About the dataset

LEGO Database dataset, link here:
https://www.kaggle.com/datasets/rtatman/lego-database?select=colors.csv

Context

LEGO is a popular brand of toy building bricks. They are often sold in sets in order to build a specific object. Each set contains a number of parts in different shapes, sizes and colors. This database contains information on which parts are included in different LEGO sets. It was originally compiled to help people who owned some LEGO sets already figure out what other sets they could build with the pieces they had.

This dataset contains the LEGO Parts/Sets/Colors and Inventories of every official LEGO set in the Rebrickable database. These files are current as of July 2017.

EDA & Data Quality Analysis

This dataset consists of eight different csv files, containing data about inventories, themes, inventory sets, parts, part & categories, colors, sets and inventory parts.

Importing and reading data

Using *python,* all the csv files for the analysis were read and imported to different data frames, and two variables were created to 1) store the list of all data frames and 2) store the names of each dataframe, both lists following the same order. These lists will allow to iterate when needed, for each dataframe.

Print & check data

Using the *head* function, it was possible to check all the first rows and columns titles in each dataframe and understand the kind of data stored in each file. Also, this allowed to understand that all of the dataframes had *id* columns, which may be the primary keys that will serve as the unique identifiers for each row in each of the tables. A new variable was created, with a dictionary of all the pairs dataframe : title of primary key column.

Also, for dataframe *colors*, it was noticed that a column name could be updated for a better understanding - from *is_trans* to *is_transparent*. Along with this, on dataframes *colors* and *inevntory_parts*, were identified two columns whose values are *f* and *t* that could also be updated to *false* and *true*, for a better comprehension.

Duplicates validation

A duplicate validation was done for each of the primary key columns identified, and returned only the cases where the column had duplicate values.

Two dataframes/columns were returned (*inventory sets* and *inventory parts*), and all the other *id* columns in this dataset were subject to duplicated values analysis, but none of the columns has unique values. It was decided to still keep the dataframes to keep any possible relevant information, but having in mind that these need extra attention when considering their data for possible joins.

Missing values

A function to check for missing values in all columns in all dataframes was implemented, and returned, for all dataframes, 1) the number of columns, 2) the complete list of columns in the dataframe along with the amount of values missing in that column.

Only one dataframe had values missing - for table *Themes,* column *parent_id*, were missing 111 values. For this dataset specifically, it was analyzed the total of rows and the total of missing values to understand if the missing values represented a big amount of the total. In this case, missing values are 18.1% of the total, which is a considerable amount. It was decided to still keep the column, so any important information is kept, with the note that this column contains missing information on parent ids.

Data revision

As previously identified, for *colors* dataframe, the column name *is_trans* was updated to *is_transparent*, and for this dataframe and for *inventory parts*, columns whose values were *f* and *t* were updated to *false* and *true*, respectively.

Importing to MySQL

A connection with *MySQL* was established and a new database *lego_database* was created. The connection was closed so it was established again but directly to the new database created, and all the dataframes were imported as distinct tables into *lego_database*.

Business Questions to Explore:

1. Which is the theme and year of the most popular lego sets (by the number of parts included)?
2. Which LEGO themes have the highest average number of parts per set?
3. What is the distribution of LEGO sets across the first and last 5 years?
4. What is the difference between lego sets launched between each year and the previous year?
5. How many unique LEGO parts are there in the dataset, per category?

6. Which are the sets that have the parts of the least frequent category?
7. Are there any LEGO parts that are used as spares more frequently than others?
8. Which are the 10 most common colors for lego parts?
9. Which are the sets that have the 5 least used colors for lego parts?
10. How does the distribution of transparent and non-transparent colors vary across LEGO parts?