multiple times for the same title in an identical role.

- No exact duplicates were found.

4. Standardizing Data

To ensure uniformity:

- **Converted category values** to lowercase to maintain consistency (actor → Actor).
- **Trimmed leading/trailing whitespaces** from job and characters fields to prevent inconsistencies.
- **Ensured ordering is stored as an integer**, avoiding incorrect data types.

5. Format & Integrity Checks

- Ensured that tconst and nconst are correctly formatted as alphanumeric values.
- ordering was checked to confirm it contained only numeric values.
- Ensured that category contains only expected values (e.g., actor, director, producer), eliminating any erroneous values.

6. Referential Integrity Checks

- Verified that all tconst values exist in the **title.basics** dataset.
- Verified that all nconst values exist in the **name.basics** dataset.
- Any orphaned records (where tconst or nconst was missing from the respective dataset) were **removed** to maintain integrity.

7. Data Transformations

- **Created a surrogate key**: Introduced Title_Principal_ID as a unique identifier for each row in the transformed table.
- **Reformatted character names**: If the characters column contained lists (["John Doe"]), extracted and stored only the primary name.

8. Final Cleaned Dataset

After the cleaning process:

- **Final Row Count**: Slightly reduced after removing orphaned records.
- **Null values in critical columns**: Eliminated.
- **Data consistency & accuracy**: Improved through standardization.

---

2. title.akas

1. Data Overview

The title.akas dataset contains alternative titles for movies and TV shows in different languages and regions. This dataset helps in identifying how titles are localized across different countries.

- **Total Rows Before Cleaning**: 51,409,880
- **Total Columns**: 8
- **Key Columns**:
    - titleId: Unique identifier for the title (matches tconst from title.basics).
    - ordering: Position of the alternative title.
    - title: Alternative or localized title.
    - region: The country/region where this title is used.
    - language: The language of the title.
    - types: Type of alternative title (e.g., "DVD", "festival", "TV").
    - attributes: Additional descriptors (e.g., "uncut version").
    - isOriginalTitle: Boolean flag indicating if this is the original title.

2. Handling Missing Values

During profiling, we found that some columns contained missing values:

- **region**: Some records lacked a defined region.
- **language**: Some titles did not have a specified language.
- **types** and **attributes**: Many records had missing values.

Cleaning Actions:

- **Replaced missing values in region and language with "Unknown"** to ensure consistency.
- **Filled types and attributes with "N/A"** where missing to maintain uniformity.
- **For isOriginalTitle**, missing values were assumed to be "0" (not original title).

3. Handling Duplicates

- Identified duplicate records based on (titleId, ordering, title, region, language).
- Removed **exact duplicate rows** to prevent redundancy.

4. Standardizing Data

- **Converted all text columns to lowercase** to ensure uniformity across records.
- **Trimmed spaces** from title, region, language, types, and attributes fields.
- **Ensured ordering was stored as an integer** for accurate sequencing.
- **Converted isOriginalTitle to Boolean (0 or 1)** to maintain consistency.

5. Format & Integrity Checks

- Ensured **titleId** matched values in the title.basics dataset.
- Standardized **region codes** to match the **ISO 3166-1 standard** (Country Code for Movies).
- Standardized **language codes** to match **ISO 639-1 standards**.
- Any orphaned records (where titleId did not exist in title.basics) were **removed** to maintain referential integrity.

6. Data Transformations

- **Created a surrogate key**: Introduced Title_Akas_ID as a unique identifier for each row.
- **Standardized title column**: Removed special characters and ensured proper UTF-8 encoding.

7. Final Cleaned Dataset

After applying all cleaning steps:

- **Final Row Count**: Slightly reduced due to duplicate removal and orphaned records.
- **Null values in critical columns**: Eliminated.
- **Data integrity and consistency**: Improved through standardization.

---

3. name.basics

1. Introduction

The name.basics file contains information about individuals in the film industry, including their unique identifier (nconst), primary name, birth and death years, primary professions, and titles they are known for. The data cleaning process ensures consistency, accuracy, and completeness for further analysis.

2. Issues Identified

During data profiling, we identified the following issues in the dataset:

- Missing values in birthYear, deathYear, primaryName, primaryProfession, and knownForTitles.

- birthYear and deathYear contained values marked as \N, indicating missing data.
- Inconsistent data types across numeric fields (birthYear and deathYear were stored as strings).
- Erroneous values in birthYear and deathYear (e.g., values outside a reasonable range).
- Some records contained primary names marked as null or empty.
- primaryProfession and knownForTitles had inconsistent and missing values.

3. Cleaning Steps

The cleaning process in **Alteryx** involved multiple steps to address these issues:

Step 1: Data Input

- The dataset was loaded into Alteryx from name_basics_clean.csv, containing 14,230,077 records.
- The file delimiter was set as \t (tab-separated values).

Step 2: Handling Missing Values

- birthYear: Replaced \N with 9999 to indicate unknown birth years.
- deathYear: Replaced \N with 9999 to signify unknown or still-living individuals.
- primaryName: Replaced missing values with Unknown to avoid blank entries.
- primaryProfession: Replaced missing values with Unknown for consistency.
- knownForTitles: Replaced missing values with Missing to indicate unavailable records.

Step 3: Data Type Corrections

- Converted birthYear and deathYear from string to numeric format.
- Ensured birthYear values were within the range **1500 to 2025**; outliers were replaced with 9999.
- Ensured deathYear values followed a similar constraint.

Step 4: Error Handling and Validation

- Applied the **Auto Field Tool** to automatically assign correct data types.
- Checked for inconsistencies in primaryProfession and knownForTitles.
- Reviewed error messages, such as missing fields in the dataset, and resolved them where possible.

Step 5: Data Output

- After cleaning, the refined dataset was exported as name_basics_clean.tsv.

- The output maintained the original structure but ensured standardized and accurate values.

---

4. title.basics File

1. Overview of the Cleaning Process

The **title.basics** dataset contains essential information about movies, TV shows, and other audiovisual content. It includes fields such as **tconst (unique identifier), primaryTitle, originalTitle, titleType, isAdult, startYear, endYear, runtimeMinutes,** and **genres**. The data cleaning process focused on ensuring consistency, handling missing values, and standardizing formats to prepare the dataset for analysis and integration into Snowflake.

2. Handling Missing and Null Values

- **startYear and endYear:** Missing values for **startYear** were identified and replaced with a default placeholder value of **9999** to indicate unknown years.
- **runtimeMinutes:** Null or missing runtime values were replaced with **"-1"** to differentiate between unavailable and zero-duration entries.
- **genres:** Entries with missing genre values were updated to **"Unknown"** to ensure no empty values remained.
- **isAdult:** The dataset contained missing values in this column, which were converted to numerical format using **0 for non-adult content** and **1 for adult content**.

3. Standardizing Data Types

To maintain data integrity and optimize performance, appropriate data types were enforced:

- **tconst (Unique ID):** Converted to a **string (length 10)** to retain IMDb's standard identifier format.
- **titleType:** Stored as a **string** since it contains categorical values like *movie, short, tvSeries, etc.*
- **primaryTitle & originalTitle:** Standardized as **variable-length string (V_String) with a length of 500** to accommodate longer titles.
- **startYear and endYear:** Converted to **integer (Int16)** to ensure proper numerical sorting and calculations.
- **runtimeMinutes:** Maintained as a **variable-length string (V_String)** to avoid inconsistencies in numerical storage.
- **genres:** Converted to **variable-length string (V_String, length 32)** to support multiple genre categories.

4. Removing Inconsistencies

- Checked for **invalid characters** and non-numeric values in **startYear, endYear,** and **runtimeMinutes**.
- Applied **Auto Field** transformation to automatically assign optimal field types based on data values.

5. Final Output

- The cleaned dataset was exported as **Title_Basics_Clean.csv**.
- The dataset is now structured, consistent, and ready for further transformation and analysis.

---

5. title.crew

1. Overview of the Dataset

The **title.crew** dataset provides information about the directors and writers associated with movies and TV shows. It consists of three key columns:

- **tconst**: Unique identifier for each title (primary key).
- **directors**: A list of director IDs associated with a title.
- **writers**: A list of writer IDs associated with a title.

This dataset is crucial for understanding the creative contributors behind IMDb-listed titles and is used in conjunction with other IMDb datasets.

2. Identified Data Issues

After performing data profiling, we identified the following key issues that required cleaning:

- **Missing Values**: The dataset contains missing values, represented as \N, in both the directors and writers columns.
- **Data Format Inconsistencies**: The directors and writers columns contain multiple IDs separated by commas, requiring validation.
- **Column Data Types**: Initially, directors and writers were read as generic string fields with an inconsistent structure.

3. Data Cleaning Steps

To address these issues, the following steps were applied using **Alteryx**:

Step 1: Input Data Configuration

- The **title.crew.tsv** file was loaded using a tab-separated format (\t) to preserve column structure.
- Preview of the first 100 records confirmed that \N was used to represent missing values.

Step 2: Handling Missing Values

- The \N values in directors and writers columns were replaced with **UNKNOWN** to maintain consistency and avoid null values.
- This replacement ensures that missing values are properly categorized instead of being left blank.

Step 3: Standardizing Data Types

- The tconst column was kept as a **string** with a length of **10 characters** to ensure compatibility across datasets.
- The directors and writers columns were converted to **variable-length string (V_WString)** with a size of **15,000 characters** to accommodate multiple IDs separated by commas.
- This step ensures that all values conform to the expected format and eliminates inconsistencies in data interpretation.

Step 4: Field Selection and Verification

- The cleaned dataset was previewed to confirm that UNKNOWN was correctly assigned to missing values.
- Data distribution analysis was performed, highlighting that a significant portion of records (~3.6M) have unknown directors, and ~4.1M records have unknown writers.
- The final schema was reviewed to ensure compatibility with downstream processes.

Step 5: Exporting the Cleaned Data

- The cleaned dataset was exported as **Title_Crew_Clean.csv** to be used in further transformations and integrations.

---

6. title.episode

1. Overview

The **title.episode** file contains information about TV show episodes and their associated parent titles. This dataset includes details such as the episode title identifier, its corresponding parent title, and the season and episode numbers.

2. Data Issues Identified

After profiling the dataset, we identified several key data quality issues:

- **Missing Values:** The columns seasonNumber and episodeNumber contained missing values represented as \N.
- **Inconsistent Data Types:** The seasonNumber and episodeNumber columns were stored as string types, which are not suitable for numerical operations.
- **Outliers:** There were some extremely high values in seasonNumber and episodeNumber, indicating potential data errors.

3. Data Cleaning Steps

To address the above issues, we performed the following cleaning steps in **Alteryx**:

Step 1: Handling Missing Values

- Replaced \N values in seasonNumber and episodeNumber with -1 to indicate missing data while preserving numerical consistency.

Step 2: Correcting Data Types

- Converted seasonNumber and episodeNumber from string type to numerical type using the **ToNumber** function.

Step 3: Handling Outliers

- Implemented a condition where:
    - If seasonNumber was greater than 50, it was set to 50 (assuming a logical upper limit for TV seasons).
    - If episodeNumber was greater than 500, it was set to 500 (assuming an upper threshold for total episodes per series).

Step 4: Field Type Standardization

- Applied **Auto Field** to optimize data types across all columns.
- Ensured tconst and parentTconst remained as string identifiers while seasonNumber and episodeNumber were converted into integer types.

Step 5: Final Review and Export

- Verified the transformed dataset by reviewing the metadata and value distributions.
- Exported the cleaned data as Title_Episode_Clean.csv for further use.

4. Summary of Changes

Column Name	Action Taken
tconst	Retained as string identifier

parentTconst	Retained as string identifier
seasonNumber	Converted from string to integer, missing values replaced with -1, and capped at 50
episodeNumber	Converted from string to integer, missing values replaced with -1, and capped at 500

---

7. Language Codes Dataset

1. Overview of the Dataset

The **Language Codes** dataset provides a mapping between language names and their corresponding ISO codes. This dataset is essential for standardizing language representation across various applications. The dataset consists of two primary fields:

- **ISO Language Names**: The official language name.
- **Language Codes**: The corresponding standardized language codes, which may include multiple codes separated by commas.

2. Identified Data Quality Issues

Based on our initial data profiling in Alteryx, we identified the following issues:

- **Inconsistent Data Types**: The language codes were stored in a non-uniform format.
- **Potential Formatting Issues**: Variations in case sensitivity and spacing across records.
- **Missing or Unknown Values**: Some language codes were incomplete or had placeholder values.

3. Data Cleaning Steps

To ensure data consistency and usability, we performed the following cleaning steps in Alteryx:

3.1. Input Data Configuration

- We loaded the raw dataset from **Language_Codes_Clean.csv** using the **Input Data Tool** in Alteryx.
- The dataset was separated by commas, ensuring proper parsing of fields.

3.2. Standardizing Data Types

- We applied the **Auto Field Tool** to automatically detect and assign the optimal data types for each column:
    - **ISO Language Names** was converted to **V_WString (variable-length string)** to accommodate longer names.

- **Language Codes** was assigned the **String** type to maintain uniformity in text representation.

3.3. Removing Formatting Issues

- We checked for leading or trailing spaces in the **ISO Language Names** and **Language Codes** fields.
- Applied standard text formatting to ensure uniformity across all entries.

3.4. Handling Missing or Unknown Values

- We identified instances where language codes were missing or contained placeholder values such as **"\N"**.
- For such records, we replaced **"\N"** with **"Unknown"** to make it more interpretable.

3.5. Verifying Uniqueness and Data Integrity

- We confirmed that each **ISO Language Name** had a corresponding **Language Code**.
- Ensured that no duplicate records existed by checking for unique combinations of language names and codes.

4. Summary of Improvements

Issue Identified	**Cleaning Approach Implemented**
Inconsistent data types	Used **Auto Field Tool** to standardize column types
Formatting inconsistencies	Trimmed spaces and standardized text formatting
Missing or unknown values	Replaced **"\N"** values with **"Unknown"**
Potential duplicates	Verified unique language name and code pairs

---

8. Region Code Cleaning Documentation

Objective:

The purpose of this workflow is to clean and standardize the region codes dataset by ensuring proper formatting, handling missing values, and maintaining data consistency for further processing.

Source File:

- Input Data: Region_Codes_Clean.csv
- File Format: Comma-Separated Values (CSV)
- Fields:

- Country_SK (Surrogate Key)
- Country Code (ISO 3166-1 Alpha-2 Code)
- Country Name (Full Country Name)

Data Cleaning Steps:

1. Data Input

- The dataset is imported from Region_Codes_Clean.csv.
- The delimiter is set as a comma (,).

2. Data Cleansing

- **Handling Null Values:**
  - Null values in string fields (Country Code, Country Name) are replaced with blanks.
  - Null values in numeric fields (Country_SK) are replaced with 0.
- **Removing Unwanted Characters:**
  - Leading and trailing whitespaces are removed.
  - Tabs, line breaks, and duplicate whitespaces are cleaned.

3. Uppercasing Country Code

- A Formula tool is applied to ensure all Country Code values are in uppercase.
- Expression used: Uppercase([Country Code]).

4. Handling Duplicate Entries

- The Unique tool is used to remove any duplicate records based on Country Code.

5. Assigning a New Unique Identifier

- A Record ID tool is used to generate a unique identifier for each record.
- Country_SK2 column is added to store this new unique identifier.

6. Auto Field Conversion

- The Auto Field tool is applied to dynamically assign optimal data types to fields.
  - Country Code converted to String (4).
  - Country Name converted to V_String (46).
  - Country_SK converted to Byte.

7. Final Output

- The cleaned dataset is stored back into Region_Codes_Clean.csv for further usage.
- The final dataset contains **249 unique records**.

Summary of Cleaning Process:

Step	Action
Data Import	Loaded Region_Codes_Clean.csv
Data Cleansing	Removed null values, cleaned unwanted characters
Uppercasing	Converted Country Code values to uppercase
Duplicate Removal	Used Unique tool to filter out duplicates
Assigning Unique ID	Generated new Country_SK2 identifier
Auto Field Conversion	Optimized field data types
Export	Saved cleaned data to Region_Codes_Clean.csv

---

9. Title Ratings

Objective

The purpose of this workflow is to clean and standardize the **title ratings dataset**, ensuring that rating values are properly formatted and stored for further analysis.

Input Data

- **File Name:** title.ratings.tsv
- **Fields:**
    - tconst (Unique Identifier for Titles)
    - averageRating (IMDb Rating)
    - numVotes (Number of Votes Received)

Workflow Steps

1. Input Data

- The dataset is loaded using the **Input Data Tool** from title.ratings.tsv.

- The file format is **tab-separated (.tsv)**.
- The preview shows the dataset consists of tconst, averageRating, and numVotes.

2. Auto Field Conversion

- The **Auto Field Tool** is applied to optimize field types by converting them automatically:
    - tconst → Converted to **String** (to maintain the unique identifier format).
    - averageRating → Converted to **Double** (for floating-point rating values).
    - numVotes → Converted to **Integer** (as it represents count data).

3. Data Profiling and Validation

- The **Browse Tool** is used to inspect the distribution of values:
    - Ensures averageRating falls within the valid range (typically 0-10).
    - Checks numVotes for anomalies (such as negative or unrealistic values).
    - Verifies that tconst values remain unique.

4. Output Data

- The cleaned dataset is exported as Title_Ratings_Clean.csv.
- The dataset now contains **1,544,438 records**, properly formatted and structured.