**Data Analysis and Preprocessing Documentation**

**Project Description:** To analyse side effect data to understand the side effects associated with users' medication use and to correct missing/data errors.

**Objective:** My task is to complete missing data, normalise data, analyse side effects and prepare data for further predictive modelling.

Firstly I converted the file you sent to CVS format. then I connected to my Google Drive.

**Errors**

The column headings of the file are separated by ‘’ ;‘’.

Column names are: User\_id, Gender, Date of Birth, Nationality, Province, Drug\_Name, Drug\_Basic\_Date, Drug\_Basic\_Date, Drug\_Bitis\_Date, Side\_Effect, Side\_Effect\_Notification\_Date, Allergies, Chronic Diseases, Father Chronic Diseases, Mother Chronic Diseases, Daughter Chronic Diseases, Son Chronic Diseases, Blood Group, Weight, Height.

Missing Data: Some records are missing in the columns Province, Father Chronic Diseases, Brother Chronic Diseases, Weight, Height.

Empty Gender Information: There are missing values in the Gender column (e.g. User\_id 61 and 95).

Missing Values: There are empty values in some rows in the Height column. Some rows in the Province column have empty values. I used SimpleImputer and OneHotEncoder to process missing data and LabelEncoder for categorical data.

Inconsistency of Numerical Data: There are blank values between columns such as Weight and Height. This made statistical analyses difficult.

**Pre-Processing Steps:**

Filling Missing Data: For missing categorical data (e.g. Gender, Il), the most common values can be used. For numerical data (e.g. Height, Weight), the mean or median can be used.

Management of Empty Rows: If there is too much missing data in certain columns, these rows can be discarded or filling methods can be used.

Coding Categorical Data: Categorical variables (e.g. Gender, Il)

I created a bar graph to show which gender group experienced which side effects.

With pd.read\_csv I loaded the file with the ; separator.

I numericised categorical columns such as Gender and Il using OneHotEncoder.

I added the coded columns to the original data frame and removed the old categorical columns.

At the end of the script I displayed the first few rows with df.head().

By visualising the distribution of categorical columns (e.g. Gender, Il) in the data, I easily showed which categories were more or less. This helped to understand the overall distribution of your data during data analysis.

In the Gender column, categories such as Male and Female are coded as 0 and 1. Likewise, I coded the Il column as separate columns for each city.

Categorical Columns: I specified categorical columns such as Gender and Il.

Cleaning Missing Data: I cleaned the missing data with dropna() function.

OneHotEncoder: I converted categorical columns to numeric values 0 and 1 with OneHotEncoder.

New Columns: I created a DataFrame by getting new column names with get\_feature\_names\_out().

Merge: I merged the coded columns with the original dataset.

Removing Categorical Columns: Since we no longer need the Gender and Il columns, I removed these columns.

I used df.head() to see the first few lines at the end of the code for Result Display.

df\_cleaned = df.dropna() # Deletes all rows with missing values.