**Business area of the project**

I have chosen Lego as an area for my project.

LEGO is a popular brand of toy building bricks. They are often sold in sets with in order to build a specific object. Each set contains a number of parts in different shapes, sizes and colors.

LEGO, formed from the Danish words “**LE**g **GO**dt” (“play well”). Later, it is realised that in Latin the word means “I put together”.

It has come a long way over the past almost 85 years - from a small carpenter’s workshop to a modern, global enterprise that is now one of the world’s largest manufacturers of toys. The brick in its present form was launched in 1958. The interlocking principle with its tubes makes it unique and offers unlimited building possibilities. Over the years of its development LEGO firm has created so many sets that it is now reasonable to analyze data related to LEGO using SQL language.

**Initial dataset and problems of loading the data**

I have found the initial data set on <https://rebrickable.com/downloads/> site. The data is regularly updated and my data is of April 27, 2020.

Data cleaning

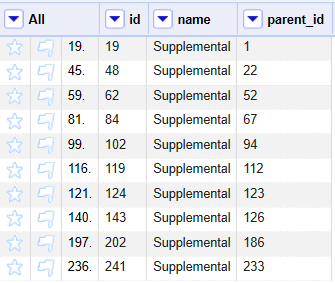
I have used OpenRefine as a data cleaning tool. Below I provide brief overview of some tables using OpenRefine

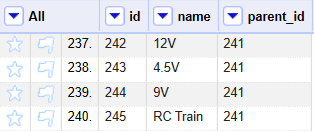
1. Themes cleaning.

Lets’s look at items which have duplicated names:

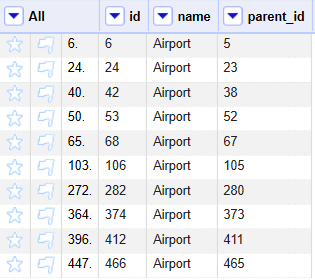
1. Supplemental

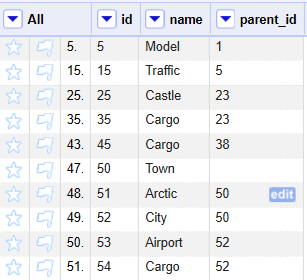
Some rows have duplicating names, but they refer to supplemental ‘subthemes’. We are excluding all parent\_id’s which refer to supplemental themes.





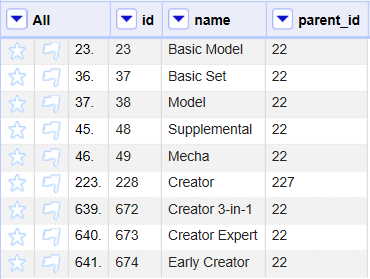
1. Airport

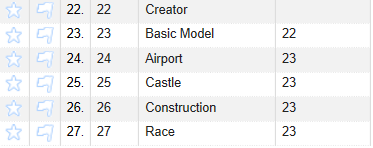




Lego theme Town has quite a long development history. It was initially named ‘Town’ and had several sum-themes, later it was called ‘City’ and also had several subthemes. For the project purposes let’s follow the classification, provided by the dataset. Further development of dataset may stipulate ultimate theme ids.

Creator theme is futher broke into basic models





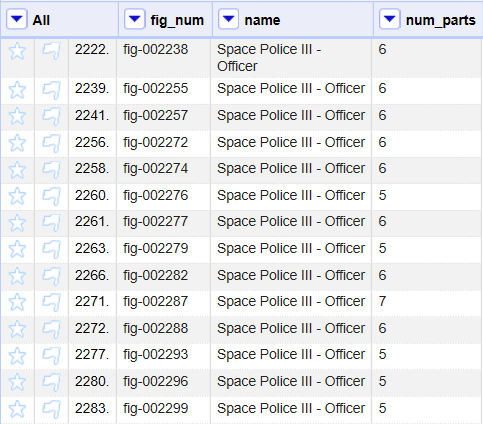
Similarly I have looked through other duplicating items and checked for their plausibility.

1. Colors cleaning

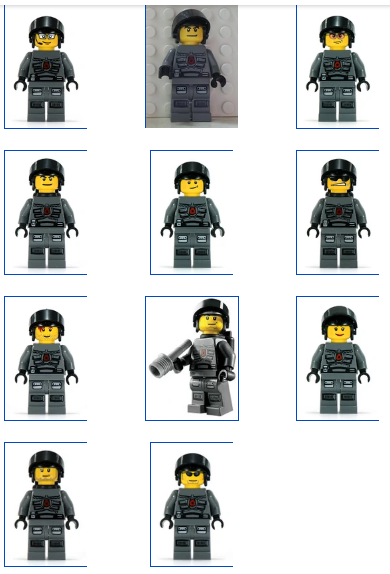
Colors table is clean and ready for use. HEX is more accurate name of the column instead of RGB. Relevant code for renaming of the column was added to SQL\_DDL file.

1. Minifigures

I have checked duplicating names and searched images for them in internet.



These are different versions of figures



To insert the data in the database I have also used OpenRefine. While inserting data I have found out that some column type was not relevant. Relevant code for changing the type of the column was added to SQL\_DDL file.

OpenRefine does not allow to insert data in BOOLEAN type, so I changed column type to text, inserted data in it and casted the column type to BOOLEAN.

Also, I could not insert the whole inventory\_parts table at a time, so I had to break it into several separate files.

**Description of the dataset**

The dataset is a summary of information on a kind of fun site. The data was collected during a long period of time and contains rather accurate and detailed information which was to some extent supported with information on official LEGO site. However the dataset is not officially checked or approved by LEGO company.

The dataset consists of the following tables:

1. Themes – The table summarizes information on the main themes of LEGO sets. The table gives a very detailed breakdown of LEGO themes, e.g. LEGO City is further broken down to Airport, Cargo, etc.
2. Colors – Gives an information on the main colors of LEGO parts through description by words and HEX code
3. Part\_categories – breaks all parts into categories e.g. bricks, wheels, etc.
4. Parts – the table is list of all parts, which are used in LEGO sets.
5. Lego\_sets table contains number of parts in the set. Zero number of parts refers to promotional packs which usually consist of two sets.
6. Minifigs –the table is a list of all minifigures in LEGO sets and minifigures sold separately. It consists of figure id (fig\_num), word description and number of parts.
7. Inventories are the smaller bags of parts that come with LEGO sets like this



Inventories can contain lego\_sets and/or Parts and/or Minifigs. Each set has a number of inventories within it, and each inventory has a given assortment of parts in it. But each inventory might also be used with multiple sets, which I think is why the set\_num (the key to the specific set) is not unique. The different versions reflect whether there were changes to the parts in a given inventory when it was republished, like a different number of bricks or if some bricks were a different color.

To see what parts are in each inventory, you can see what rows in the "inventory\_parts" table reference a specific inventory ID. To see what inventories are in a specific set, you can check the set\_id in the inventory\_sets table.