Modern Data Management

Data Warehouse Fundamentals

&

Business Intelligence

Academic Year 2022 - 2023
Full Time
Assignment 2

Dimitris Matsanganis, f2822212
Foteini Nefeli Nouskali, f2822213

Modern Data Management & Business Intelligence

Assignment 2

Description of the Case

You are going to use SQL Server Database, SQL Server Analysis Services and Power BI or Tableau for this project. You are going to design and develop a data warehouse, build one or more data cubes on top of it, develop some OLAP reports and visualize your results. You are going to present your project in Teams (10'-15' each group). This should be in the form of a business case. This includes:

- business goals, description of the problem/domain
- description of data sources, where did you find the datasets
- design of the data warehouse, cubes, etc
- import/cleaning/transformation challenges and what did you do
- examples of OLAP queries, reports, etc.
- visualization examples

Try to make it as a story – you are the story teller!

1. Find a dataset in the web that seems attractive and interesting to you. Possible links:
www.kaggle.com
https://github.com/caesar0301/awesome-public-datasets
http://www.kdnuggets.com/datasets/index.html
https://catalog.data.gov/dataset?tags=data-warehouse

or, search google for "datasets for data warehousing / data mining / OLAP / etc."

2. Understand the facts and the dimensions of the application. Define a star/snowflake schema in your database SQLServer. Populate the fact and the dimension tables from the dataset you found - for example by using the import task in your database server. You may have to clean, transform the dataset, manually define dimension tables or insert values.

3. Use SQL Server Analysis Services to define a multi-dimensional model (a cube) over your schema. Play with the reporting capabilities of your tool and show some OLAP reports (drill down/roll up, pivoting, ranking, etc.)

4. Install Power BI and, using your database schema, show OLAP examples and visualize these - or whatever else you consider interesting. Better (and more interesting/interactive/etc.) visualizations mean better grade

The deliverables (aside the presentation) should be a document (.doc or .pdf) describing in detail each of the above steps - with a lot of screenshots:
(a) what kind of application you are targeting, description of the dataset you used, where did you find it, what problems you are trying to solve, what analysis you want to do,
(b) description of the relational design of your fact and dimension tables, import methods, cleaning/transformation procedures in detail,
(c) what cube you have built on top of your schema, dimensions, measures, calculated - if any - measures; description (in English) of OLAP reports and screenshots, and
(d) visualizations of these reports and description of the visualization, how it was produced, etc

Contents

Dataset Selection

After extensive research in Kaggle to find a dataset containing data among interdisciplinary fields that meet the specifications and simultaneously offer interesting content that depicts business needs. The initial specification refers to the dataset observation size that must extend above 300.000 rows, present enough categorical variables that will constitute the dimensions of the database schema, enough quantitative variables to be used as metrics, and finally, present opportunities for unpivot variables, drill up or drill down in case of hierarchical relationships among variables and new measures based on the initial existing metrics. Consequently, a dataset concerning info for mobile applications available on Google App Store was selected as it met all the criteria. The dataset concerns the application of different types that present development trends, but they have not yet been established as very well-known and commercially successful apps.

The dataset which settled for analysis contains the following 19 columns (after the data transformation procedure):

- Id
- App Name
- Publisher
- App Size
- Genre
- Price
- isFree
- Rating
- Total Reviews
- Rating_Perc_5
- Rating_Perc_4
- Rating_Perc_3
- Rating_Perc_2
- Rating_Perc_1
- Total Installations
- Required Android OS
- Content Rating
- In-App Purchases MIN
- In-App Purchases MAX

A short description for the above columns can be found below:

Id : A row indicator used for evaluating purposes during the data cleaning and transformation stage.

App Name : The name of each application in the dataset.

Publisher : The name of the publisher - producer of each application in the dataset.

App Size : The application's required size.

Genre : The application's category - genre (eg. Music & Audio).

Price : The price to purchase the application (if it is not a free application).

isFree : A Boolean vector which defines if the application is free or not.

Rating : The average review's rating for each application.

Total Reviews : The total amount of reviews for each application.

Rating_Perc_5 : The 5-star reviews percentage among all the submitted reviews for each application.

Rating_Perc_4 : The 4-star reviews percentage among all the submitted reviews for each application.

Rating_Perc_3 : The 3-star reviews percentage among all the submitted reviews for each application.

Rating_Perc_2 : The 2-star reviews percentage among all the submitted reviews for each application.

Rating_Perc_1 : The 1-star reviews percentage among all the submitted reviews for each application.

Total Installations : The total amount of installation for each application.

Required Android OS : The required android os version in order the application being installed properly.

Content Rating : The contentment rating category for each application.

In-App Purchases MIN : An indicator that shows the minimum possible billing transactions for each application.

In-App Purchases MAX : An indicator that shows the maximum possible billing transactions for each application.

Business Case

Regarding the Business Case that this study covers, we supposed that we are an independent analysts' team that was hired by an IT company with the ambition to be expanded to the mobile application market and more specifically, to the android app market (Google Play Store) for its first published app. The IT vendor wanted either to buy an existing one or develop an app that belongs in one of the seven following genres, which are considered categories with great impact that present commercial interest from the company's perspective. These categories are presented below:

- Tools
- Education
- Health & Fitness
- Music & Audio
- Finance
- Business
- Sports

One of the primary goals is to assure that these categories are among the mobile app genres that actually belong to the most commercially successful ones and the business investment in one of these would be valid. Sequentially, an analysis of the applications' genres, their popularity, profit potentials from various sources, and analysis among age groups that a mobile app address. For all the aforementioned reasons the analysis contains explanatory reports that present the guidelines for the selection of the investment app genre categories that are proposed as the most efficient choices to be picked for investment domain.

ETL (MS SSMS, MS Excel and Python)

In this section we are going to describe the ETL procedure followed for this Assignment with as many details and screenshots as possible.

Description of the Dataset

The initial dataset was a csv file containing 18 columns:

- app_name
- genre
- rating
- reviews
- cost_label
- rate_5_pc
- rate_4_pc
- rate_3_pc
- rate_2_pc
- rate_1_pc
- updated
- size
- installs
- current_version
- requires_android
- content_rating
- in_app_products
- offered_by

We choose to rename all the columns to a more formal and representable title, remove the columns, and add an indicator which will not be used in the analysis procedures but will be very useful for the cleaning and transforming evaluation procedures to redefine and transform the initial cost label and in-app products columns into two different columns, to separate the string values from measures like the reviews, size and in-app products columns (word total and per item or M and k from size). All the aforementioned steps, among others, will be explained in the data cleaning and transformation stage.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
app_name	genre	rating	reviews	cost_label	rate_5_pc	rate_4_pc	rate_3_pc	rate_2_pc	rate_1_pc	updated	size	installs	current_version	requires_an	content_ratin	in_app_products	offered_by
Sonic the Hedgehog Classic	Platform	4.1	193,898 total	Install	68.0272	10.2041	5.44218	3.40136	12.9252	March 24,	61M	10,000,00(	3.5.1	4.4 and up	Rated for 3+	₫ 45,000 per item	SEGA
Push'em all	Action	3.8	119,679 total	Install	56.1798	12.9213	8.42697	5.05618	17.4157	April 16, 2	41M	10,000,00(	1.14	4.4 and up	Rated for 3+	₫ 92,000 per item	VOODOO
Sky Fighters 3D	Action	4.2	196,685 total	Install	67.1141	11.4094	7.38255	3.3557	10.7383	May 14, 2(	19M	10,000,00(	1.5	4.0 and up	Rated for 7+	₫ 42,000 - ₫ 2,100,000 pe	Doodle Mobile Ltd.

Figure 1: Initial CSV Preview

Therefore, the dataset which settled for cleaning was containing the following 19 columns (after the data transformation procedure):

- Id
- App Name
- Publisher
- App Size
- Genre
- Price
- isFree
- Rating
- Total Reviews

- Rating_Perc_5
- Rating_Perc_4
- Rating_Perc_3
- Rating_Perc_2
- Rating_Perc_1
- Total Installations
- Required Android OS
- Content Rating
- In-App Purchases MIN
- In-App Purchases MAX

Data Cleaning and Transformation

The selected dataset, as mentioned before, contains missing data, numerical string values, and various currencies that need to be transformed into a specific one. Therefore, if we can clean and transform them, we can make our analysis more efficient.

Removal of unnecessary initial columns

To begin with, we removed the 'current_version' column because many values were missing or would be outdated since the mobile application updated on a daily basis, so we concluded that it would not contribute to our analysis.

Addition of Id Row-Indicator

Before starting the cleaning and transforming procedures, we thought that a row indicator would be helpful to evaluate mainly the automated executed R and Python scripts and queries. Therefore, we choose to implement the addition of this column since it will not affect our analysis and reporting tasks.

In-app products transformation

Then, we convert all the currencies of the 'in-app products' column to euro currency (previously US dollars and Vietnamese đồng). However, to do that, we used the MS Excel functionality to separate the column into two new columns regarding the in-app products min and max value (the in-app purchasable min and max cost amount per item), we removed the string words 'per item', and then we filter the columns per initial currency and divided with the exchange rate for every conversion. Finally, we remove the US dollars and Vietnamese đồng currency sign in order to make these two columns measures tables as floats. We thought the min and max differentiation would be more efficient for our analysis.

Removal of unnecessary strings from numeric columns and transformation to floats and integers

We follow the same procedure we used above in order to remove the unnecessary string part from the numeric columns and then transform it to numeric (int and float formats). To be more precise, we remove the word 'total' from the reviews' columns and the '+' sign from the installations' column.

Transformation and Conversion to megabytes and then to the numeric column of the app's size

Observing the initial dataset, we noticed that the app's size column separated its observations into M (megabytes) and k (kilobytes). Therefore, we choose to completely remove the string characters (M, k) from the column after converting all the kilobytes metrics to megabytes through a filtered selection of the kilobytes observations.

Removal of non-ASCII characters

A portion of the data of the initial dataset contained non-ASCII characters, usually at the App Name and Publisher fields. Therefore, we had to clean those data and completely remove those characters, since our implementation was to build a data warehouse and not just an analysis case study.

Deletion of all the rows that did not include an App Name

Since the App Name is the main column of our dataset, and the number of rows was no more the 350 (307), we chose to remove all the rows of the dataset that did not include an App Name.

Rename all columns

Since it will not affect our dataset and maybe even assist our analysis, we choose to rename all the columns of the dataset to a more representative and formal one.

Tools used for cleaning

In this section we are going to describe the tools that used to clean the initial flat file dataset and perform the above transformations.

R - Tidyverse

Our first choice was to perform the whole cleaning phase with R language and more specifically the popular R's library Tidyverse. We opted for this since Tidyverse is a R package designed for data science and excel at cleaning dataframes. Because we could not perform the cleaning stage properly and fulfill our needs, through R and Tidyverse, we choose to move on an alternative option.

Figure 2: R logo

Figure 3: R's Tidyverse

Python - Pandas

Then we move forward to a more familiar method to perform the cleaning stage for our initial flat file. To be more precise, we select the Python programming object-oriented language and with the assistance of one of the most famous libraries among all the programming languages, the Pandas, we perform our first stage cleaning to the dataset. Regarding Pandas, is an open-source Python library that is most widely used for data science, data analysis and machine learning tasks. It is built on top of another Python's package named Numpy, which provides support for multi-dimensional arrays. Therefore, the selection of Python and Pandas was a non brainer.

With these tools, we had a more detailed look at our dataset, we re-adjusted the dataset's columns by removing the not related to our analysis ones, renaming the columns to more representative and formal name - as mentioned above, and removing the rows which had an empty App Name field (the sample screenshot describes this procedure).

Figure 4: Python logo

Figure 5: Python's Pandas.

A snapshot of the code regarding the removal of rows with an empty App Names field, follows below:

cleanning.py

```
# importing pandas as pd
import pandas as pd

# Read the csv file and construct the dataframe
df = pd.read_csv('GoogleApps.csv')

# Visualize the dataframe
print(df.head(15))

# Print the shape of the dataframe
print(df.shape)

# Filter all rows for which the player's
# age is greater than or equal to 25
df_filtered = df[df['App Name'] == ' ']

# Print the new dataframe
print(df_filtered.head(15))

# Print the shape of the dataframe
print(df_filtered.shape)
```

Figure 6: Python's snapshot

Microsoft Excel

Afterwards, we move on to the Microsoft Excel platform since it offers a variety of tools that can be used in a simple manner. Such helpful tools are the sort filters, lookup under conditions and replace a specific string sequence of features. Therefore, through Microsoft Excel we perform most of the transformation procedures that was described above such as the removal of string values from numeric vectors, the conversion of the prices from US dollars and Vietnam dong, the conversion to megabytes of the App Size, separate the initially one in-app purchasable products to two new columns the in app purchases MIN and the in app purchases MAX, the addition of the row indicator Id, for evaluating purposes, and completely remove the non-ASCII characters from the initial dataset (the following cleaning procedure sample snapshot depicts this procedure).

Figure 7: Microsoft Excel logo

A snapshot of a completely removal of a non-ASCII characters from the dataset follows below.

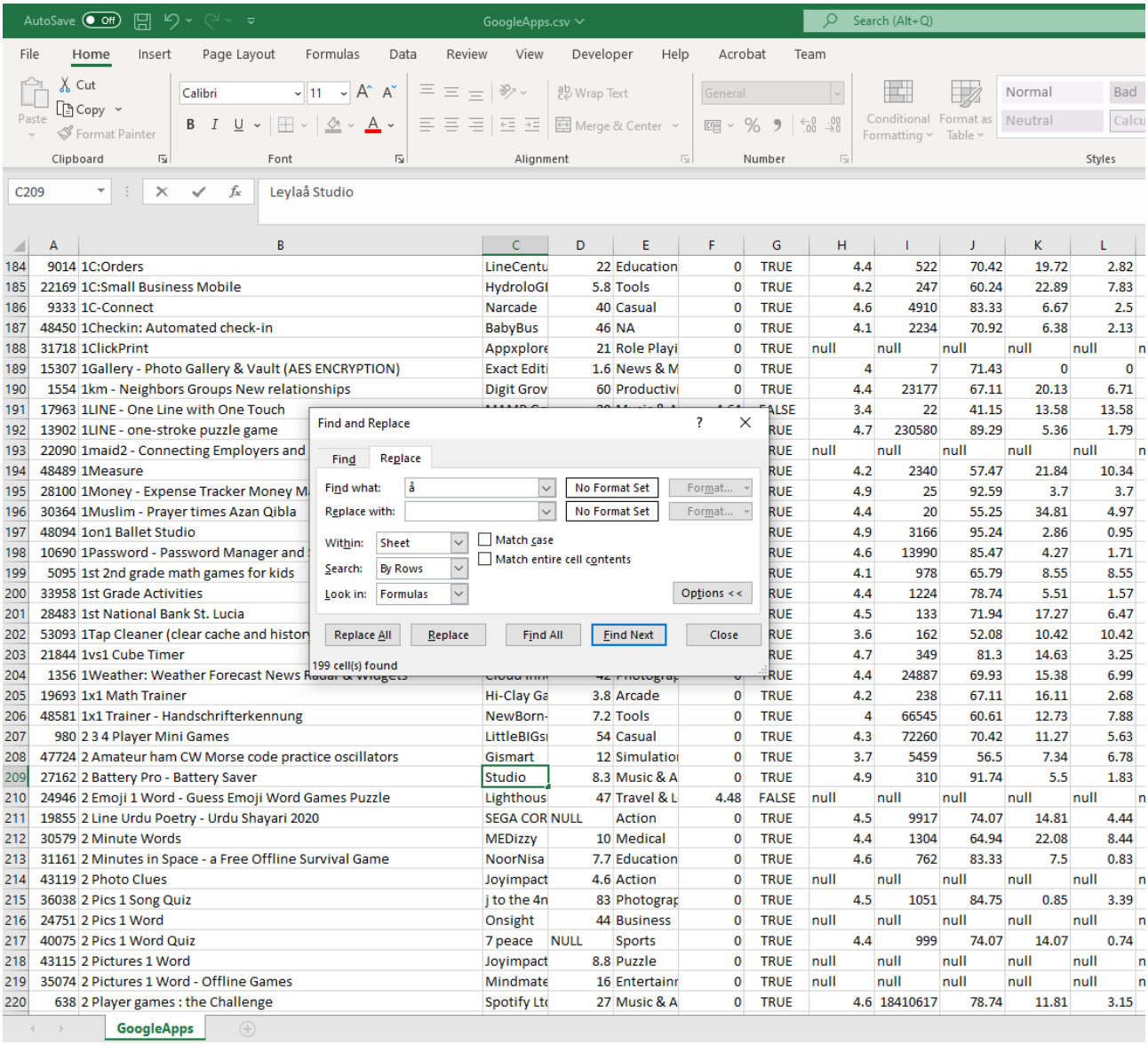


Figure 8: Microsoft Excel's Find and Remove non ASCII sample snapshot

Data Cleaning Flow Diagram

In this section a flow chart diagram follows, which summarizes the way that we performed our cleaning to our initial dataset.

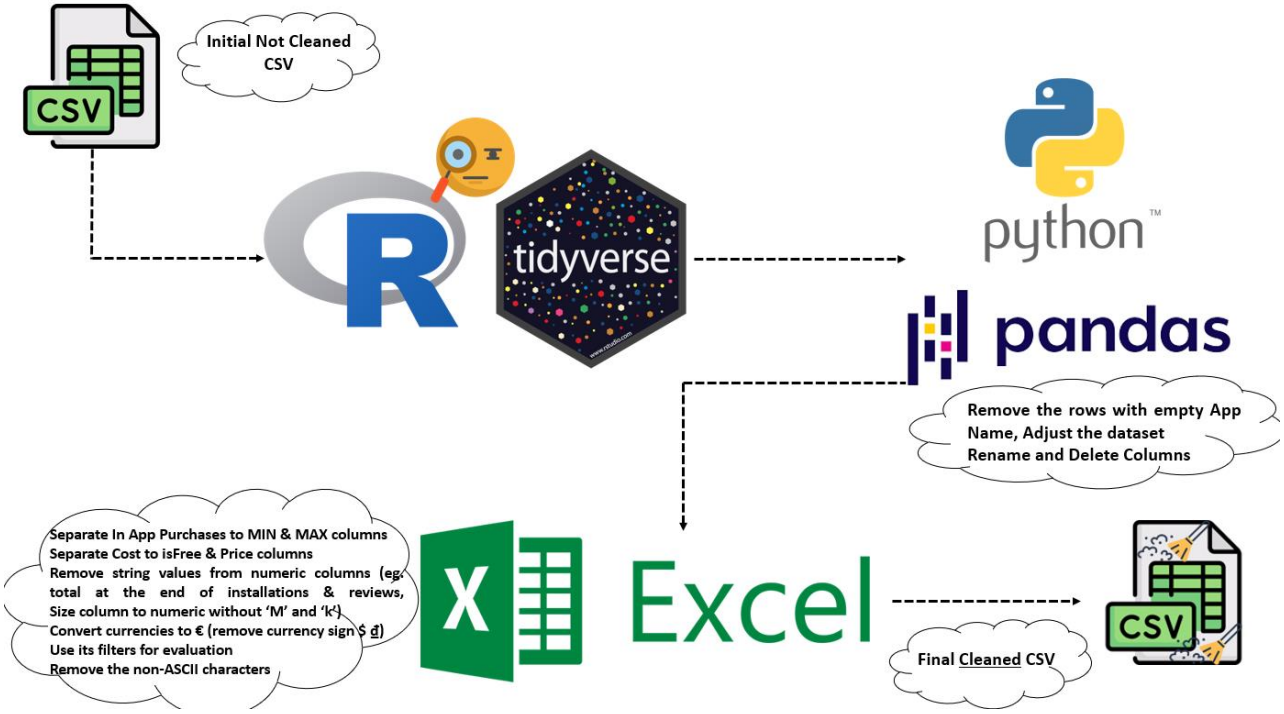


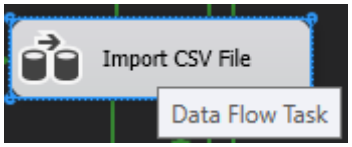
Figure 9: Data Cleaning Flow Diagram

Data Loading

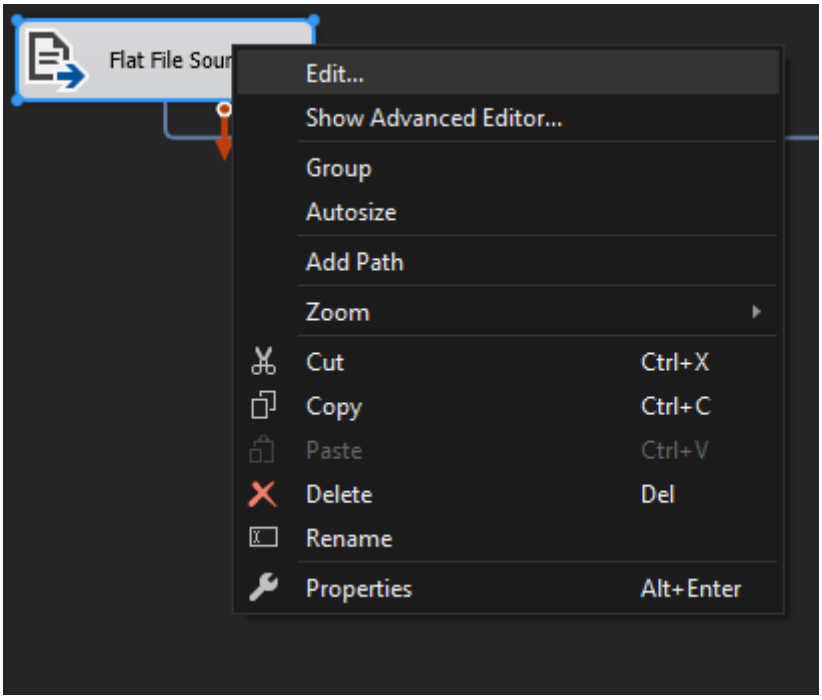
In this section we are going to explain the methodology regarding the loading process of this assignment in Microsoft SQL Server Management Studio (SSMS) and then in the Integrations Services Project of Microsoft Visual Studio (SSDT).

Flat File Connection Manager Setup

The first step is the creation of the database in Microsoft SQL Server Management Studio (SSMS) and in a new Integration Services Project in Visual Studio (SSDT). Then, a new Data Flow task was created in the SSDT Project that loads the file through a Flat File Connection Manager section.

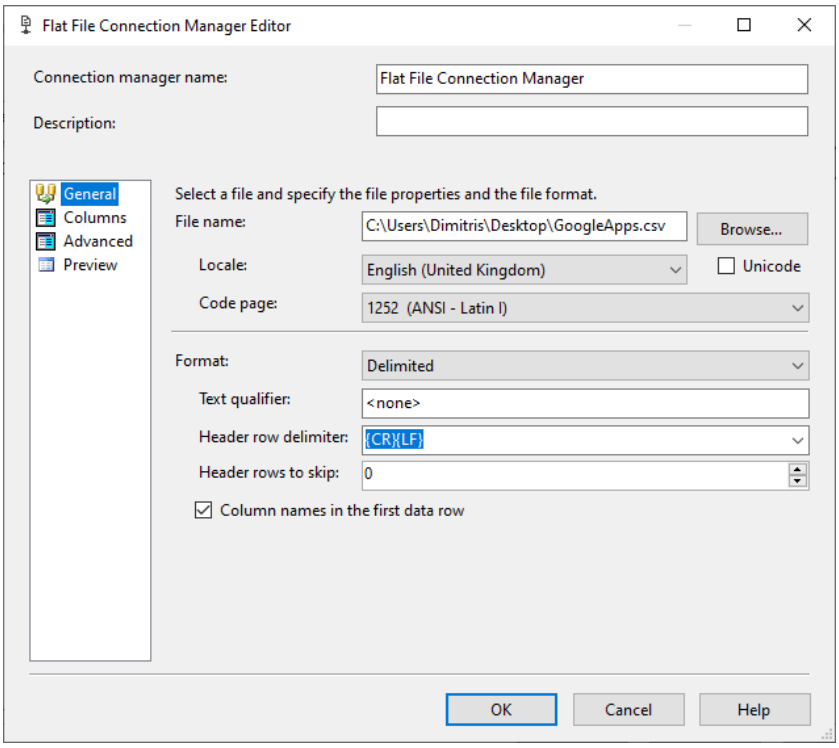


Data Loading Procedure 1



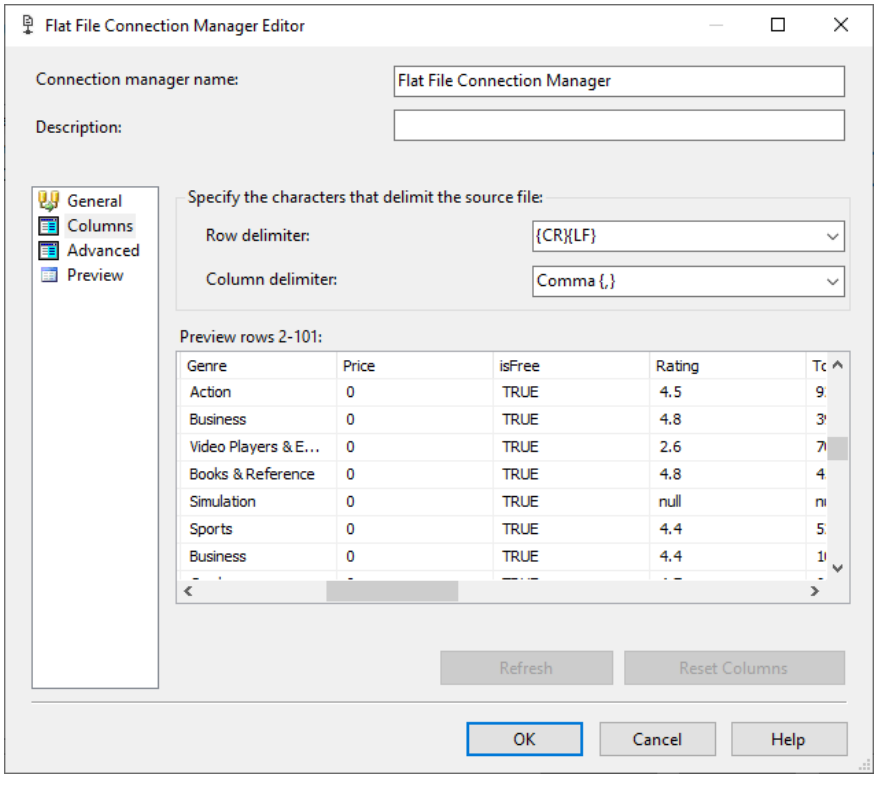
Data Loading Procedure 2

From this menu, we select the Flat File (CSV) that represents our data source (dataset) for this assignment (GoogleApps.csv) and we define the encoding of the CSV file, as well as the delimiters pre row and per column.



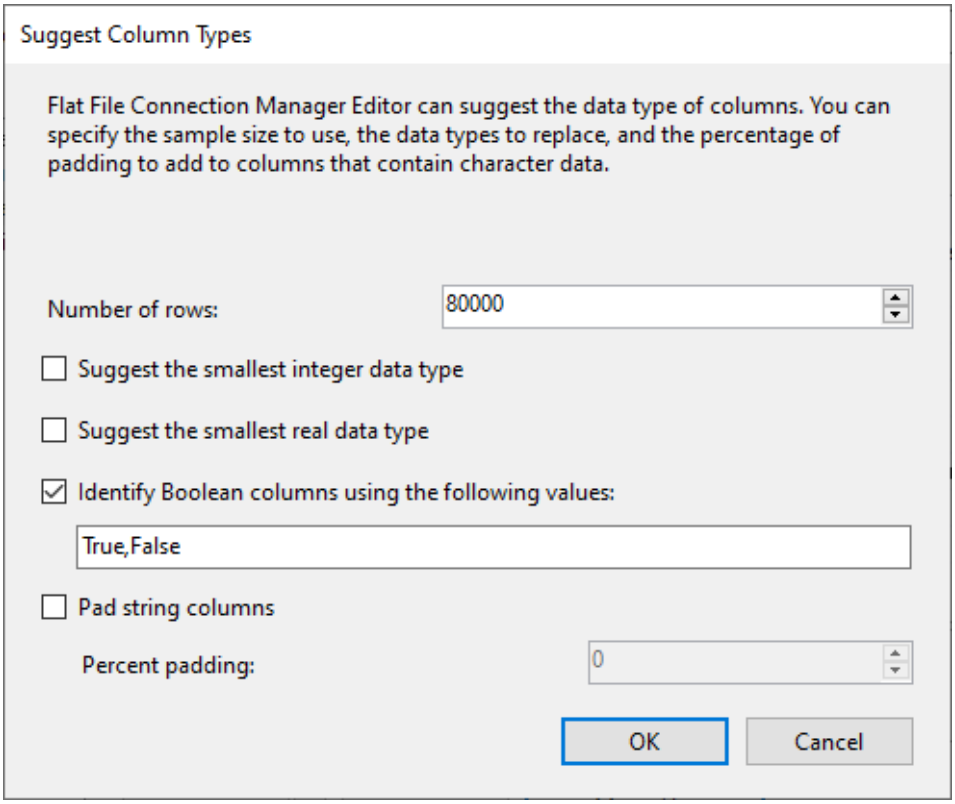
Data Loading Procedure 3

Afterward, we can see the Flat File Connection Manager's sample of the loaded data in order to ensure that we will load the correct file in the way we want to.

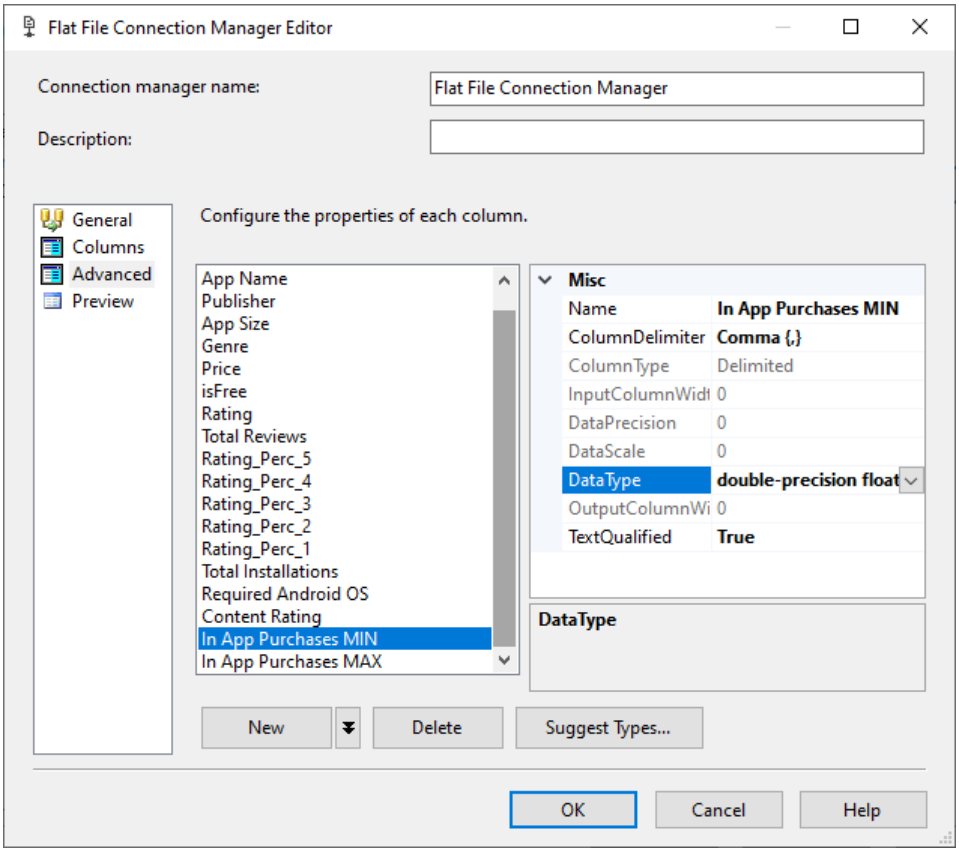


Data Loading Procedure 4

Then, in the advantage section from the left menu, we need to clarify the types of the columns. Microsoft's Visual Studio can automatically try to predict the types of the columns by collecting data for a pre-defined number of rows (80000). Therefore, this procedure does not consistently predict the wanted results, so we need to modify the type for some of the columns. You can see the procedure below:

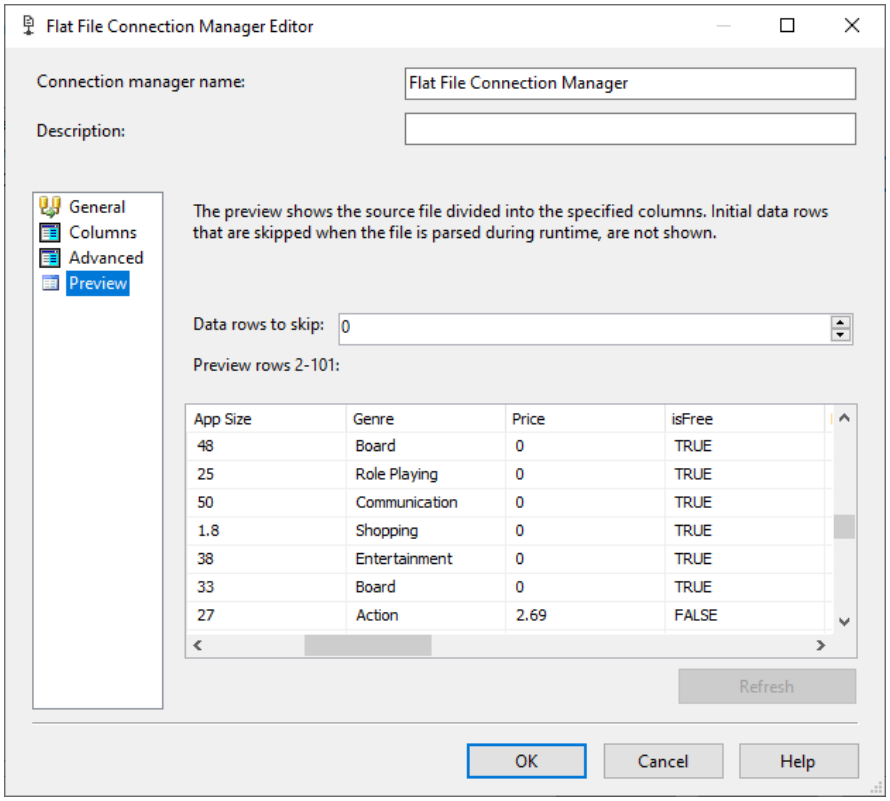


Data Loading Procedure 5



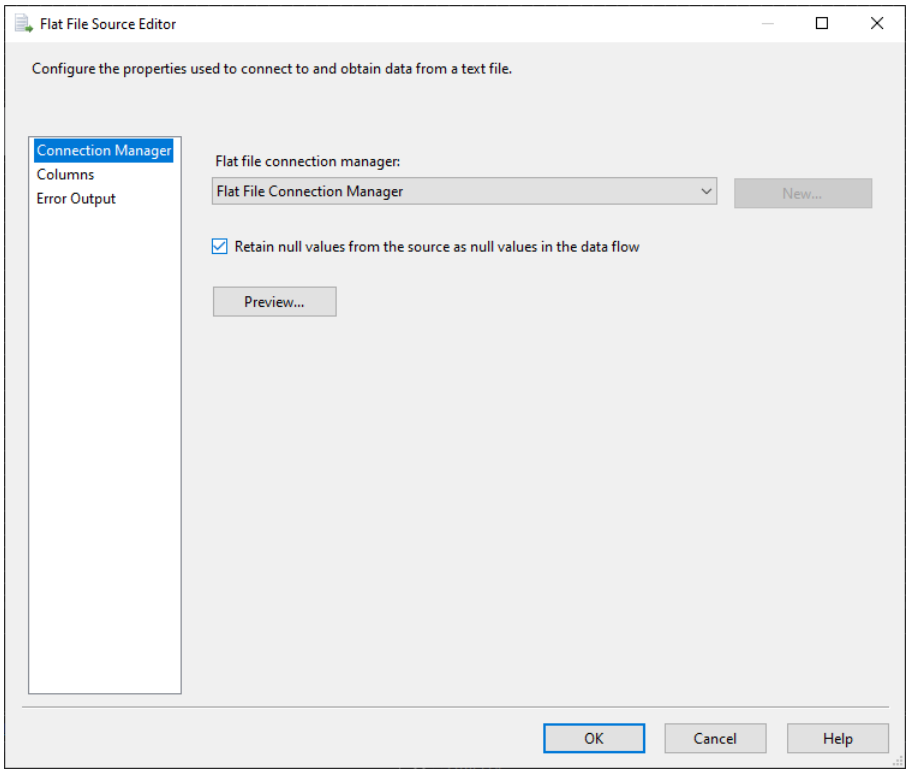
Data Loading Procedure 6

Finally, we can see a preview of the to-be-inserted dataset from the Flat File Connection Manager.

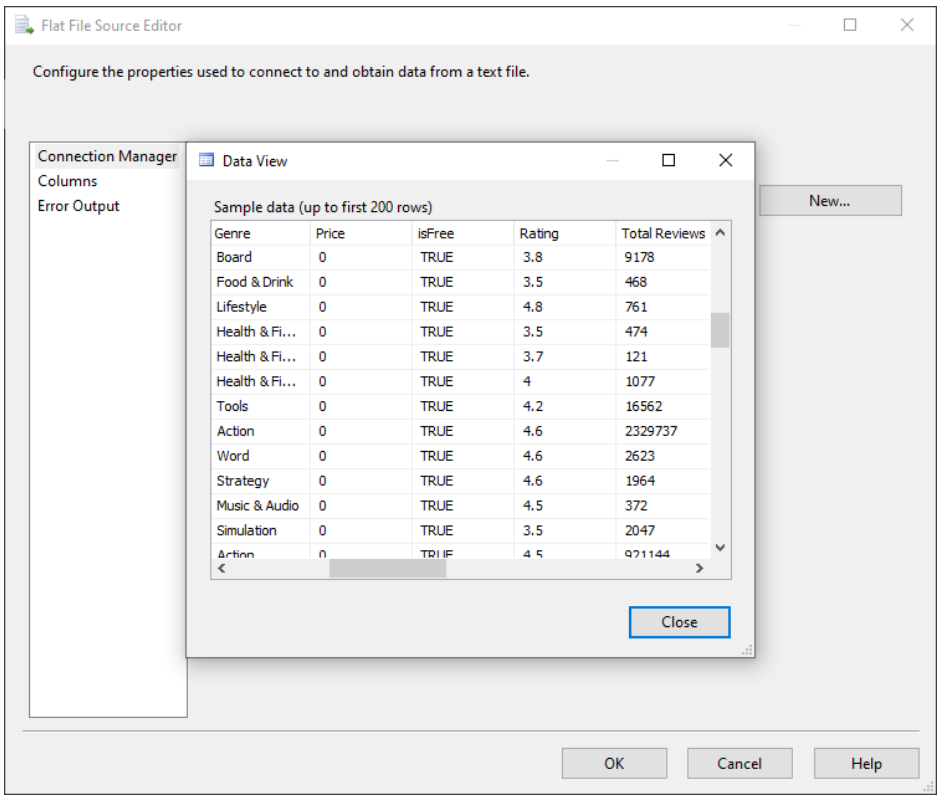


Data Loading Procedure 7

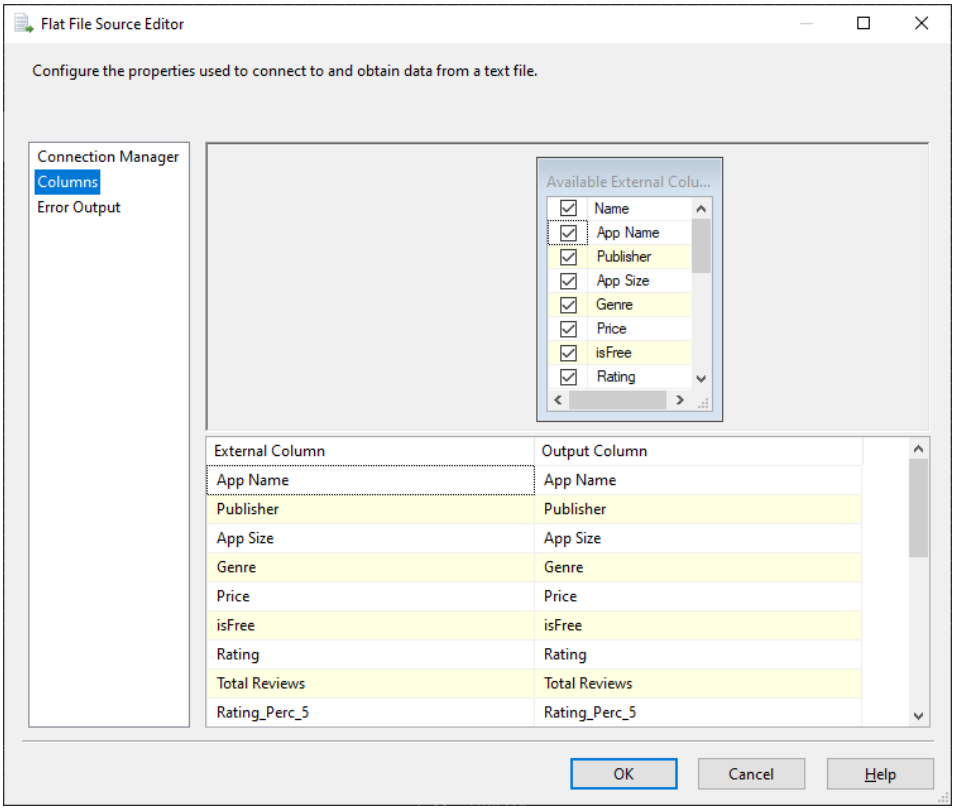
Then, we can always look at and supervise the created Flat File Connection Manager from the right-click menu, where we can further modify it, if needed. Some sample pictures of this procedure following below.



Data Loading Procedure 8



Data Loading Procedure 9

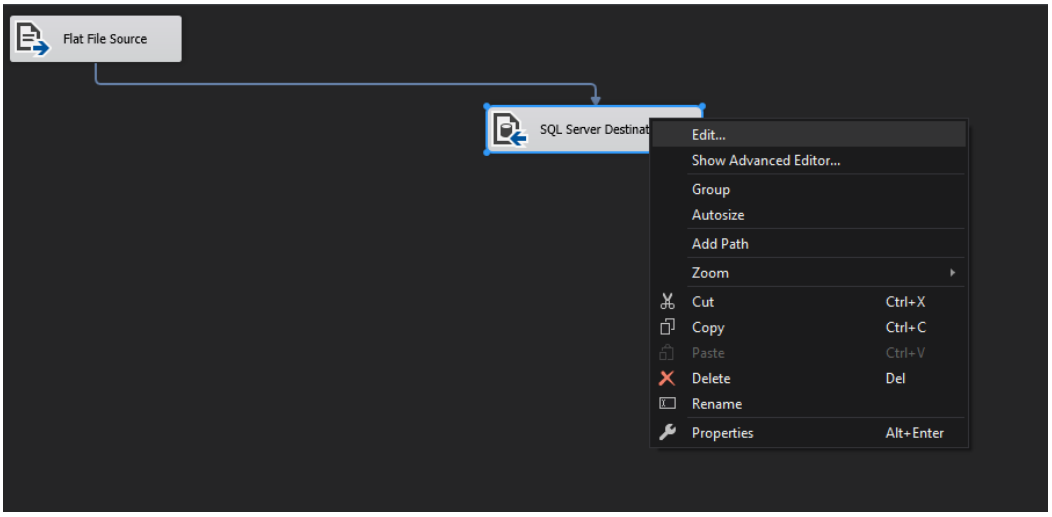


Data Loading Procedure 10

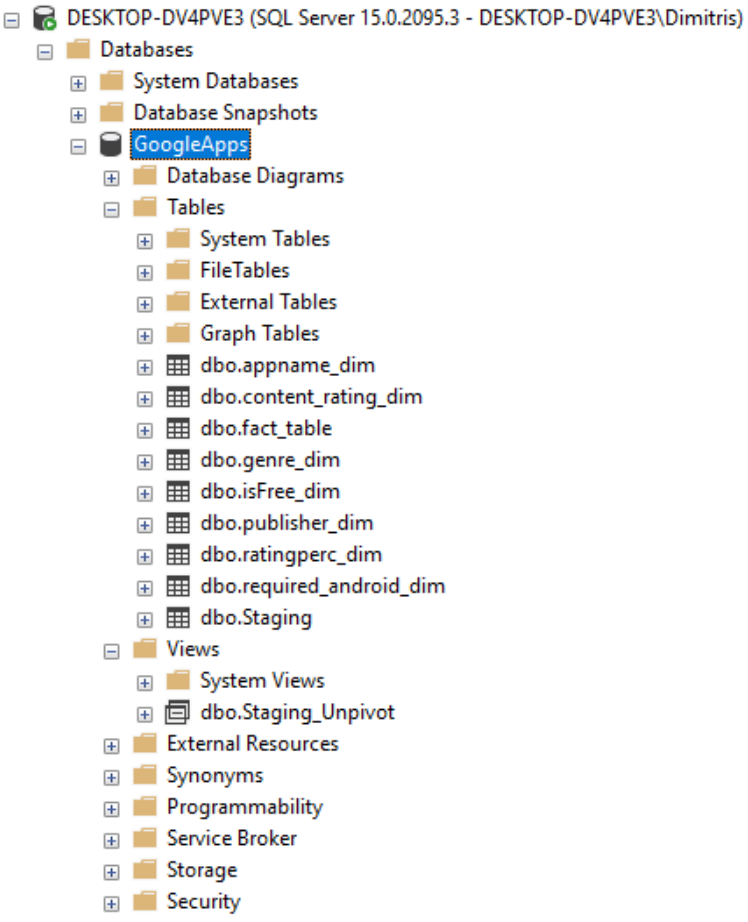
SQL Server Destination - Database Setup

Now, after we complete the above procedure regarding the Flat File Connection Manager Setup, we creating an SQL Server Destination component, in order to parse the dataset to the SQL Server Management Studio (SSMS).

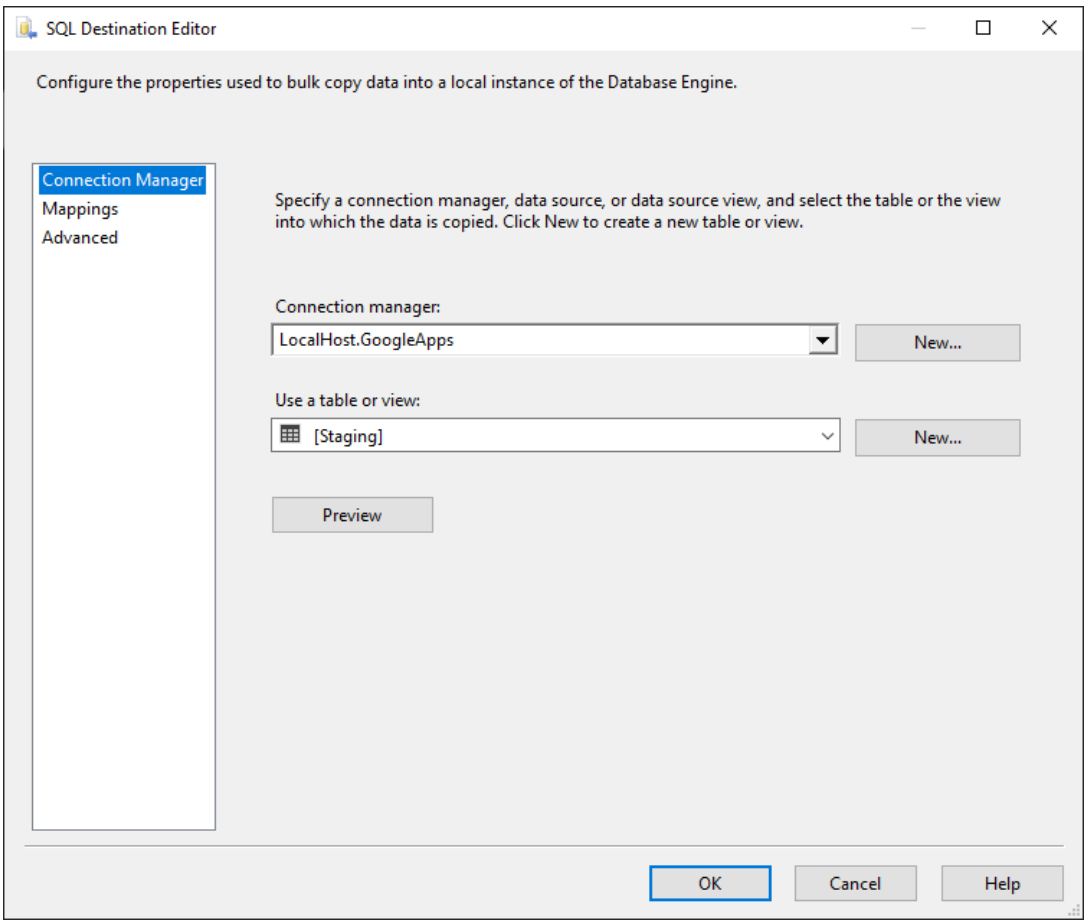
We right-click the SQL Server Destination component and we select the Edit option. Then, we select the Connection Manager to be the recently created database in SQL Server Management Studio (belongs to localhost 'server') and the created table where it will parse the data.



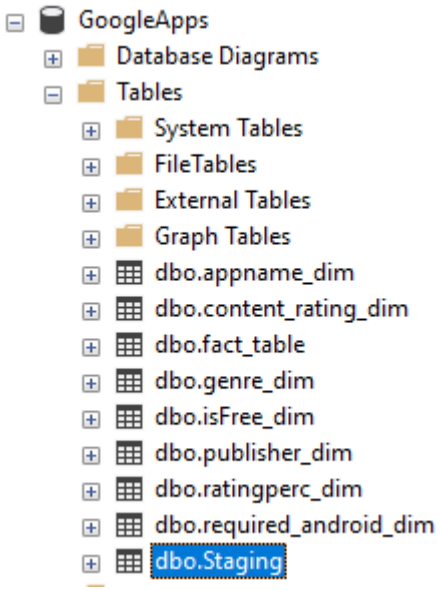
Data Loading Procedure 11



Data Loading Procedure 12

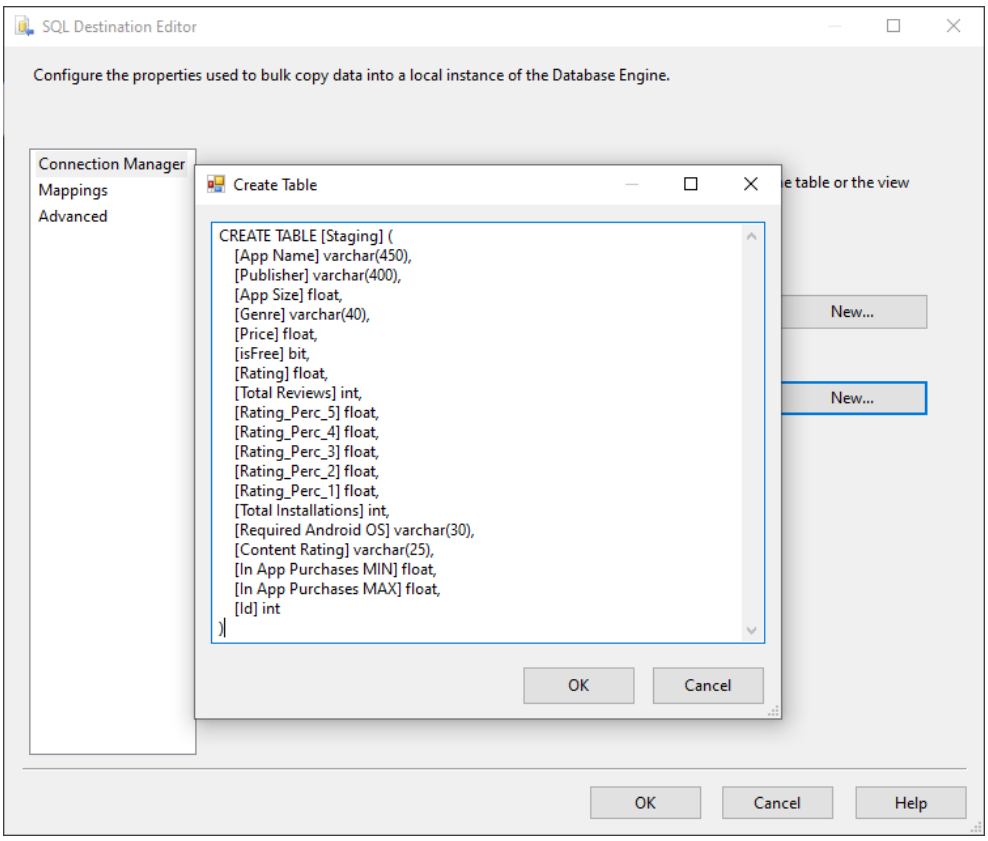


Data Loading Procedure 13

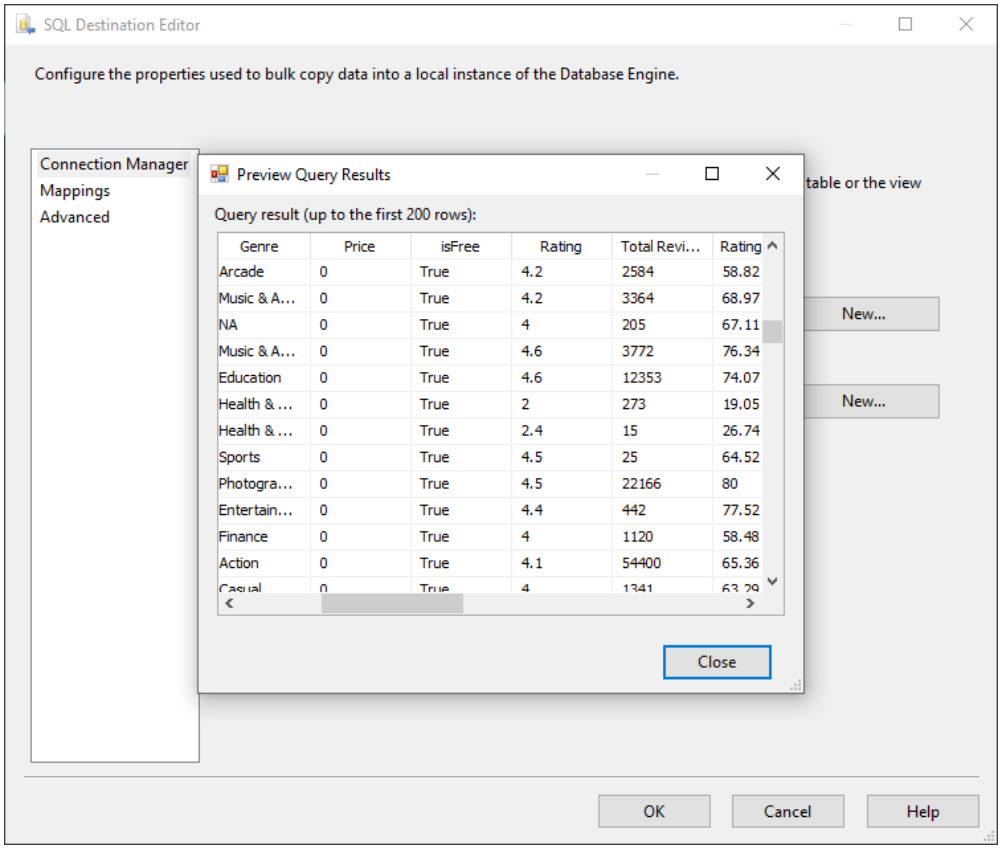


Data Loading Procedure 14

Sequentially, we can notice that the SQL Destination Editor, creates the columns of the new table with the suggested types of the Flat File Connection Manager. This way the database will maintain the column types that were previously defined and will have a consistency between the database and the SSDT.

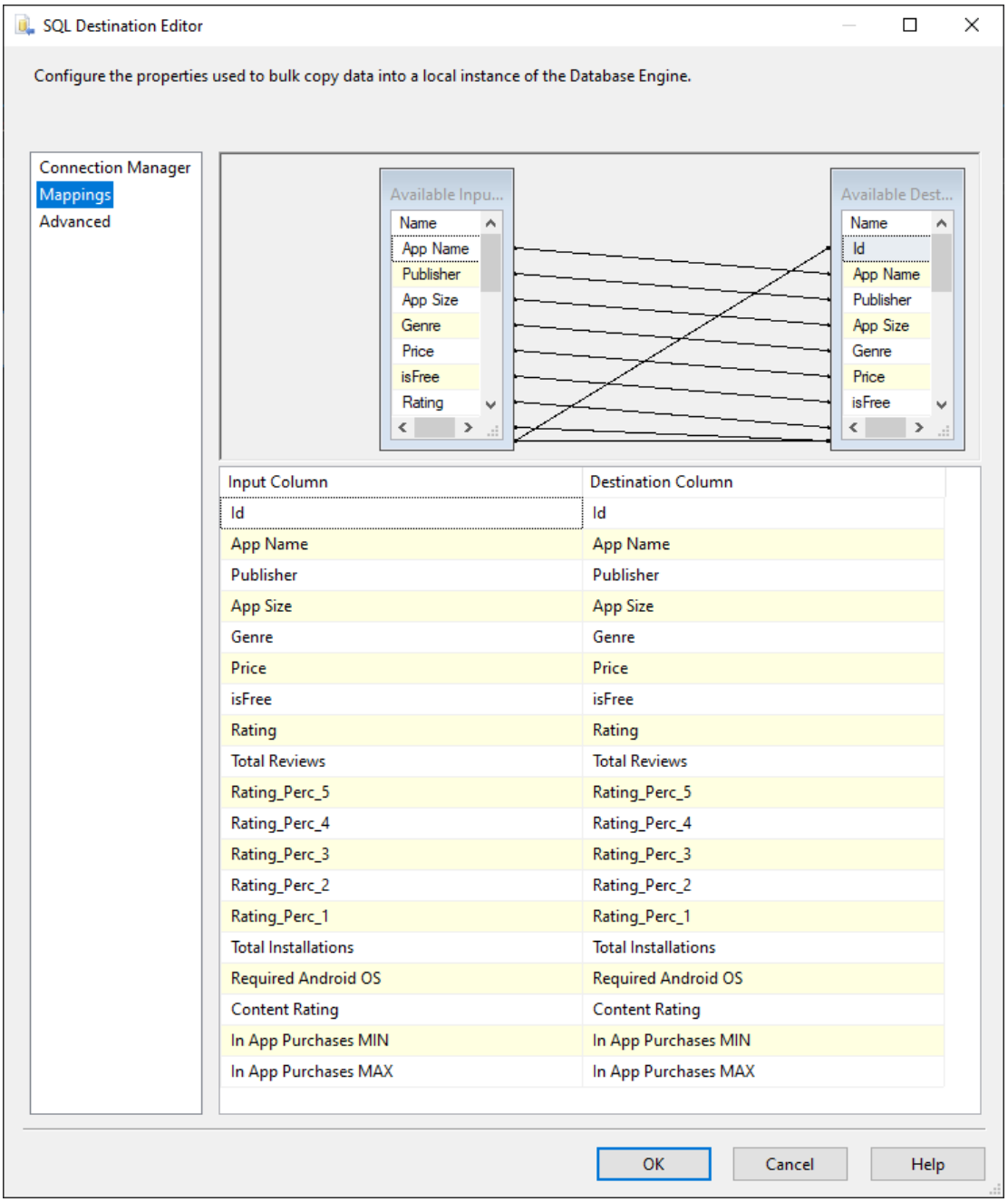


Data Loading Procedure 15

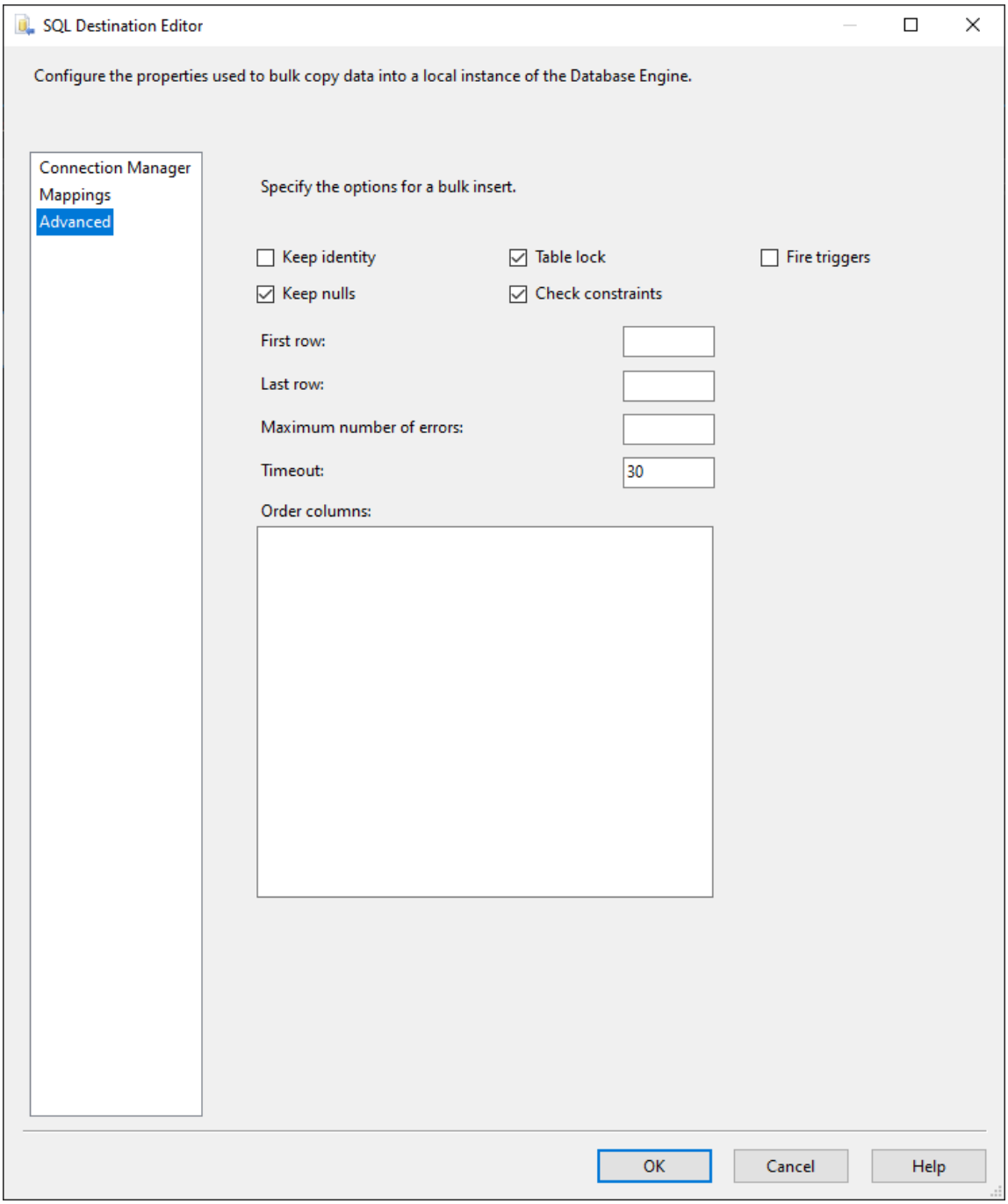


Data Loading Procedure 16

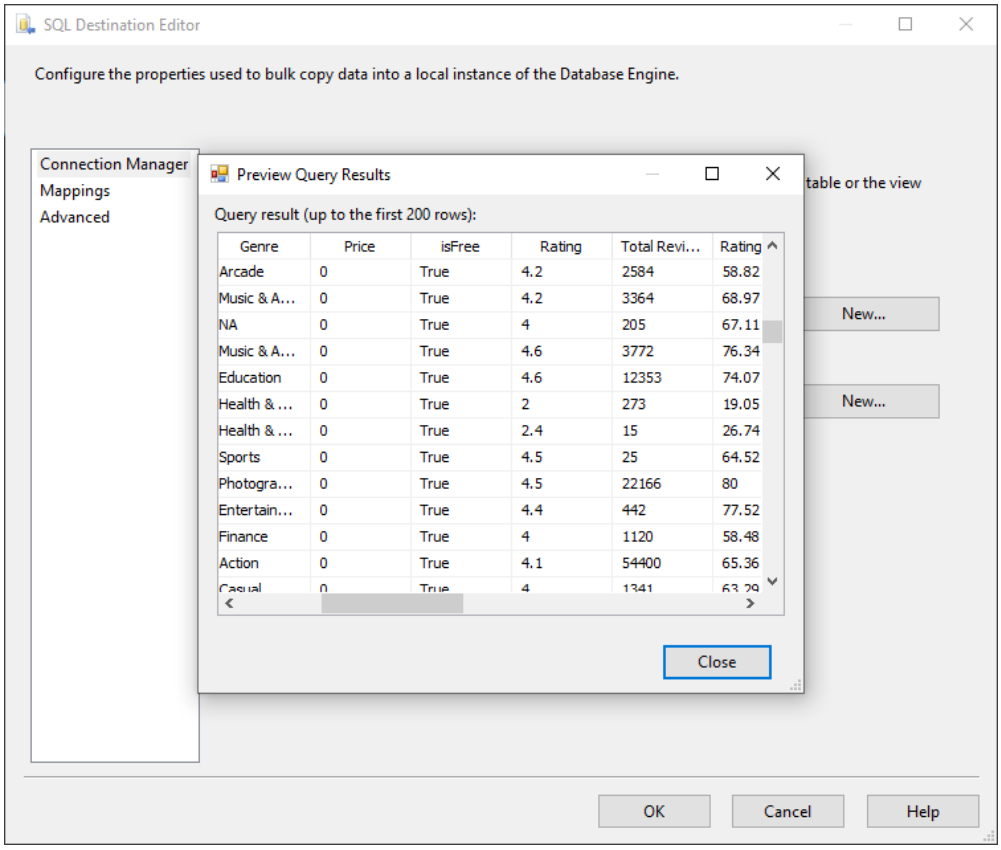
We can always monitor and modify the SQL Destination Editor by clicking the Edit option. From there we can look at a data preview of the parsed data, a columns association map and the advanced setting that the parse procedure have.



Data Loading Procedure 17



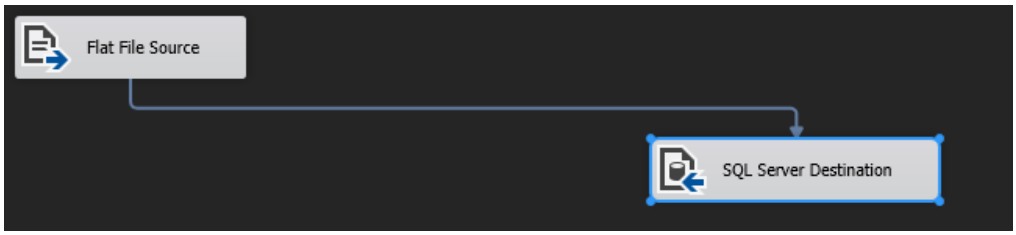
Data Loading Procedure 18



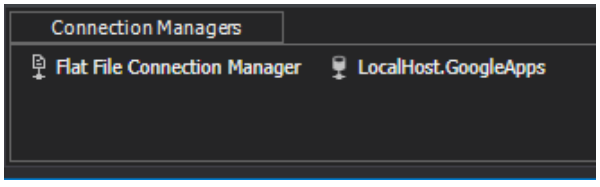
Data Loading Procedure 19

Connection of the Flat File Source and the SQL Server Destination

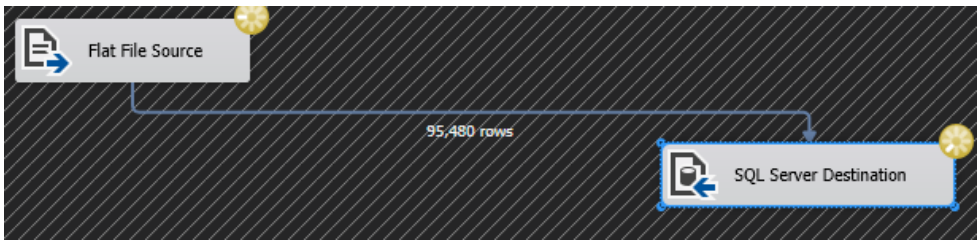
After creating the two aforementioned components, we now connect them and executing the procedure in order to see that the SSMS database will get the Flat File's data in the ordered way.



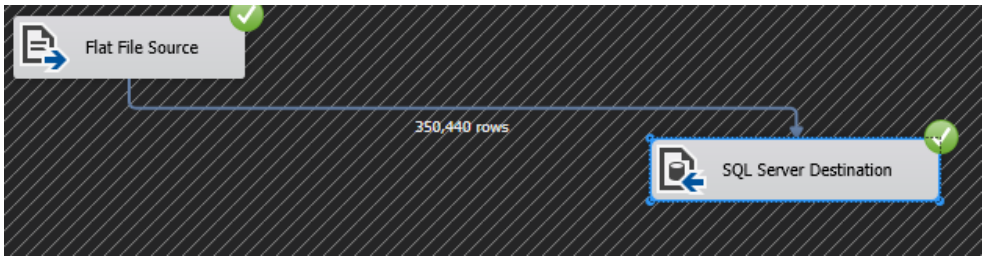
Data Loading Procedure 20



Data Loading Procedure 21



Data Loading Procedure 22



Data Loading Procedure 23

As you see from the above pictures, the dataset has been successfully connected with the SSMS database and all the rows have been inserted (350,440 observations or rows). A confirmation picture from the database follows below:

SQLQuery1.sql - DE...PVE3\Dimitris (61))

```
/****** Script for SelectTopNRows command from SSMS  ******/
SELECT TOP (1000) [App Name]
      ,[Publisher]
      ,[App Size]
      ,[Genre]
      ,[Price]
      ,[isFree]
      ,[Rating]
      ,[Total Reviews]
      ,[Rating_Perc_5]
      ,[Rating_Perc_4]
      ,[Rating_Perc_3]
      ,[Rating_Perc_2]
      ,[Rating_Perc_1]
      ,[Total Installations]
      ,[Required Android OS]
      ,[Content Rating]
      ,[In App Purchases MIN]
      ,[In App Purchases MAX]
      ,[Id]
  FROM [GoogleApps].[dbo].[Staging]
```

100 %

Results | Messages

	App Name	Publisher	App Size	Genre	Price	isFree	Rating	Total Reviews	Rating_Perc_5	Rating_Perc_4	Rating_Perc_3	Rating_Perc_2	Rating_Perc_1
25	Learn German - 50 languages	Flashpoint Games, LLC	14	Puzzle	0	1	3.7	535	41.49	23.24	12.86	6.64	15.77
26	Learn German German Alphabet Speak German ...	Holiday Educationist	26	NA	0	1	3.6	33	52.08	8.85	8.85	5.73	24.48
27	Learn German Basics	QuiverVision Limited	24	NA	0	1	3.2	89	47.85	7.66	5.26	0.96	38.28
28	Learn German from scratch	Meditationsteps	6.9	Health & Fitness	0	1	4.8	2015	90.91	6.36	0.91	0.91	0.91
29	Learn German Vocabulary - Kids	Banggood	25	Shopping	0	1	4.2	230000	66.23	13.91	5.96	1.99	11.92
30	Learn German Vocabulary Free	Dictamp	41	Books & Reference	0	1	4.1	159	57.47	21.84	7.47	2.3	10.92
31	Learn German with Assimil	Big Ocean Studio	33	Tools	0	1	4.6	2780	81.97	10.66	3.28	0	4.1
32	Learn German words with Smart-Teacher	DesenvDroid	1.1	Music & Audio	0	1	4.8	48542	89.29	7.14	1.79	0	1.79
33	Learn German WordsVerbsArticles with Flashcards	Moocall	15	Business	0	1	4	101	53.48	22.99	5.88	1.6	16.04
34	Learn German. Speak German	SvS Teaching System	2.8	Education	0	1	4.6	72	85.47	5.13	2.56	0	6.84
35	Learn German: Die Bienenretter	Learn Teach Explore Sp. z o.o.	38	NA	3.35	0	4.4	22	78.13	8.59	0	0	13.28
36	Learn GoLang	JRummy Apps	2.5	Tools	5.3	0	3.6	25619	53.48	7.49	6.42	7.49	25.13
37	Learn Greek - 50 languages	sport media group GmbH	0.5	Sports	0	1	3.8	7192	49.51	23.76	5.45	3.47	17.82
38	Learn Guitar with Simulator	Milan Bibic	36	News & Magazines	0	1	4.6	5378	78.13	14.06	2.34	1.56	3.91
39	Learn Guitar: Chords - 3000 Chords	RRT Developers Educational Apps	30	Education	0	1	4.6	2146	78.13	13.28	3.13	0.78	4.69
40	Learn Hebrew - 50 languages	Appocalypse	3.6	Education	0	1	4.6	1213	73.53	22.06	1.47	0.74	2.21
41	Learn Hebrew - FunEasyLearn	Hoardings Inc.	4.6	Educational	0	1	4.4	1287	72.46	12.32	6.52	2.17	6.52
42	Learn Hebrew Basics	G-Unit Technologies	42	Education	0	1	4.5	104	78.13	10.16	4.69	0.78	6.25

Data Loading Procedure 24

Creating Staging Unpivot View

The dataset contains five numerical variables that represent the percentages of apps that are rated with one different rating grade of one, two, three, four, or five stars. Thus, it is helpful to create one additional categorical variable of the rating class with five levels (one star rating, two star rating, three star rating, four star rating, and five star rating) and one new numerical variable that expresses the percentage of installations of each application that is rated with the corresponding grade. In this scope, the categorical variable of Rating Class and the numerical variable of Rating Perc are created, and the numerical variables Rating Perc 1, Rating Perc 2, Rating Perc 3, Rating Perc 4, and Rating Perc 5 are abolished. To be more specific, the names of the five abolished variables are contained in the new Rating Class variable, and the numerical values of the five abolished variables are contained in the new Rating Perc variable. The practical implementation of this unpivot procedure is carried out in an SQL task that takes the initial variables from the staging table that hosts the imported CSV and creates a new view with the unpivot structure named staging unpivot view. The code and the output of the materialized view are shown below.

The code executed, in order to create this unpivot view is presented below:

```
SELECT  [Id],
        [App Name],
        [Publisher],
        [App Size],
        [Genre],
        [Price],
        [isFree],
        [Rating],
        [Total Reviews],
        [Rating Perc],
        [Rating Classes],
        [Total Installations],
        [Required Android OS],
        [Content Rating],
        [In App Purchases MIN],
        [In App Purchases MAX]
FROM    (SELECT [Id],
                [App Name],
                [Publisher],
                [App Size],
                [Genre],
                [Price],
                [isFree],
                [Rating],
                [Total Reviews],
                [Rating_Perc_1],
                [Rating_Perc_2],
                [Rating_Perc_3],
                [Rating_Perc_4],
                [Rating_Perc_5],
                [Total Installations],
                [Required Android OS],
                [Content Rating],
                [In App Purchases MIN],
                [In App Purchases MAX]
        FROM    [GoogleApps].[dbo].[Staging]) pvt
        UNPIVOT ([Rating Perc] FOR [Rating Classes] IN ([Rating_Perc_1], [Rating_Perc_2], [Ra
ting_Perc_3], [Rating_Perc_4], [Rating_Perc_5])) AS unpvt
```

The results of this unpivot view are presented below.

	Id	App Name	Publisher	App Size	Genre	Price	isFree	Rating	Total Reviews	Rating Perc	Rating Classes	Total Installations	Required Android OS	Content Rating	In App Purchases MIN	In App Purchases MAX
58	30234	Learn French from scratch	Eyepetizer Inc.	11	Ente...	0	1	4.6	2382	3.97	Rating_Perc_3	100000	4	Rated for 12	NULL	NULL
59	30234	Learn French from scratch	Eyepetizer Inc.	11	Ente...	0	1	4.6	2382	13.49	Rating_Perc_4	100000	4	Rated for 12	NULL	NULL
60	30234	Learn French from scratch	Eyepetizer Inc.	11	Ente...	0	1	4.6	2382	79.37	Rating_Perc_5	100000	4	Rated for 12	NULL	NULL
61	34791	Learn French Language: Li...	Zhang Dongd...	19	Tools	0	1	4.5	184	3.62	Rating_Perc_1	50000	4	Rated for 3	1.75	2.65
62	34791	Learn French Language: Li...	Zhang Dongd...	19	Tools	0	1	4.5	184	1.45	Rating_Perc_2	50000	4	Rated for 3	1.75	2.65
63	34791	Learn French Language: Li...	Zhang Dongd...	19	Tools	0	1	4.5	184	5.8	Rating_Perc_3	50000	4	Rated for 3	1.75	2.65
64	34791	Learn French Language: Li...	Zhang Dongd...	19	Tools	0	1	4.5	184	16.67	Rating_Perc_4	50000	4	Rated for 3	1.75	2.65
65	34791	Learn French Language: Li...	Zhang Dongd...	19	Tools	0	1	4.5	184	72.46	Rating_Perc_5	50000	4	Rated for 3	1.75	2.65
66	34784	Learn French Phrases	I-Education	3.6	Edu...	0	1	5	40	0	Rating_Perc_1	500	4	Rated for 3	NULL	NULL
67	34784	Learn French Phrases	I-Education	3.6	Edu...	0	1	5	40	0	Rating_Perc_2	500	4	Rated for 3	NULL	NULL
68	34784	Learn French Phrases	I-Education	3.6	Edu...	0	1	5	40	0	Rating_Perc_3	500	4	Rated for 3	NULL	NULL
69	34784	Learn French Phrases	I-Education	3.6	Edu...	0	1	5	40	0	Rating_Perc_4	500	4	Rated for 3	NULL	NULL
70	34784	Learn French Phrases	I-Education	3.6	Edu...	0	1	5	40	100	Rating_Perc_5	500	4	Rated for 3	NULL	NULL
71	20050	Learn French Vocabulary - ...	BlueSkySoft	4.1	Edu...	0	1	4.7	67	0.83	Rating_Perc_1	10000	4	Rated for 3	NULL	NULL
72	20050	Learn French Vocabulary - ...	BlueSkySoft	4.1	Edu...	0	1	4.7	67	2.48	Rating_Perc_2	10000	4	Rated for 3	NULL	NULL
73	20050	Learn French Vocabulary - ...	BlueSkySoft	4.1	Edu...	0	1	4.7	67	0.83	Rating_Perc_3	10000	4	Rated for 3	NULL	NULL
74	20050	Learn French Vocabulary - ...	BlueSkySoft	4.1	Edu...	0	1	4.7	67	13.22	Rating_Perc_4	10000	4	Rated for 3	NULL	NULL
75	20050	Learn French Vocabulary - ...	BlueSkySoft	4.1	Edu...	0	1	4.7	67	82.64	Rating_Perc_5	10000	4	Rated for 3	NULL	NULL

Figure 10: Staging Unpivot Results

Dimension - Metrics Definition

The next step is to create the fact table of our database schema. Before that, we need to define which categorical variables could be the dimensions and which numerical variables would be the metrics. Therefore, regarding our dataset, we have the following:

Dimensions:

- App Name
- Publisher
- Genre
- isFree
- RatingPerc (Unpivot)
- Required Android OS
- Content Rating

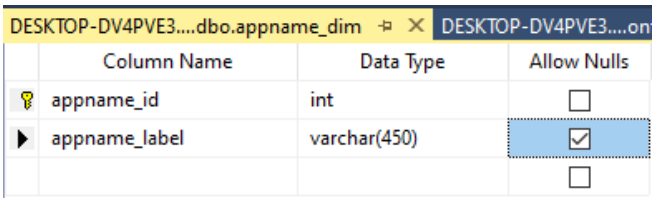
Metrics:

- Id (indicator)
- App Size
- Price
- Rating
- Total Reviews
- Rating_Perc_5
- Rating_Perc_4
- Rating_Perc_3
- Rating_Perc_2
- Rating_Perc_1
- Total Installations
- In-App Purchases MIN
- In-App Purchases MAX

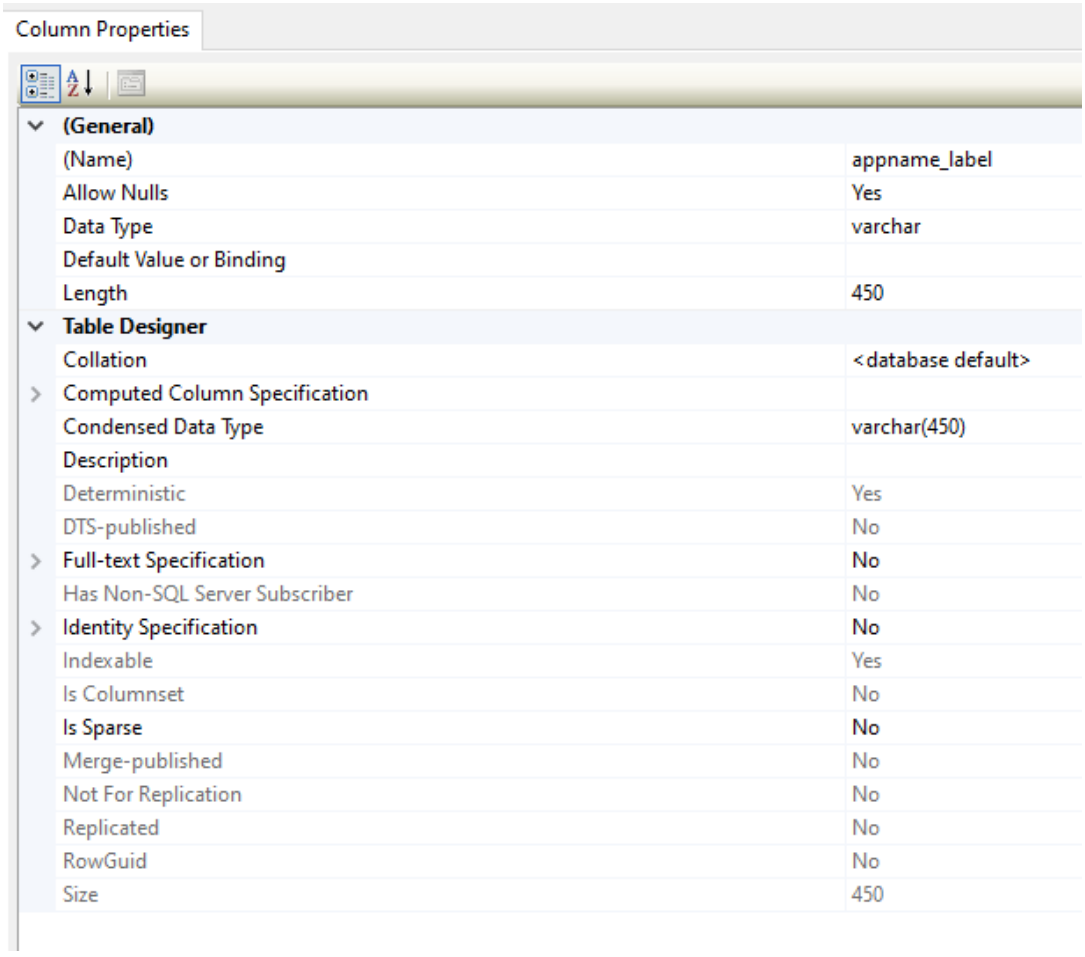
Dimension Creation

Now, for all the dimension we need to create the dimension table which will associate a distinct number (id) with every (distinct) column's label and will be used as a foreign key in the fact table. Therefore, the specifications used for the dimension table creations can be found in this section. First of all, the ID is a unique auto-increment value that starts from the value 1. The aforementioned method has been implemented to all the dimension table's id.

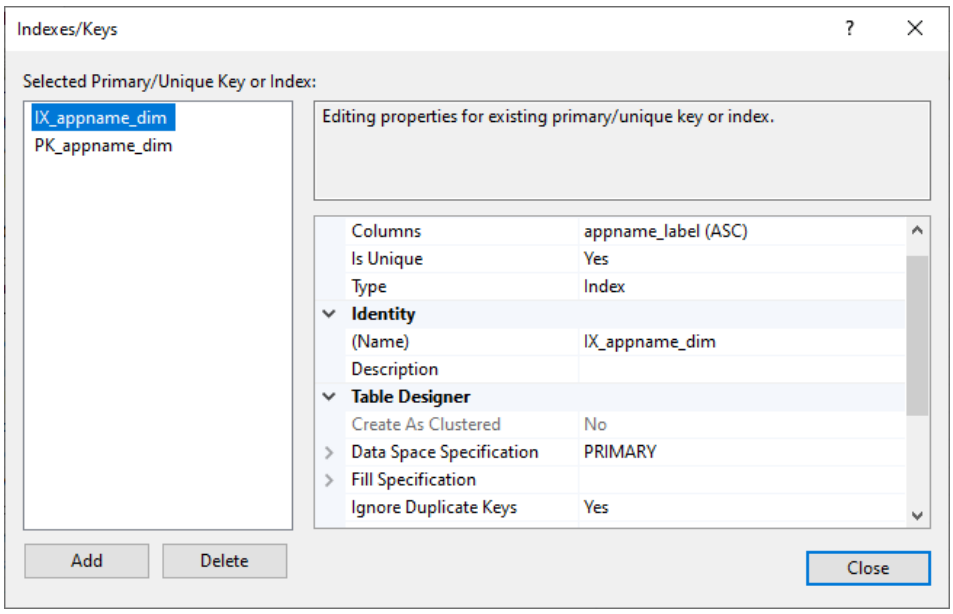
- appname_dim



Data Loading Procedure 25

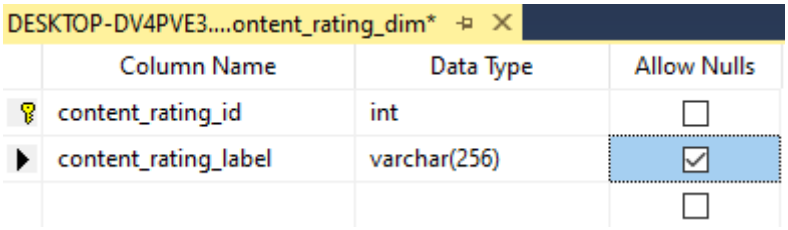


Data Loading Procedure 26

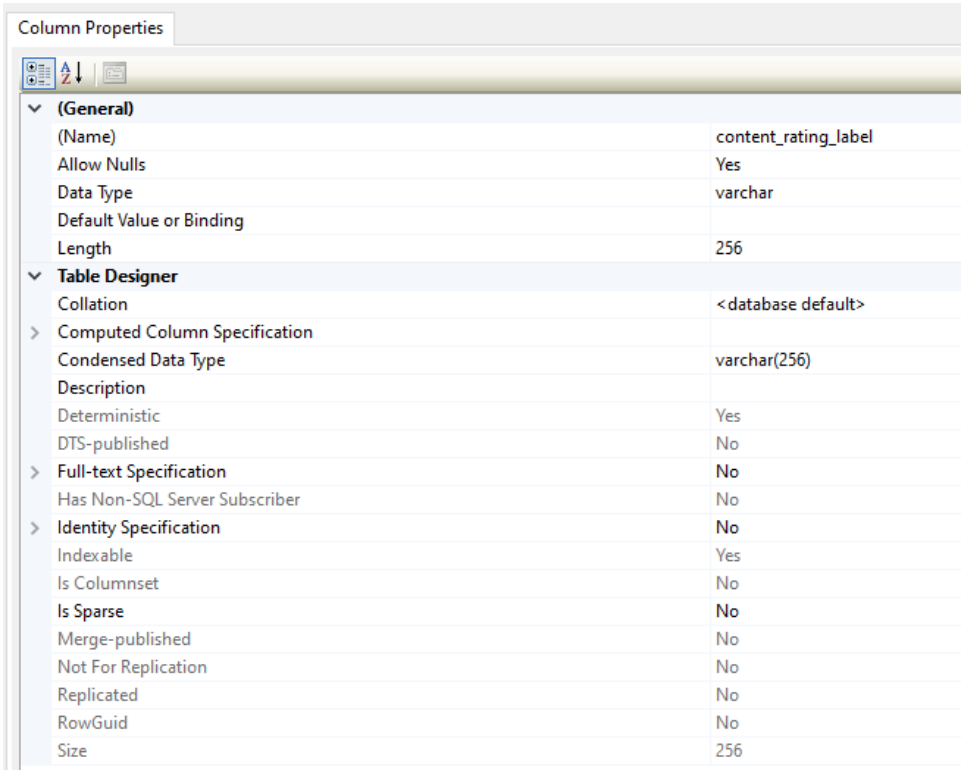


Data Loading Procedure 27

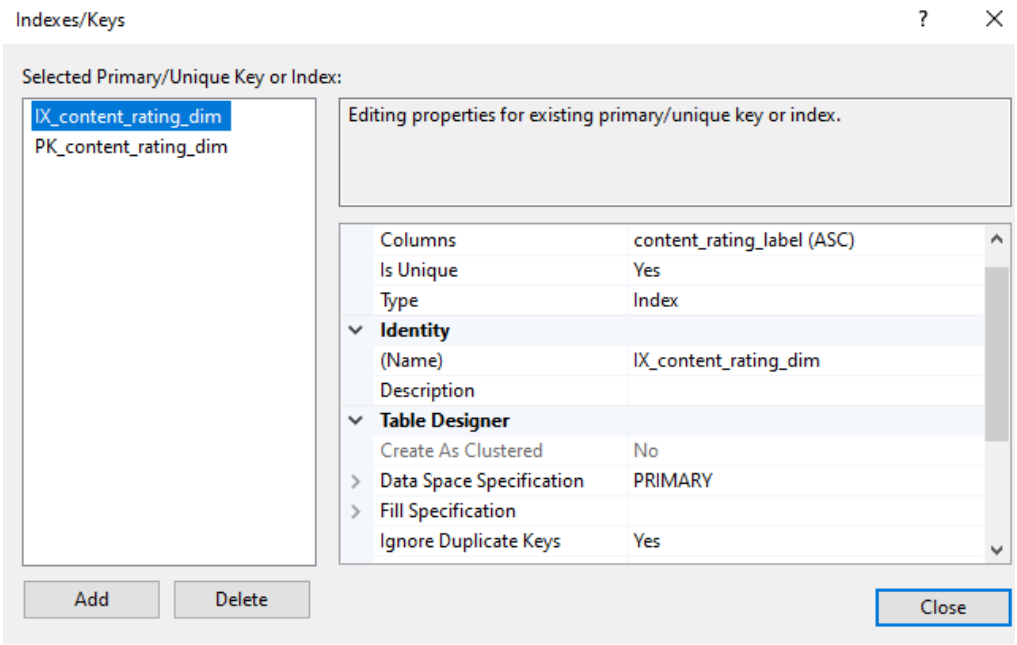
- content_rating_dim



Data Loading Procedure 28

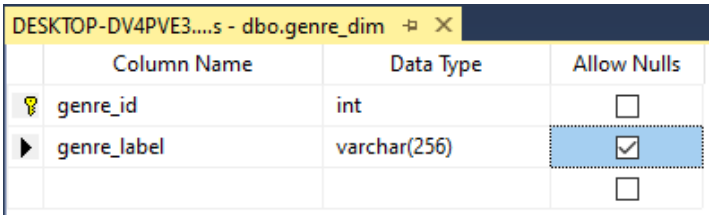


Data Loading Procedure 29

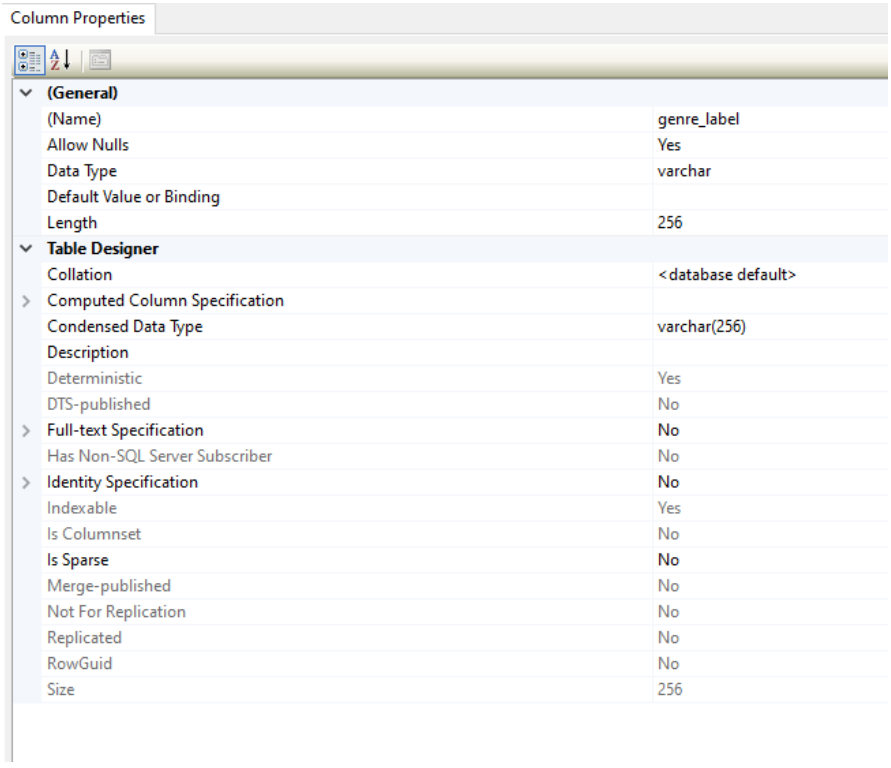


Data Loading Procedure 30

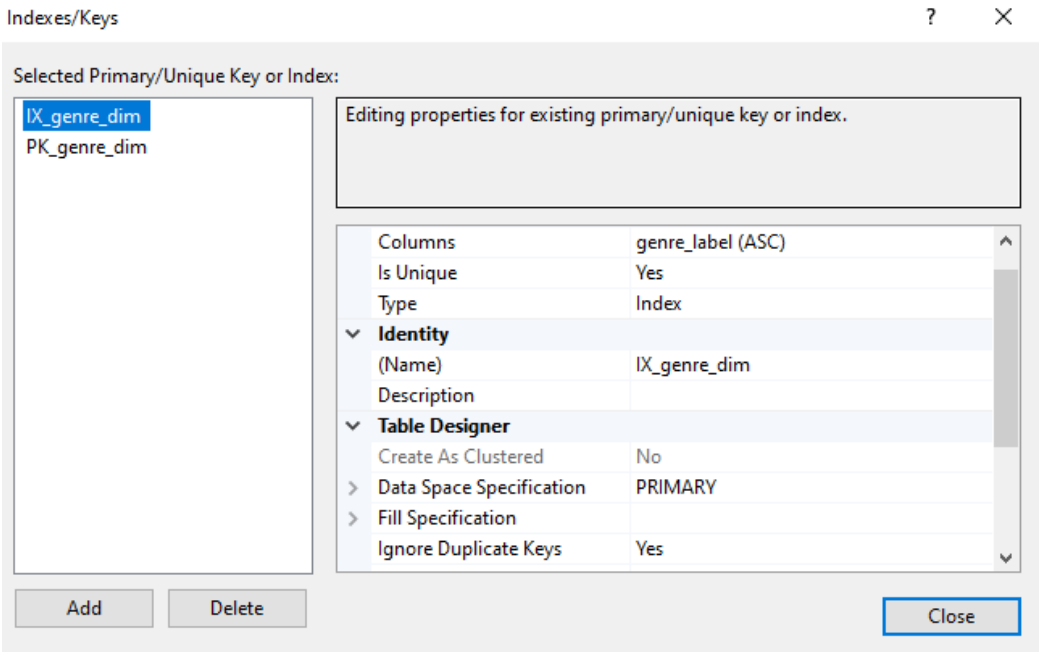
- genre_dim



Data Loading Procedure 31

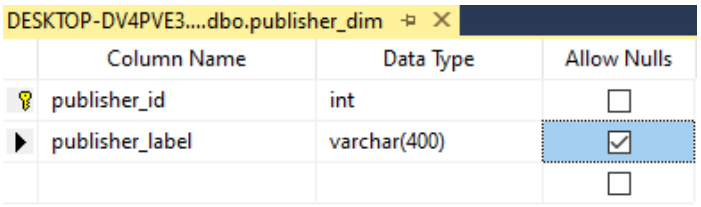


Data Loading Procedure 32

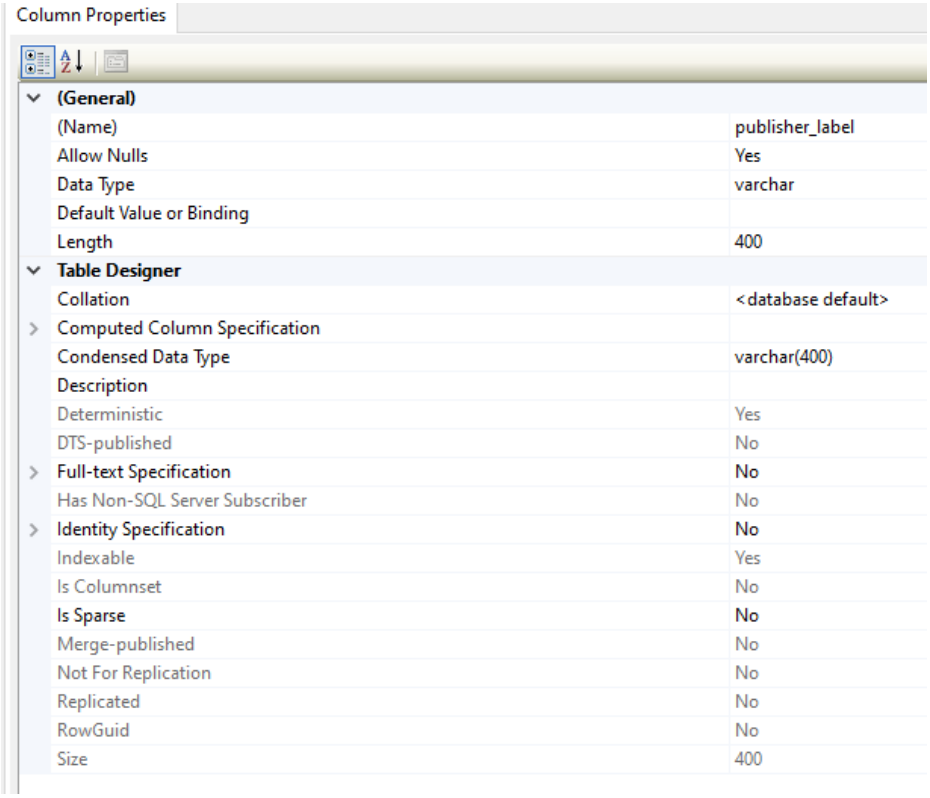


Data Loading Procedure 33

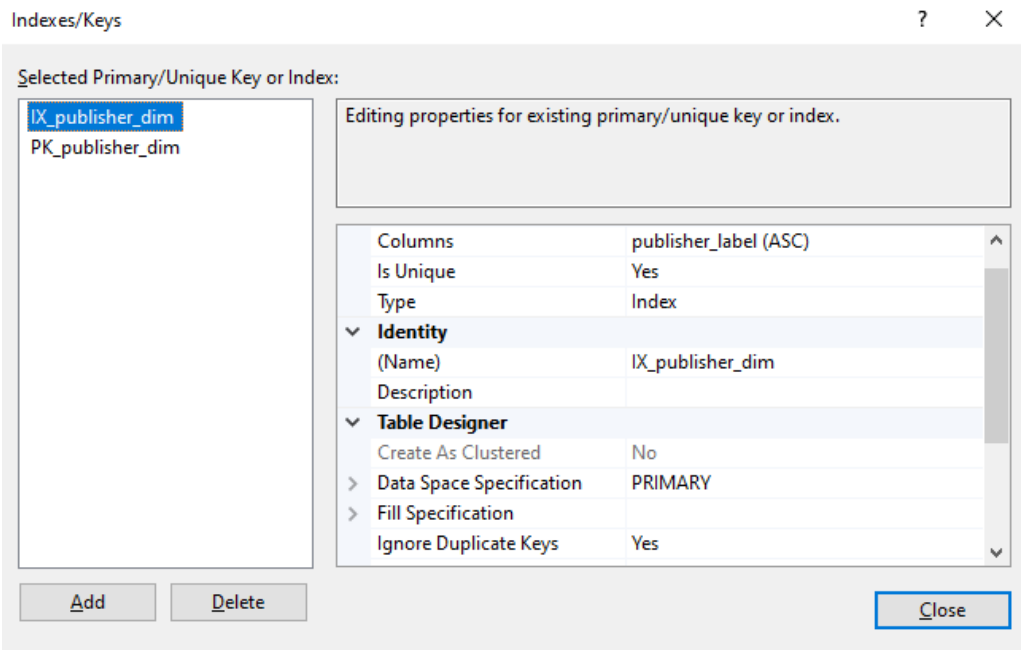
- publisher_dim



Data Loading Procedure 34

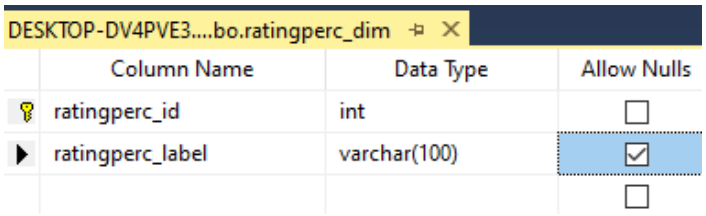


Data Loading Procedure 35

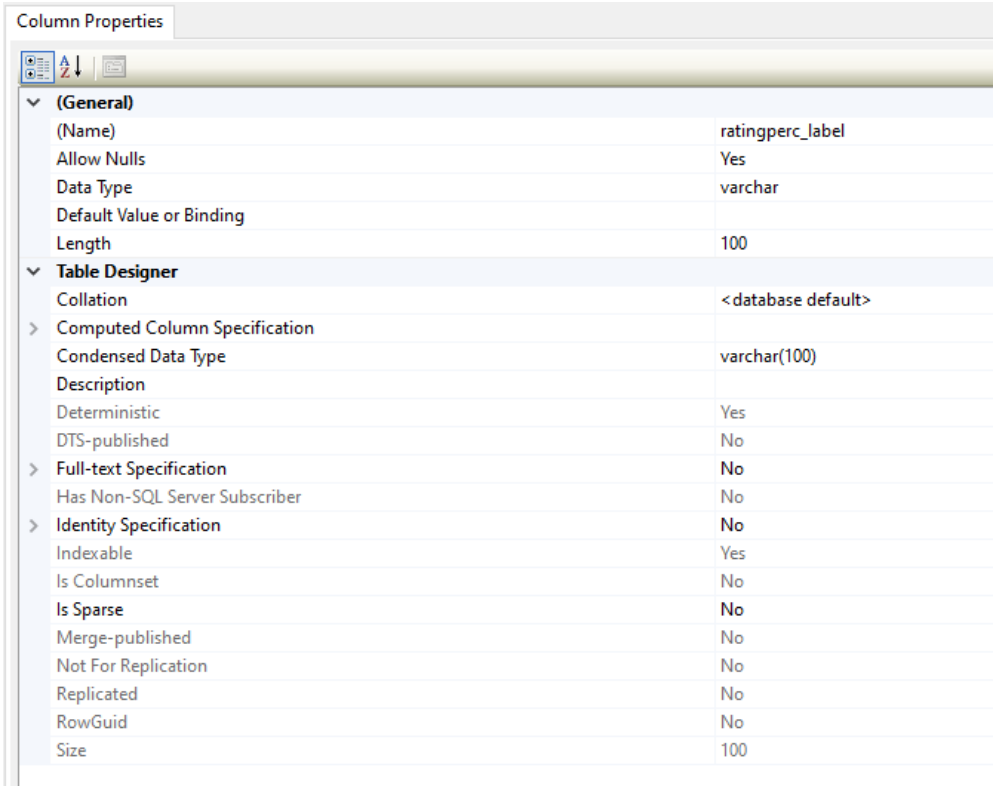


Data Loading Procedure 36

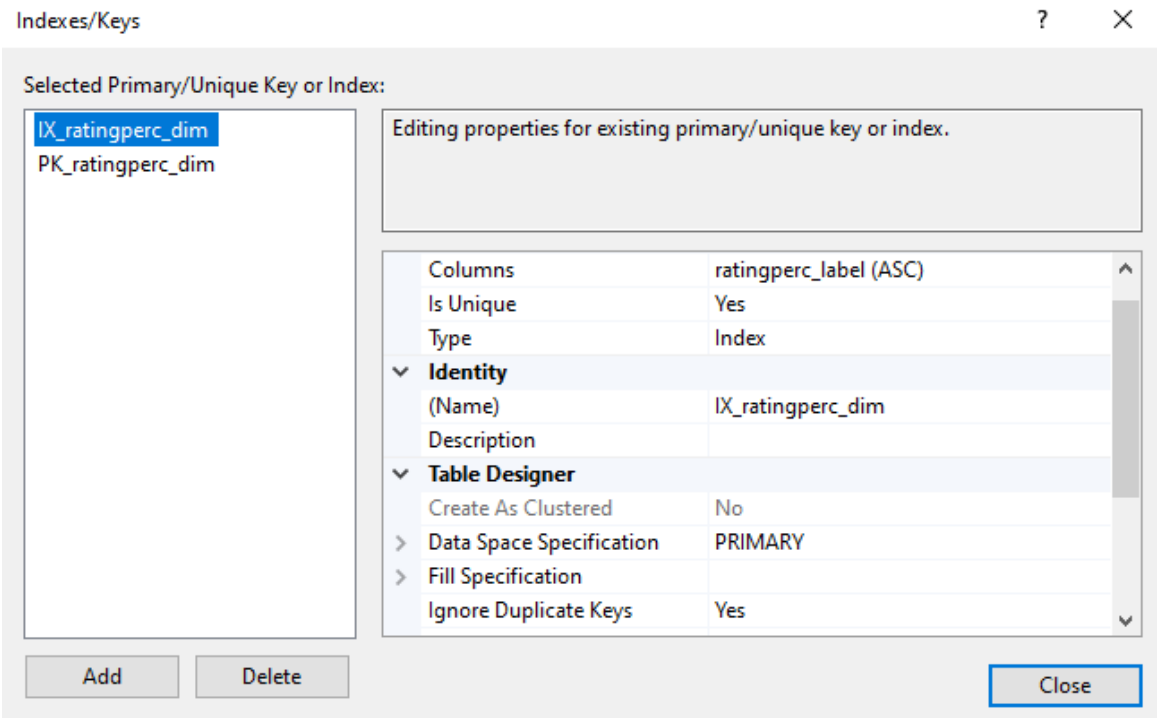
- ratingperc_dim



Data Loading Procedure 37

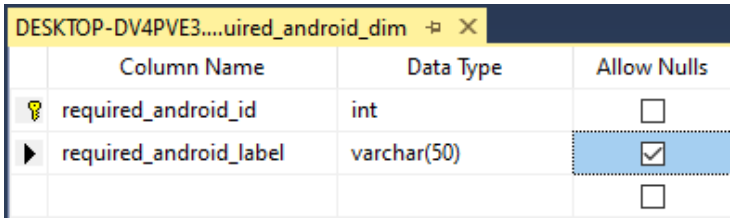


Data Loading Procedure 38

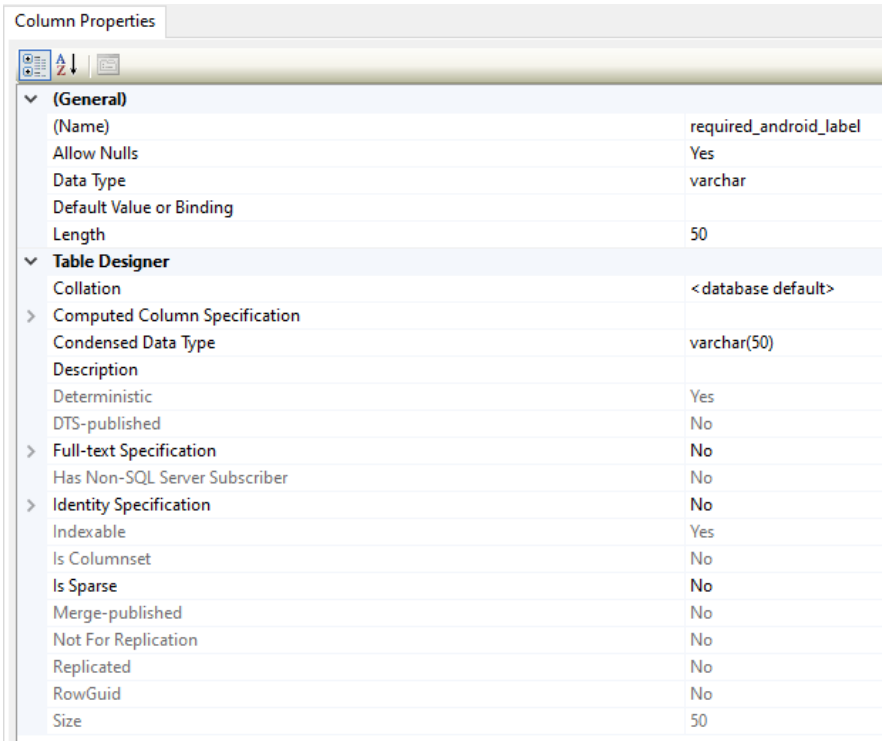


Data Loading Procedure 39

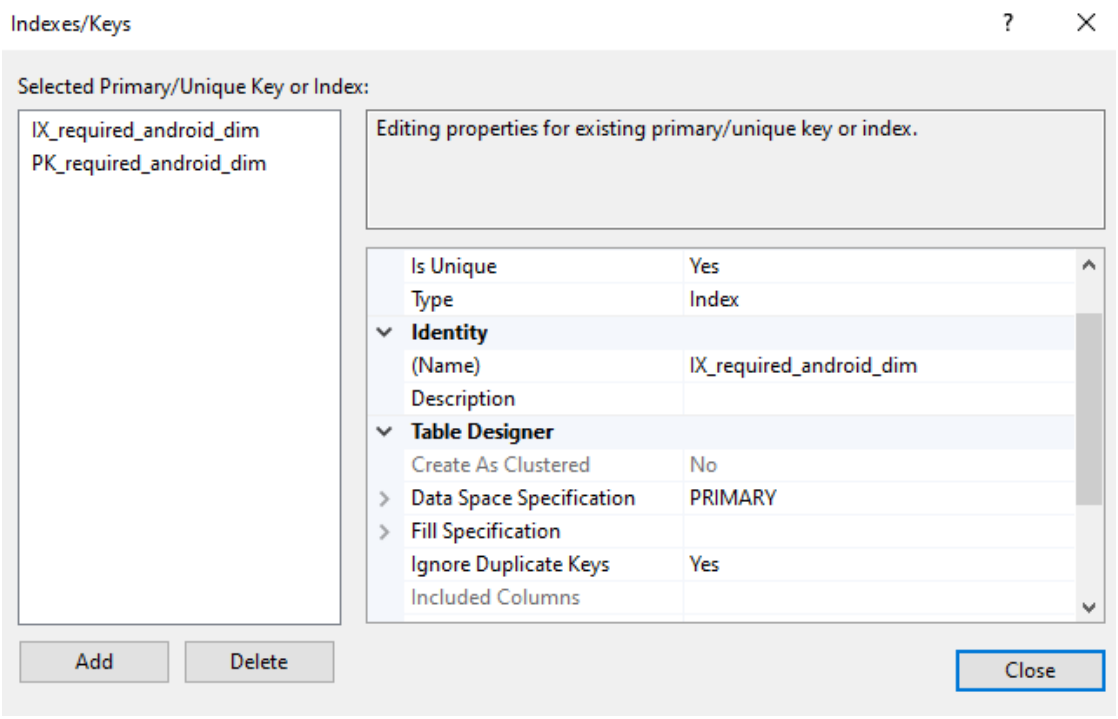
- required_android_dim



Data Loading Procedure 40



Data Loading Procedure 41

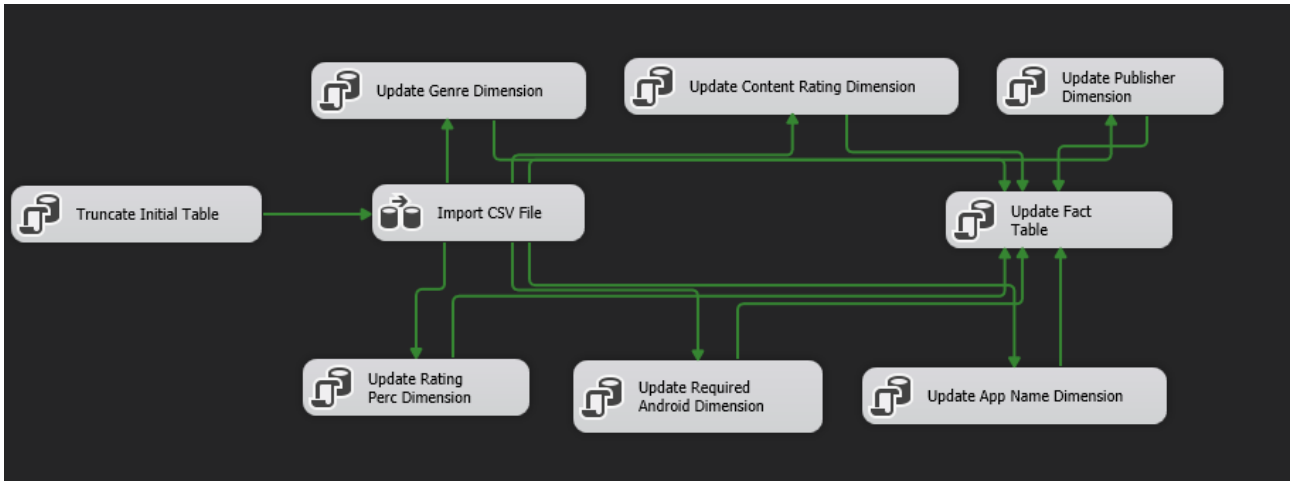


Data Loading Procedure 42

Afterwards, we create an Execute SQL Task for each dimension table in the Visual Studio, that updates the unpivot view, which has already the data from the imported flat file initially imported and settled into the staging table, each dimension retrieves its data from the staging_unpivot view and gets updated if a new observation has been recorded. Therefore, in order to fill the labels of each dimension we select the distinct values from the staging_unpivot (example given for the appname dimension):

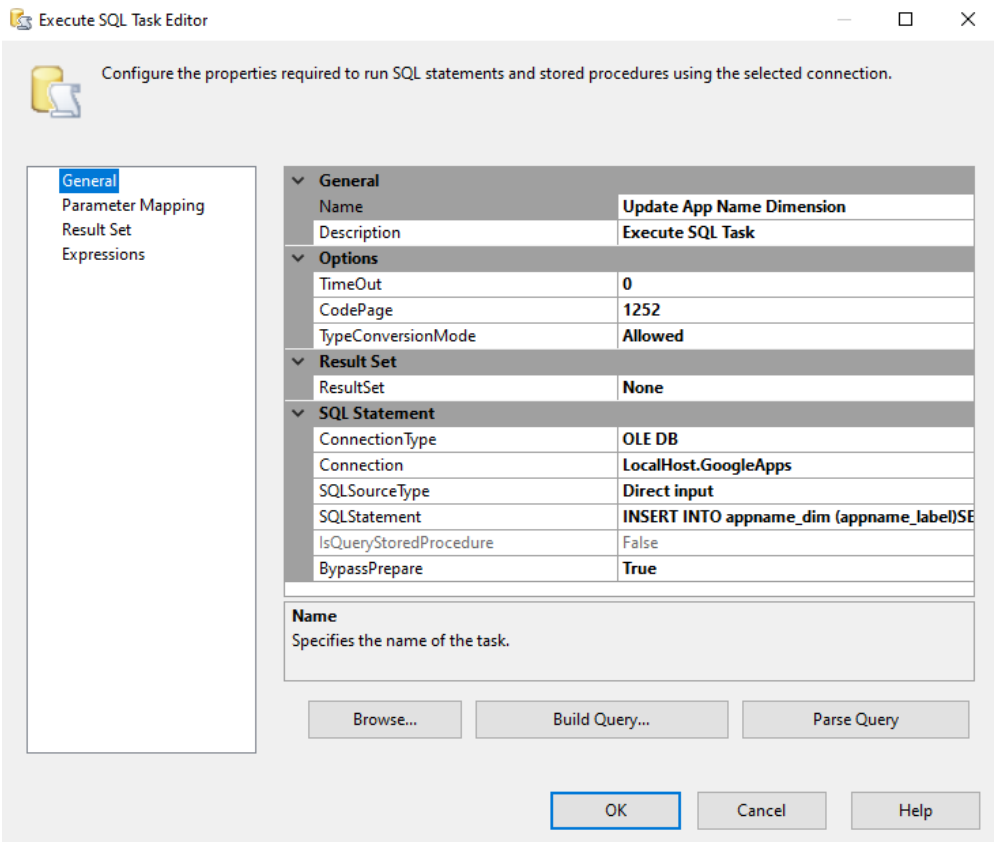
```
1.  INSERT INTO appname_dim (appname_label)
2.  SELECT DISTINCT [App Name] FROM Staging_Unpivot
```

The above procedure will be further explained below.

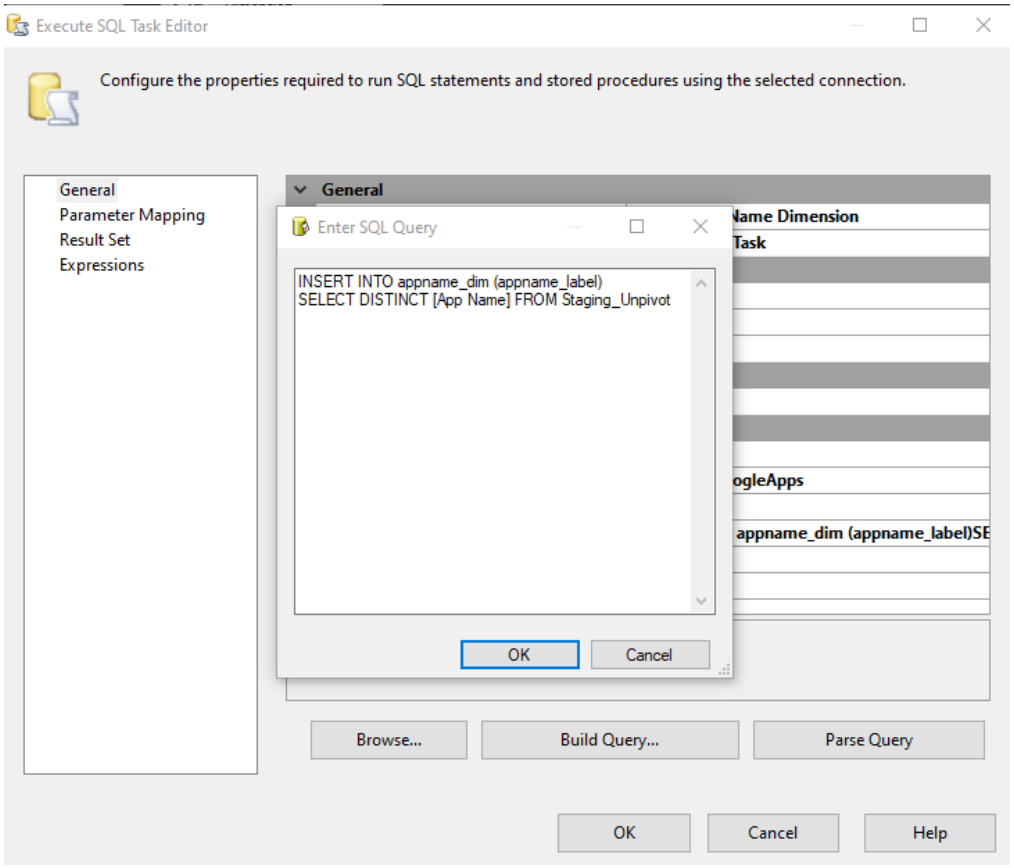


Data Loading Procedure 43

- appname_dim

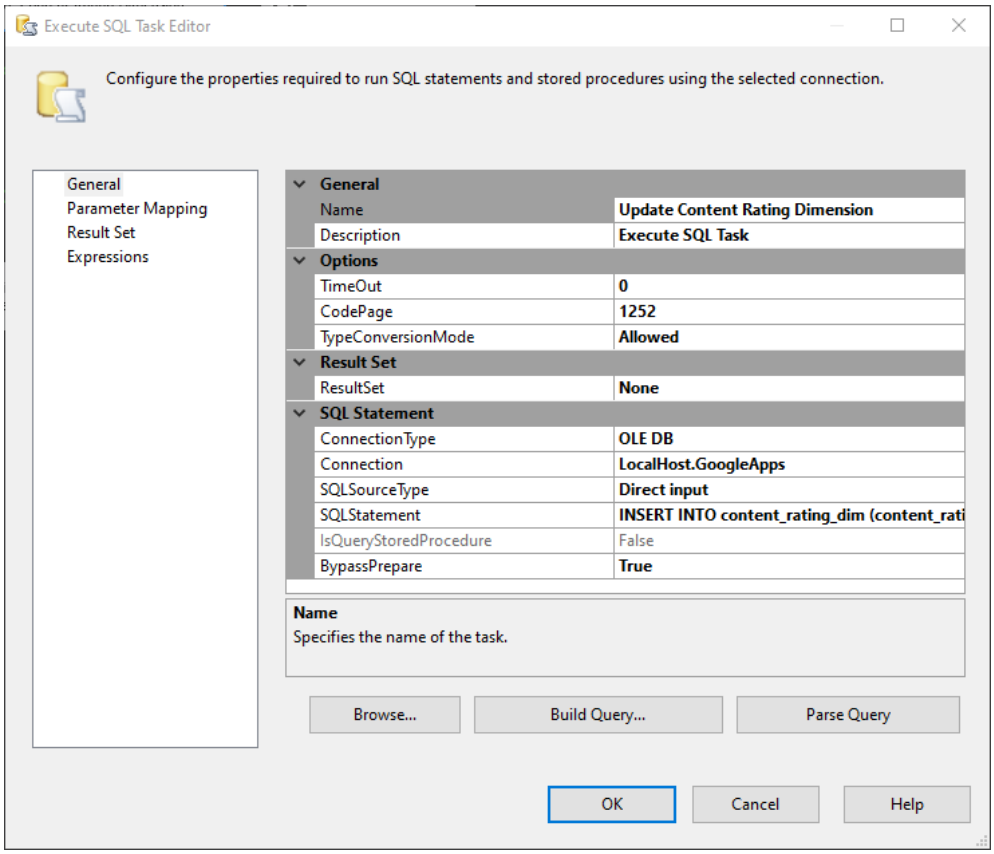


Data Loading Procedure 44

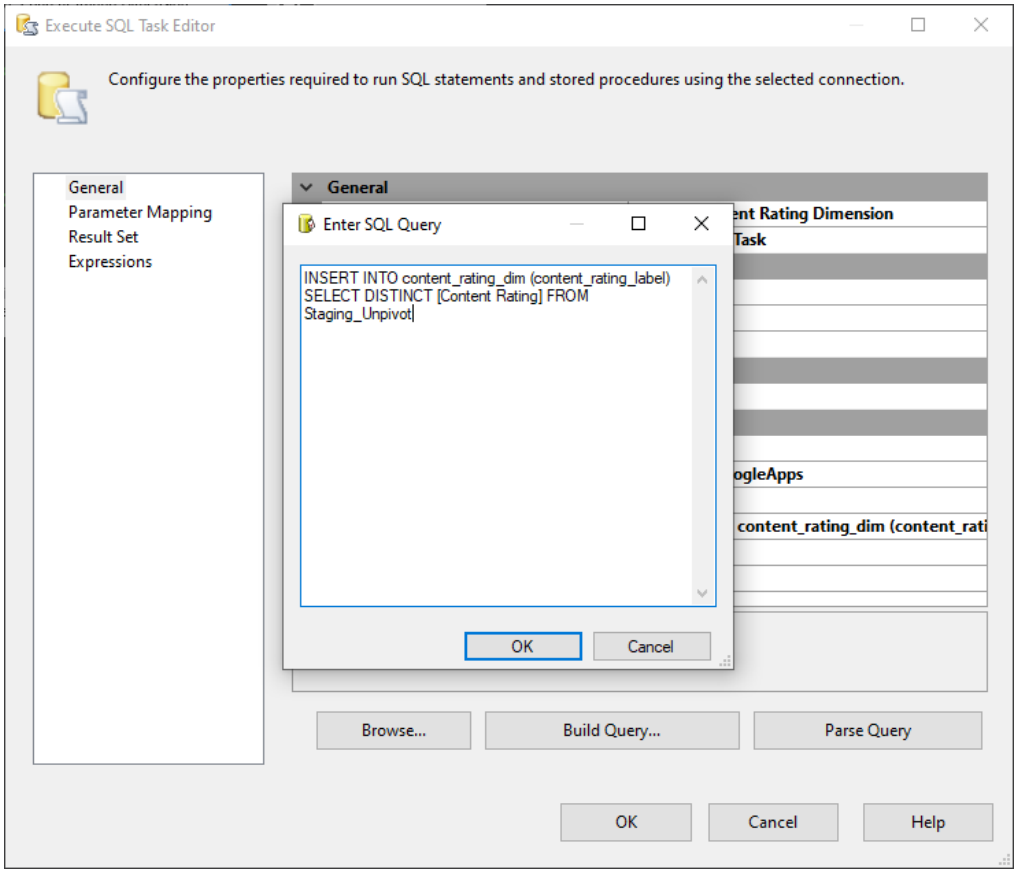


Data Loading Procedure 45

- content_rating_dim

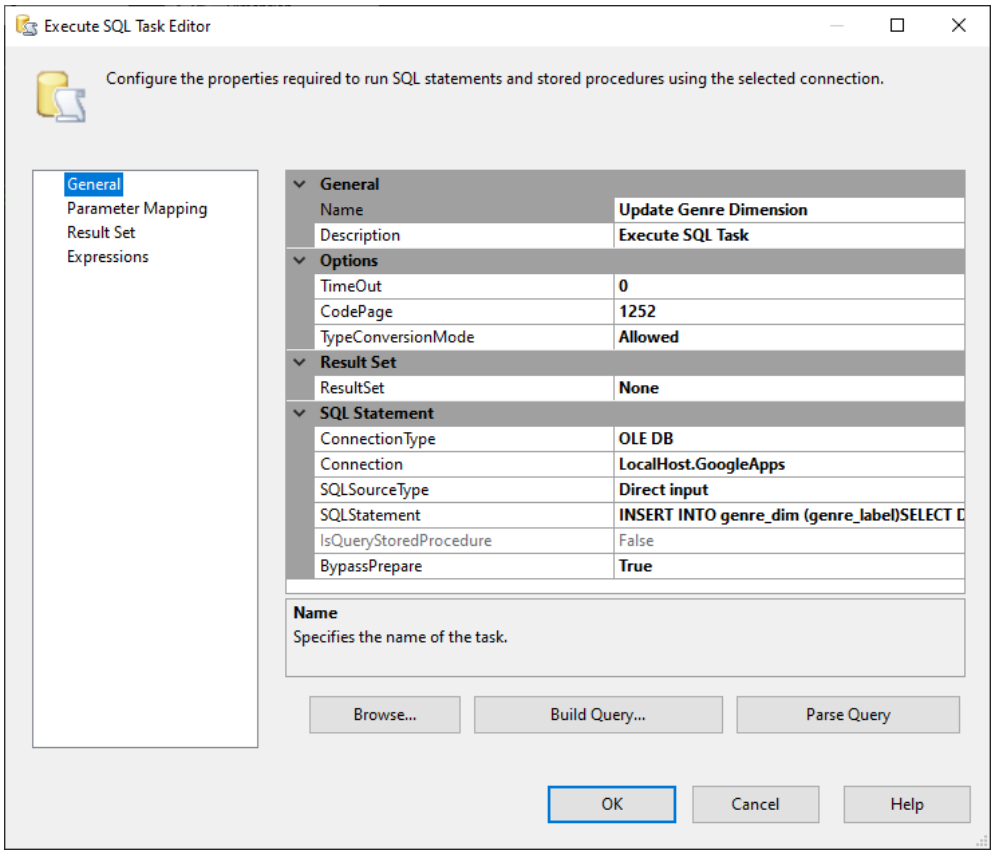


Data Loading Procedure 46

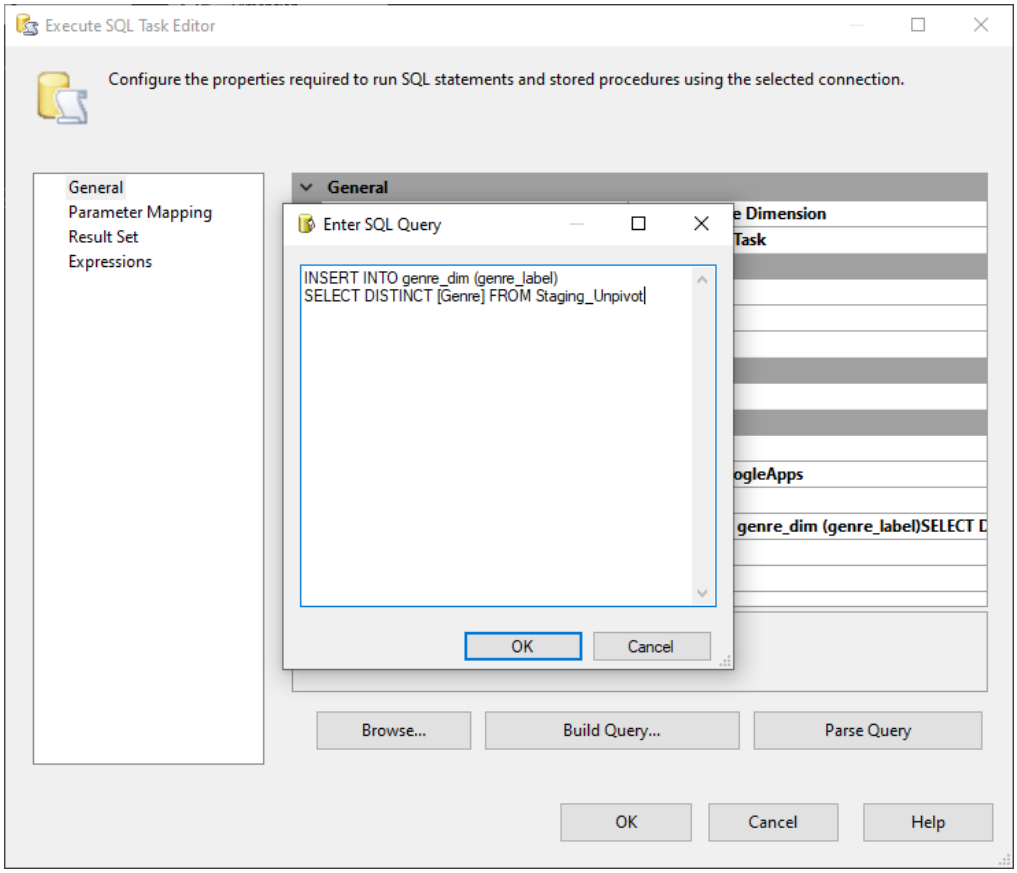


Data Loading Procedure 47

- genre_dim

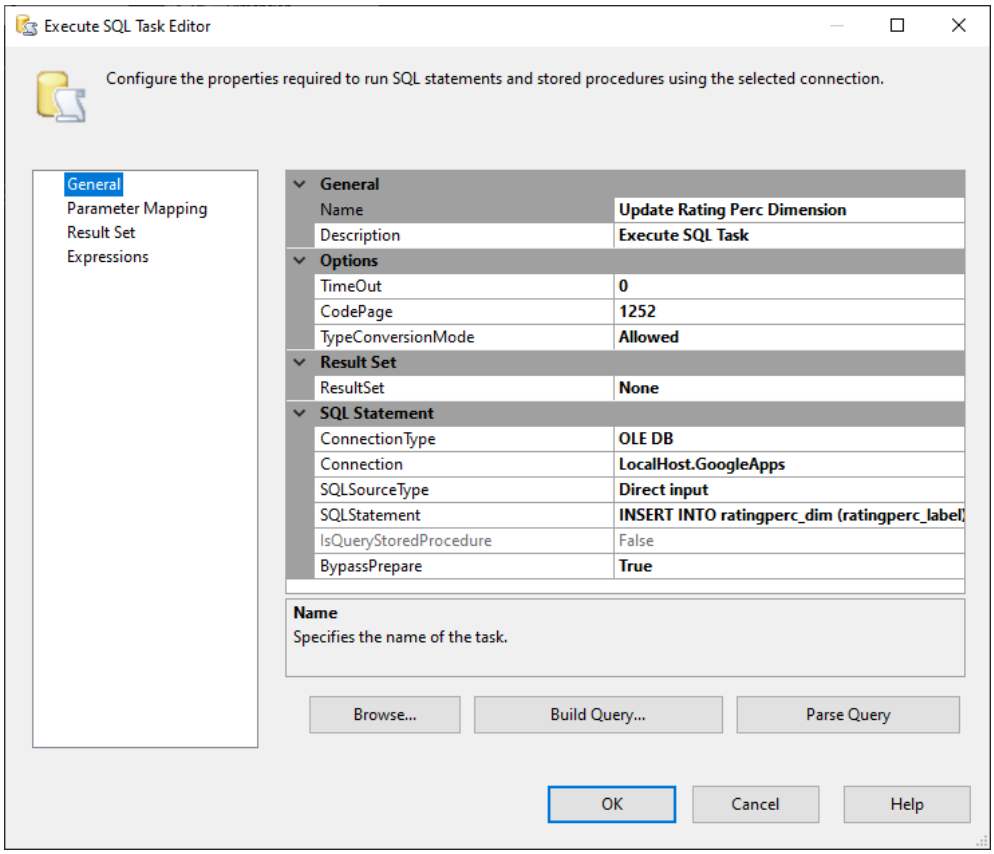


Data Loading Procedure 48

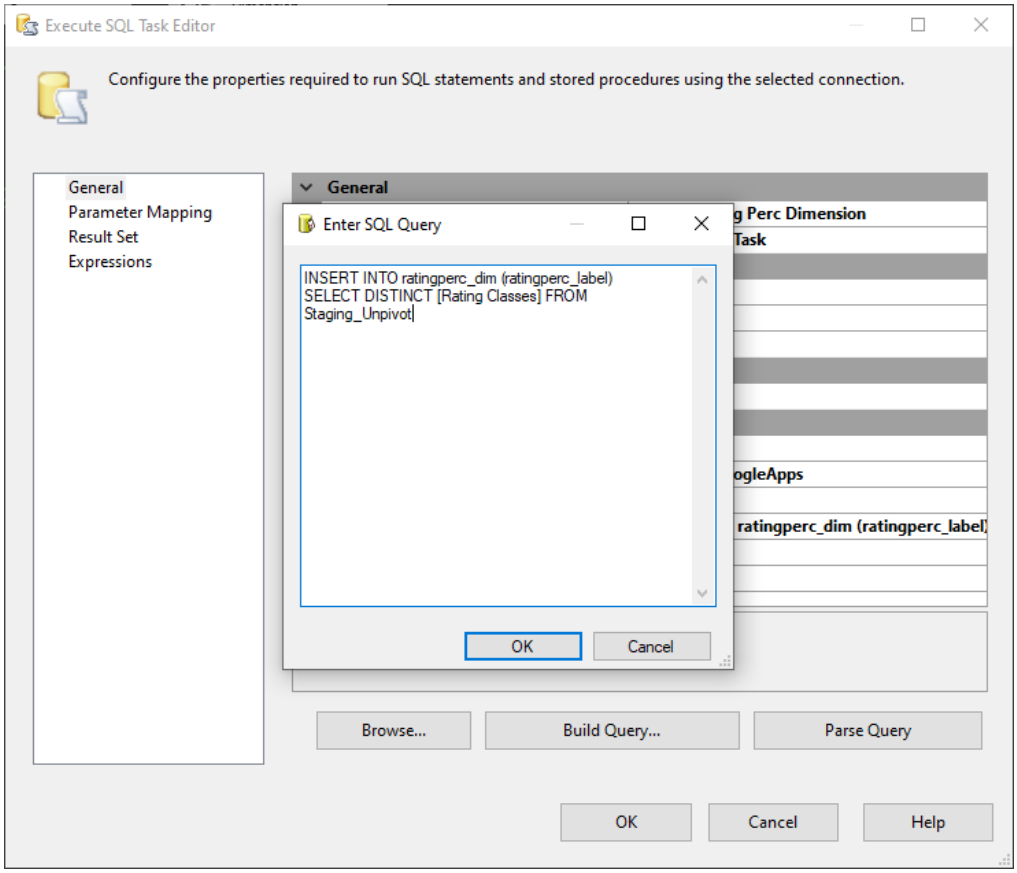


Data Loading Procedure 49

- rating_perc_dim

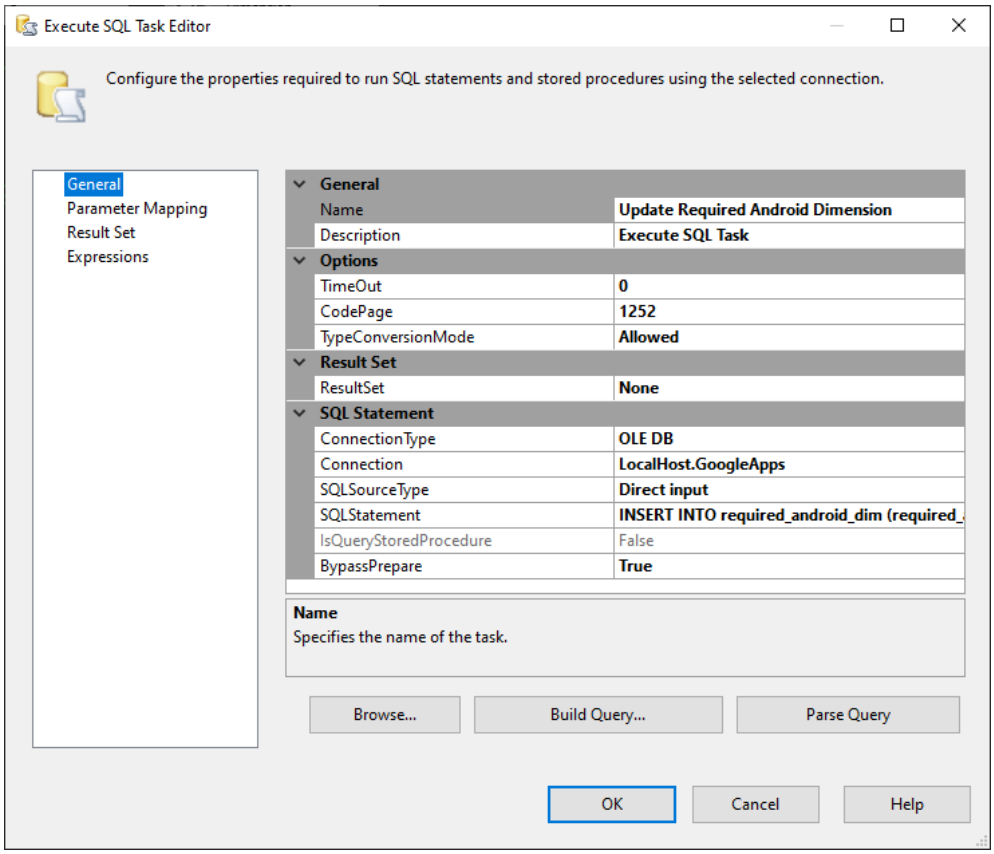


Data Loading Procedure 50

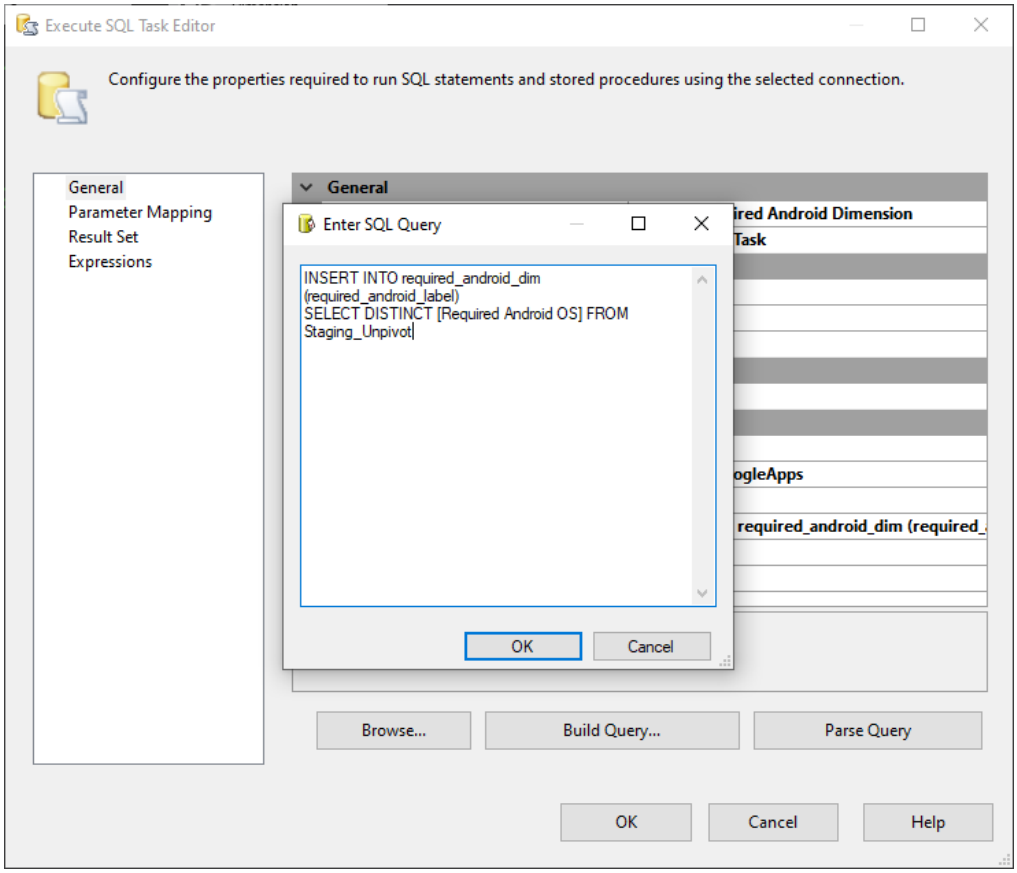


Data Loading Procedure 51

- required_android_dim

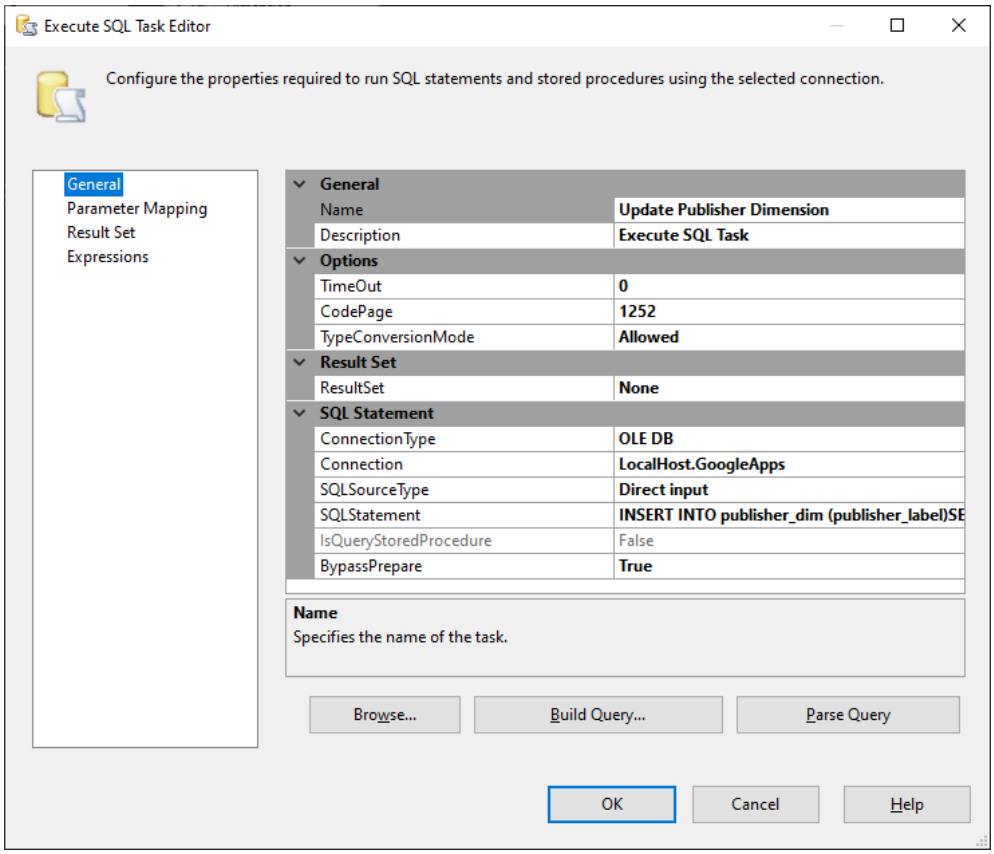


Data Loading Procedure 52

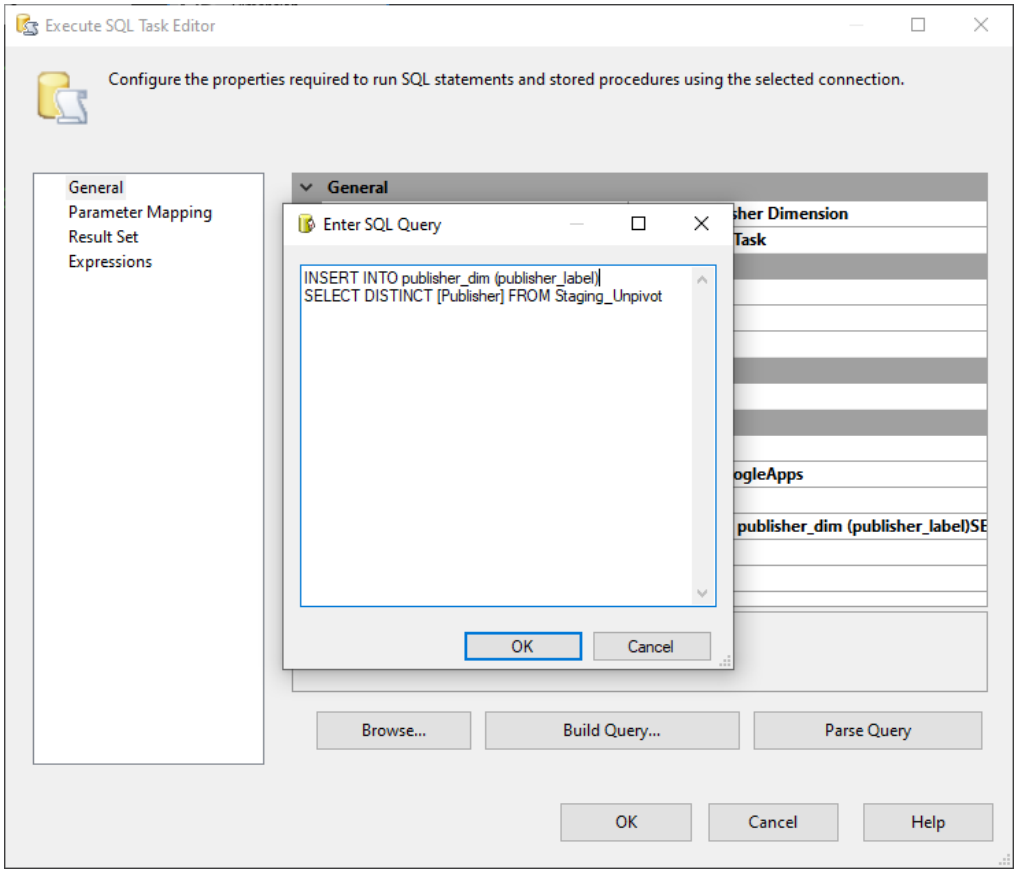


Data Loading Procedure 53

- publisher_dim



Data Loading Procedure 54



Data Loading Procedure 55

Finally, all the created dimensions have the following result's format (eg. genre_dim) and will be updated only in case that a new entry (different from the existing ones) will be inserted to the view.

	genre_id	genre_label
1	1	Social
2	2	Communication
3	3	Personalization
4	4	Health & Fitness
5	5	Simulation
6	6	Music
7	7	Maps & Navigation
8	8	Auto & Vehicles
9	9	Board
10	10	Platform
11	11	Business
12	12	Adventure
13	13	Art & Design
14	14	Photography
15	15	Shopping
16	16	Word
17	17	Sports
18	18	Travel & Local
19	19	Tools
20	20	Lifestyle
21	21	Video Players & ...
22	22	Entertainment
23	23	Books & Referen...
24	24	Parenting
25	25	Strategy
26	26	Educational
27	27	Casino
28	28	Music & Audio
29	29	Card
30	30	Events
31	31	News & Magazines
32	32	Puzzle
33	33	Role Playing
34	34	Trivia
35	35	Arcade
36	36	House & Home
37	37	Dating

Data Loading Procedure 56

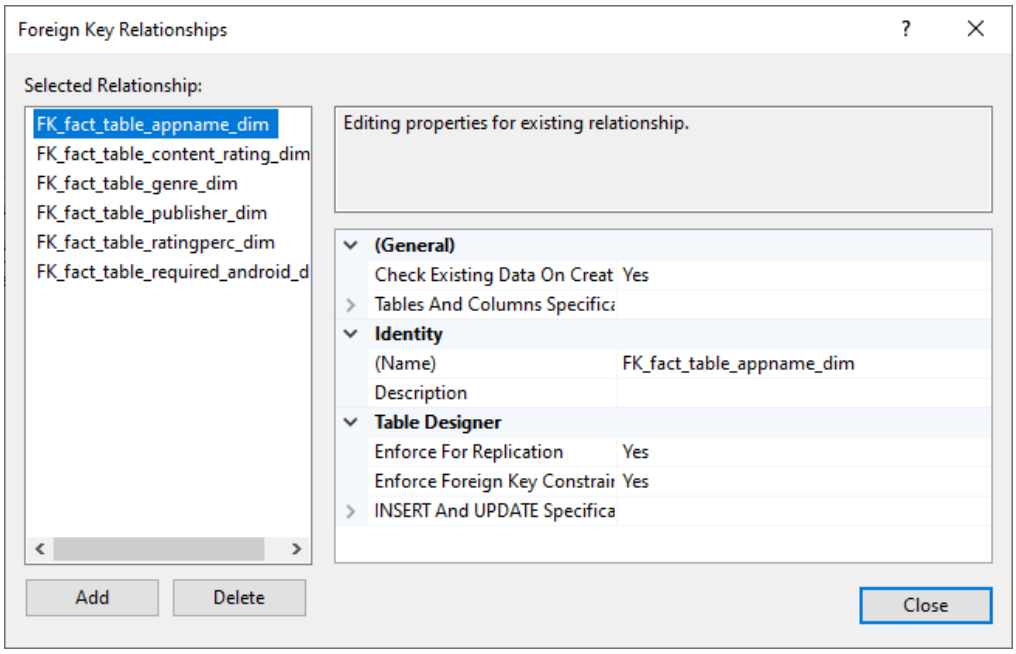
Fact Table Creation

The fact table was filled with values from all dimensions through the Visual Studio ETL procedures as another Execute SQL Task component is created. The query written to update the fact table through its dimensions uses the dimension's ids as the fact table's foreign keys is presented below.

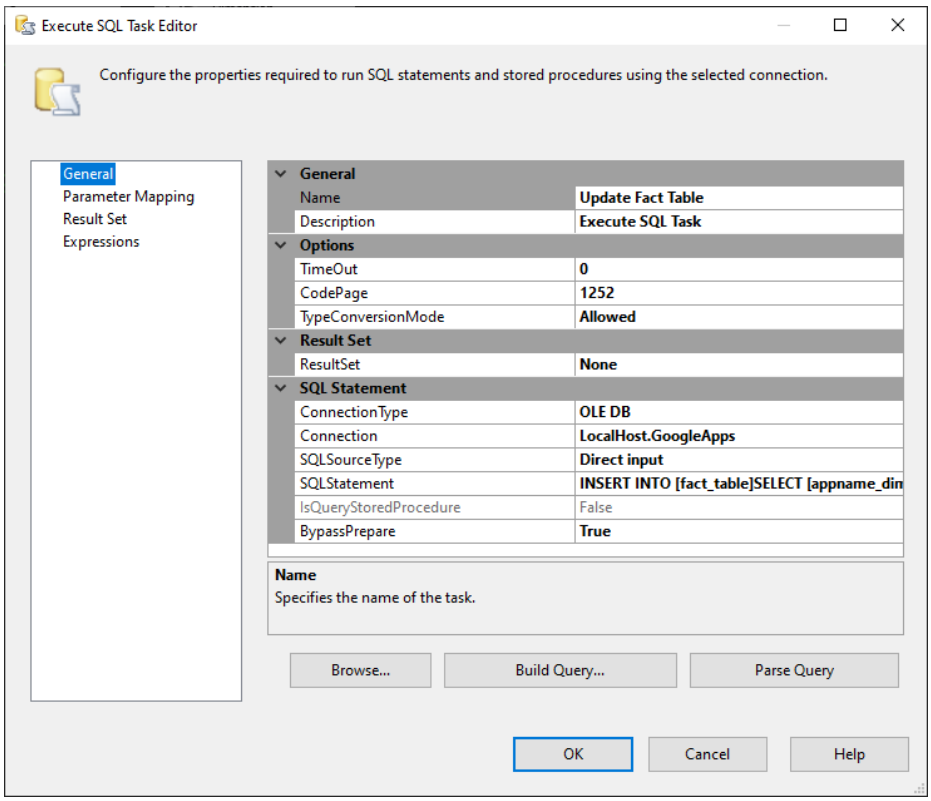
DESKTOP-DV4PVE3.G...s - dbo.fact_table

Column Name	Data Type	Allow Nulls
[App Name]	int	☐
Publisher	int	☐
Genre	int	☐
[Rating Class]	int	☐
[Required Android OS]	int	☐
[Content Rating]	int	☐
[App Size]	float	☑
Price	float	☑
isFree	bit	☑
Rating	float	☑
[Total Reviews]	int	☑
[Total Installation]	int	☑
[In App Purchases MIN]	float	☑
[In App Purchases MAX]	float	☑
[Rating Perc]	float	☑

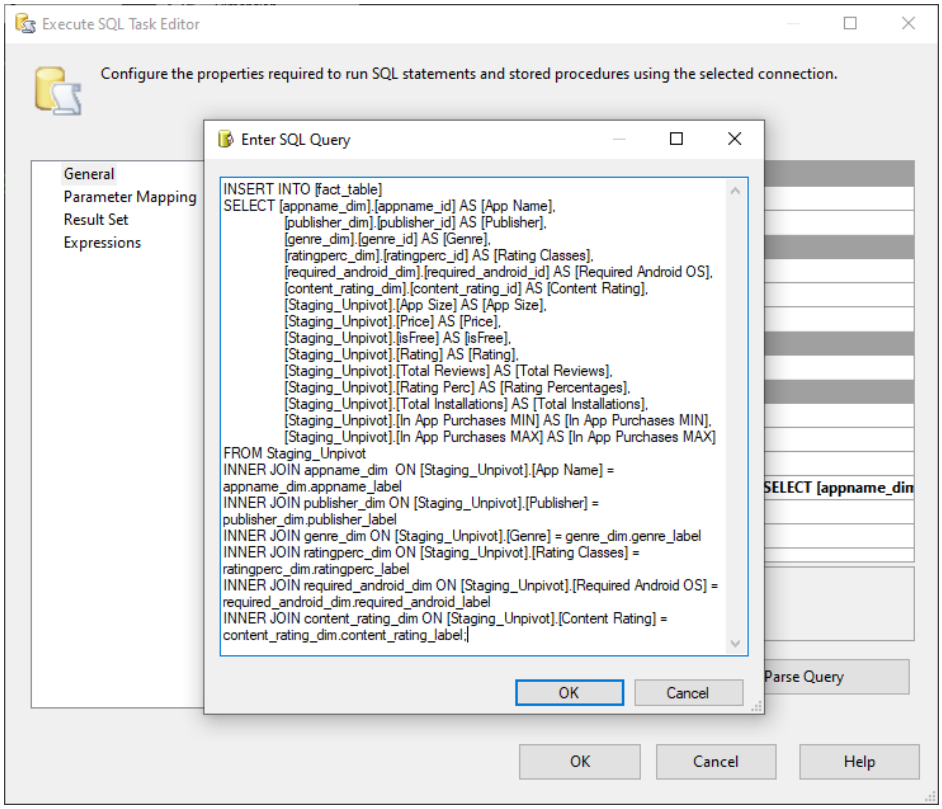
Data Loading Procedure 57



Data Loading Procedure 58



Data Loading Procedure 59



Data Loading Procedure 60

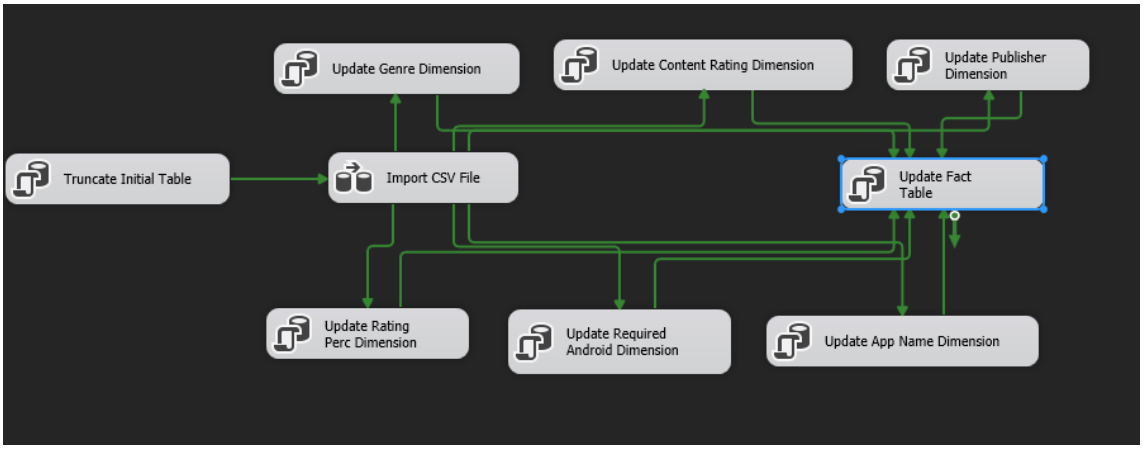
The query that used to fill the table is the following :

```
INSERT INTO [fact_table]
SELECT [appname_dim].[appname_id] AS [App Name],
    [publisher_dim].[publisher_id] AS [Publisher],
    [genre_dim].[genre_id] AS [Genre],
    [ratingperc_dim].[ratingperc_id] AS [Rating Classes],
    [required_android_dim].[required_android_id] AS [Required Android OS],
    [content_rating_dim].[content_rating_id] AS [Content Rating],
    [Staging_Unpivot].[App Size] AS [App Size],
    [Staging_Unpivot].[Price] AS [Price],
    [Staging_Unpivot].[isFree] AS [isFree],
    [Staging_Unpivot].[Rating] AS [Rating],
    [Staging_Unpivot].[Total Reviews] AS [Total Reviews],
    [Staging_Unpivot].[Rating Perc] AS [Rating Percentages],
    [Staging_Unpivot].[Total Installations] AS [Total Installations],
    [Staging_Unpivot].[In App Purchases MIN] AS [In App Purchases MIN],
    [Staging_Unpivot].[In App Purchases MAX] AS [In App Purchases MAX]
FROM Staging_Unpivot
INNER JOIN appname_dim  ON [Staging_Unpivot].[App Name] = appname_dim.appname_label
INNER JOIN publisher_dim ON [Staging_Unpivot].[Publisher] = publisher_dim.publisher_label
INNER JOIN genre_dim ON [Staging_Unpivot].[Genre] = genre_dim.genre_label
INNER JOIN ratingperc_dim ON [Staging_Unpivot].[Rating Classes] = ratingperc_dim.ratingperc_l
abel
INNER JOIN required_android_dim ON [Staging_Unpivot].[Required Android OS] = required_android
_dim.required_android_label
INNER JOIN content_rating_dim ON [Staging_Unpivot].[Content Rating] = content_rating_dim.cont
ent_rating_label;
```

A results sample of the fact table is presented below :

	App Name	Publisher	Genre	Rating Class	Required Android OS	Content Rating	App Size	Price	isFree	Rating	Total Reviews	Total Installation	In App Purchases MIN	In App Purchases MA
103	104406	59671	23	1	7	5	82	0	1	4.7	442	0	100000	NULL
104	104406	59671	23	2	7	5	82	0	1	4.7	442	1	100000	NULL
105	104406	59671	23	3	7	5	82	0	1	4.7	442	88	100000	NULL
106	104406	59671	23	4	7	5	82	0	1	4.7	442	6	100000	NULL
107	104406	59671	23	5	7	5	82	0	1	4.7	442	3	100000	NULL
108	104407	56585	40	1	9	5	12	2.65	0	4.5	12	8	100	NULL
109	104407	56585	40	2	9	5	12	2.65	0	4.5	12	8	100	NULL
110	104407	56585	40	3	9	5	12	2.65	0	4.5	12	75	100	NULL
111	104407	56585	40	4	9	5	12	2.65	0	4.5	12	8	100	NULL
112	104407	56585	40	5	9	5	12	2.65	0	4.5	12	0	100	NULL
113	104408	82946	28	1	7	5	32	0	1	4.7	191	0	10000	0.82
114	104408	82946	28	2	7	5	32	0	1	4.7	191	5	10000	0.82
115	104408	82946	28	3	7	5	32	0	1	4.7	191	85	10000	0.82
116	104408	82946	28	4	7	5	32	0	1	4.7	191	6	10000	0.82
117	104408	82946	28	5	7	5	32	0	1	4.7	191	2	10000	0.82
118	104409	79350	19	1	5	5	49	0	1	3.7	15817	5	1000000	NULL
119	104409	79350	19	2	5	5	49	0	1	3.7	15817	8	1000000	NULL
120	104409	79350	19	3	5	5	49	0	1	3.7	15817	48	1000000	NULL

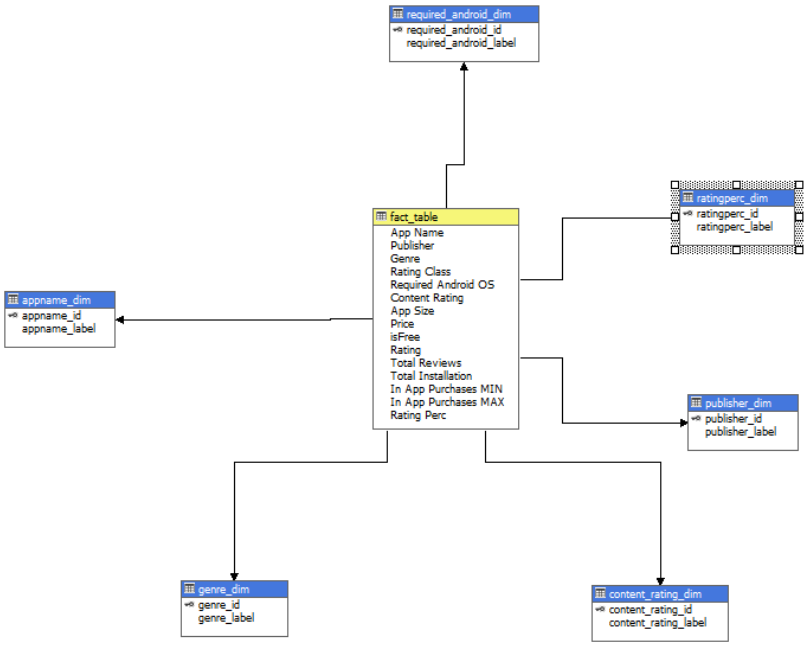
Data Loading Procedure 61



Data Loading Procedure 62

Database Schema

For this assignment we selected to develop a Star schema, since is a mature modelling approach widely adopted by relational data warehouses and covers our dataset and our forthcoming business case analysis needs. Therefore, the Star schema is presented below.



Data Loading Procedure 63: Star Schema

Multidimensional Model (MS SSAS)

A multidimensional model is composed of cubes and dimensions that can be annotated and extended to support complex query constructions. Business Intelligence developers create cubes to support fast response times, and to provide a single data source for business reporting. Therefore, it is an essential step toward our reporting procedure.

Build the Data Cube

In this section, we will explain the steps followed regarding the data cube construction. After completing the construction of the Data Warehouse, we designed the cube in Visual Studio using the Multidimensional Analysis Service and Data Mining Tool. The steps that are needed for the construction of the data cube are presented below.

Creation of the Multidimensional Analysis Service and Data Mining Project

The first step is to create a new project in Visual Studio. To be more precise, we need to create a new Multidimensional Analysis Service and Data Mining Project. As you can see below, we create the GoogleAppsAnalysis project as a new Multidimensional Analysis Service and Data Mining Project.

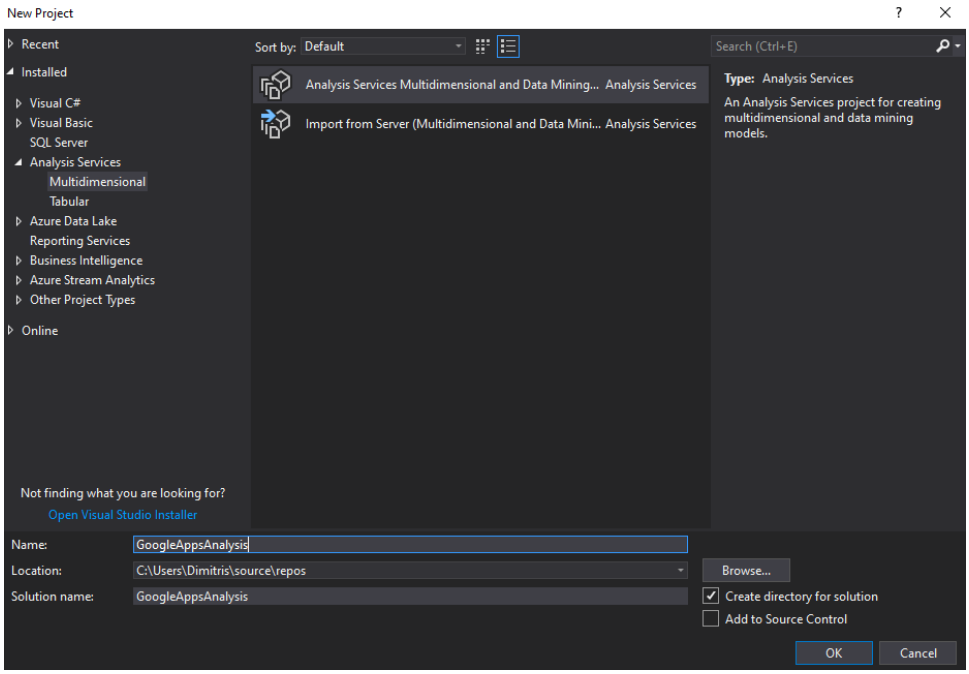


Figure 11 : Creation of the Multidimensional Analysis Service and Data Mining Project

Connection of the Data Warehouse with the Multidimensional Model

Now, we need to retrieve the data stored in the data warehouse, through the Visual Studio, and following the Data Source Wizard. In order to do that, we follow the steps below:

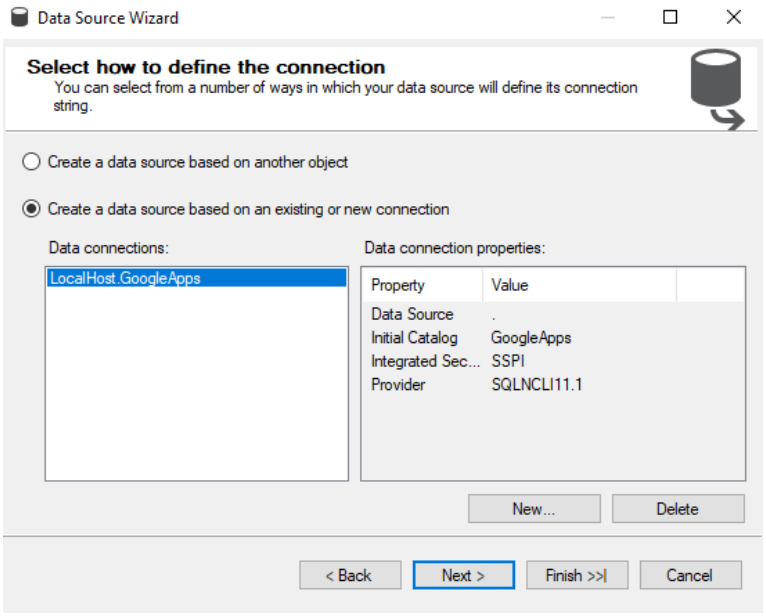


Figure 12: Connection of the project with the GoogleApps Database

Creation of the Data Cube - Definition of the measures of the GoogleApps Database fact table

For the construction of the data cube, we select the data source setup wizard of Visual Studio, and then we choose only the fact table of the database.

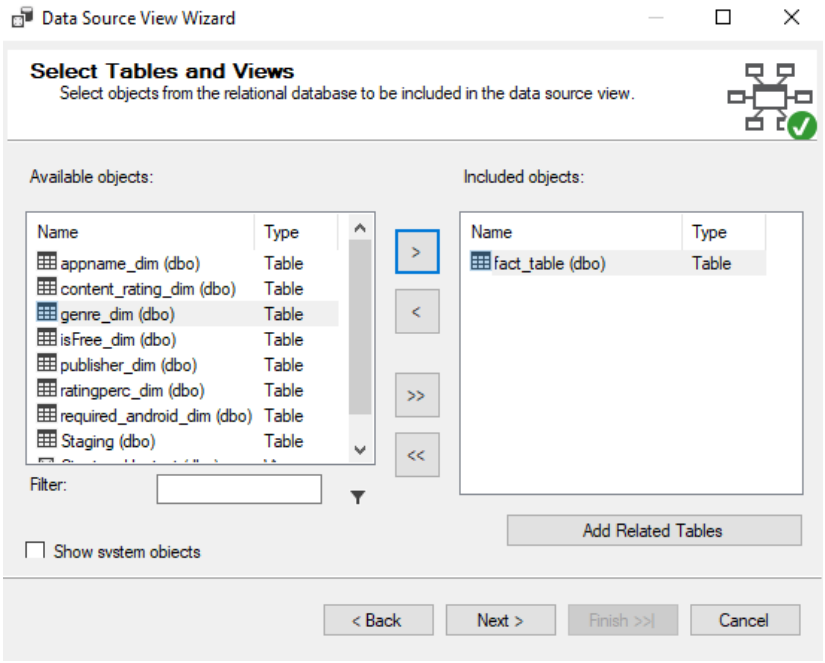


Figure 13: Definition of the measures (1/2)

Afterwards, we press the add related tables, and Visual Studio finds all the related tables to the database fact table.

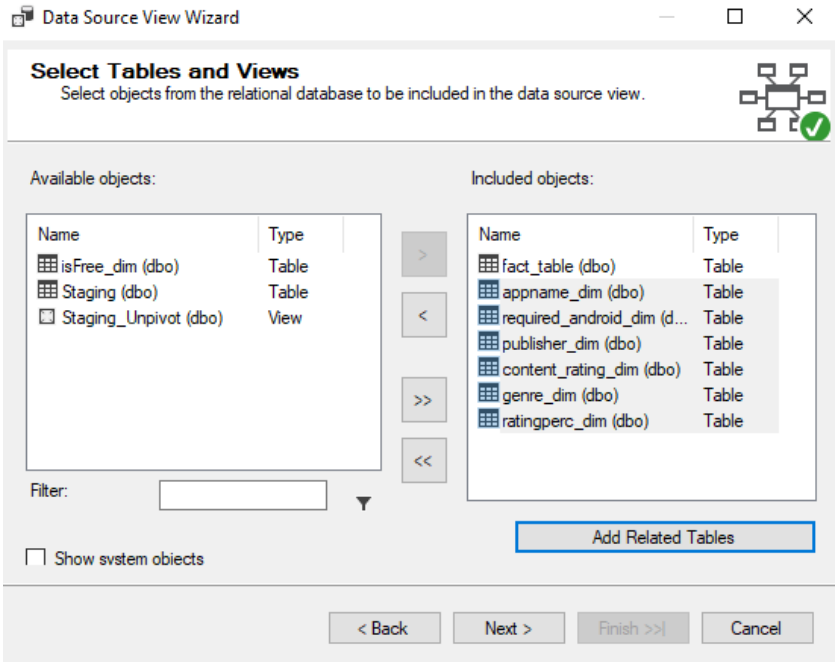


Figure 14: Definition of the measures (2/2)

Configuring the Data Cubes' Dimension Tables

A very important process that has to be done is the association between the dimension table id with its label in order to showcase the reports more efficiently. Therefore, as you can see in the following sample picture, we add to the dimension table attribute the label among the id to be able to perform this association.

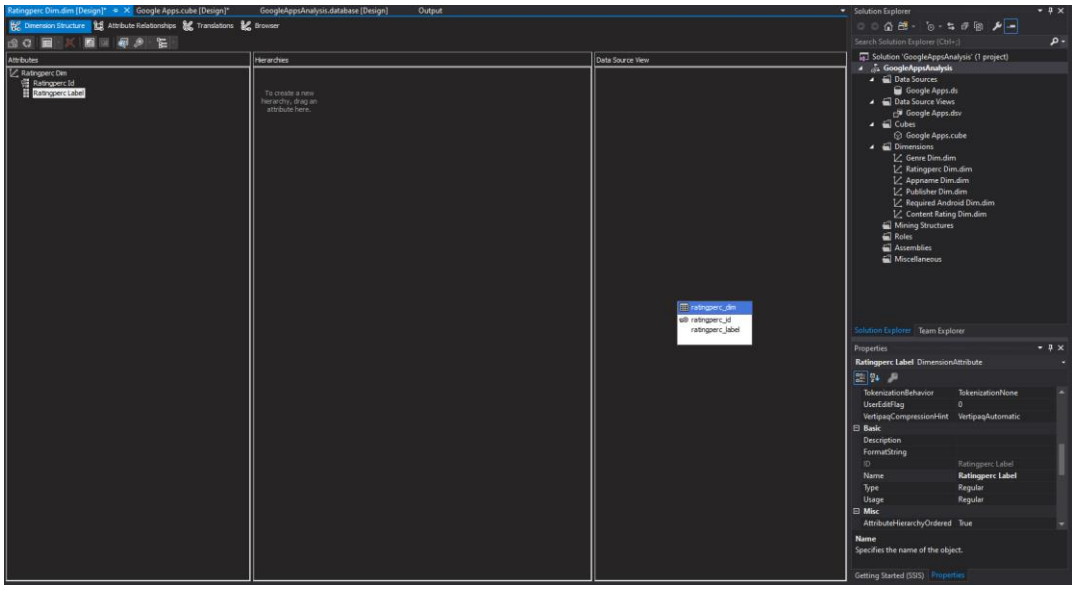


Figure 15: Configuring the Data Cubes' Dimension Tables (1/2)

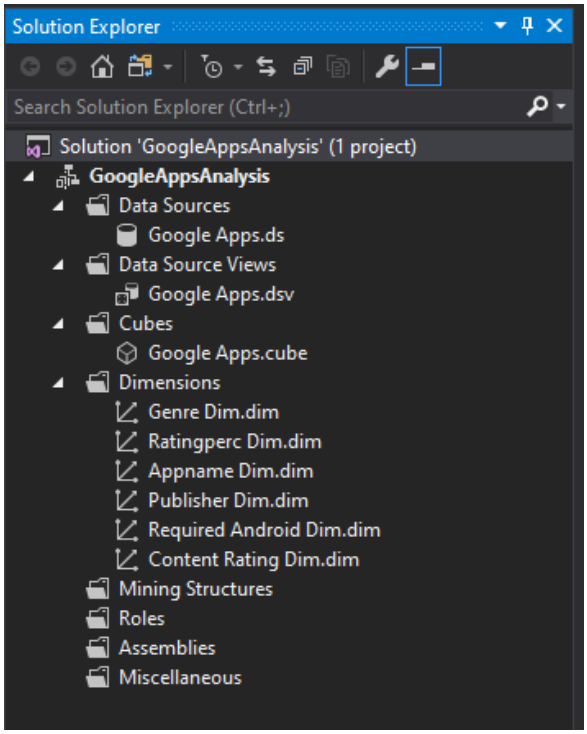


Figure 16: Configuring the Data Cubes' Dimension Tables (2/2)

Deployment and Process

Since our dimensions are properly defined, we can move on to the cube's deployment and process. To be more precise, in order to do that, we select the 'Process' option, and then we can see the schema through Visual Studio. The following images depict the aforementioned process:

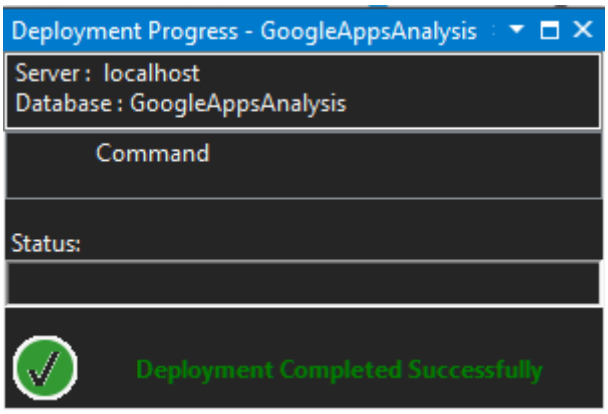


Figure 17: Deployment and Process (1/3)

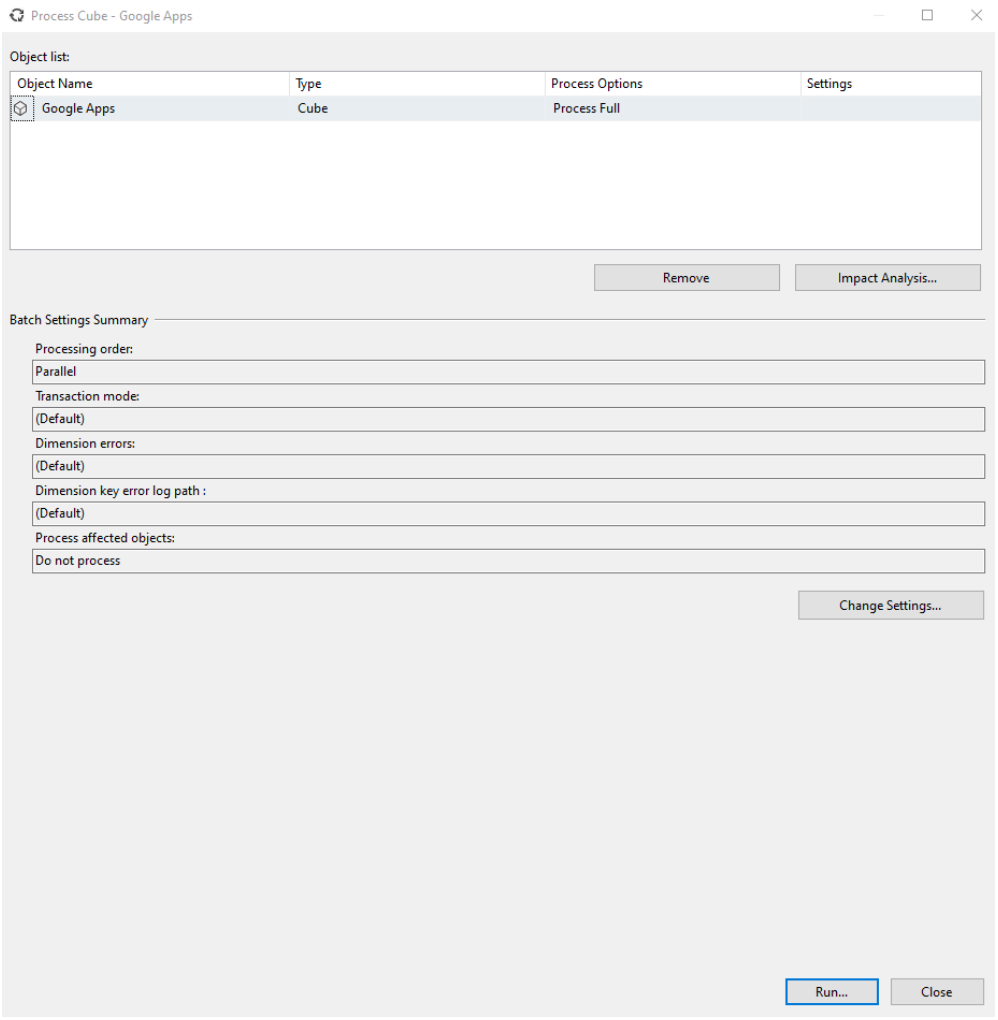


Figure 18: Deployment and Process (2/3)

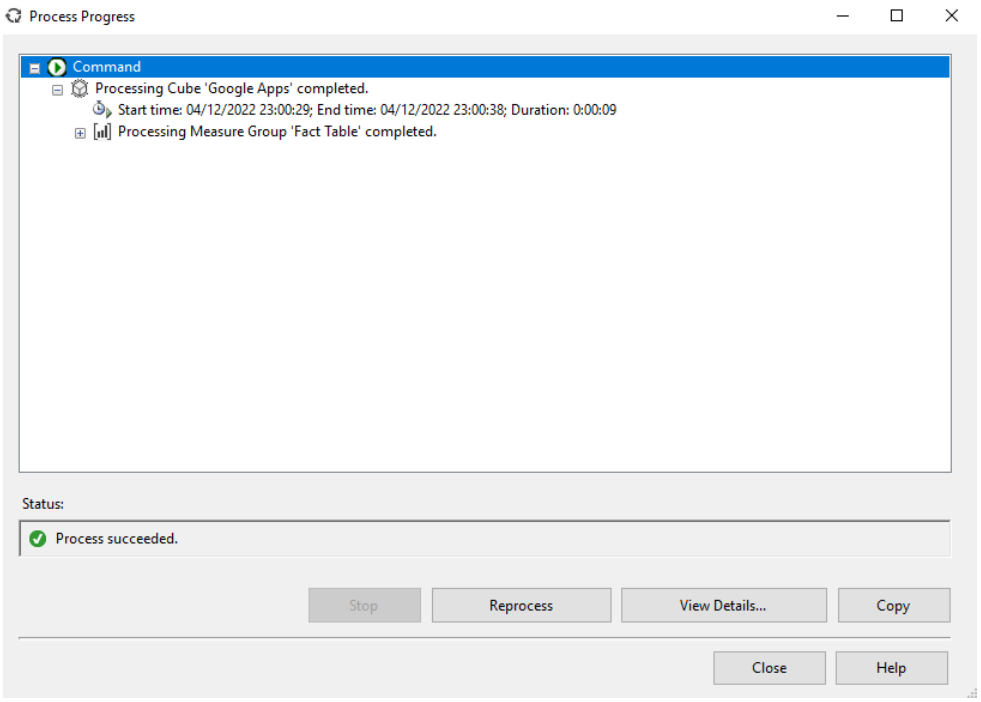


Figure 19: Deployment and Process (3/3)

Data Cube's Schema Output

After the successful completion of the deployment and the process of the data cube we can see the produced fact table. Through the deployment, we are able to make adjustment to our cube and supervise how we construct the cube to fulfill our analysis needs.

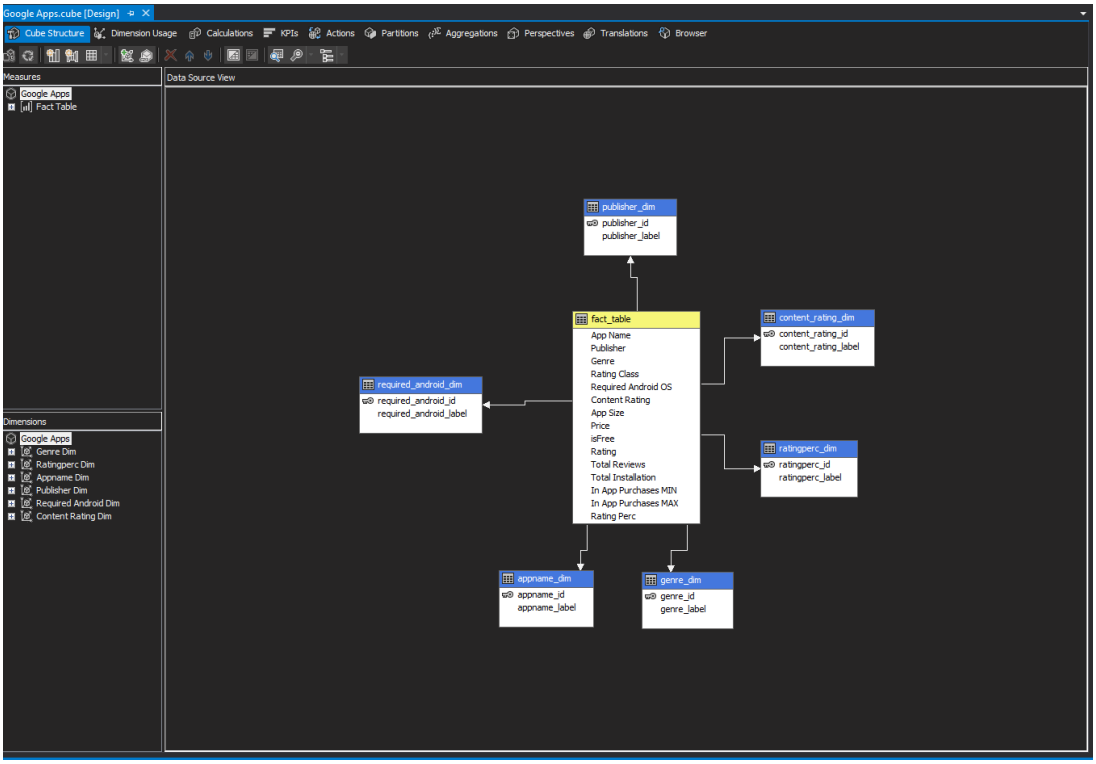


Figure 20: Data Cube's Schema

OLAP Operations in SSAS

To begin with, OLAP (Online Analytical Processing Server) has 4 main operations, which are the following:

- **Drill down**: The drill down operation, the less detailed data is converted into highly detailed data. Drill down can be done by moving down (zoom in) in the data hierarchy between dimensions that are hierarchal related. Therefore, when Drill down is performed one or more dimensions have to be added to the current cube's view.

- **Roll up**: The opposite of the drill down operation. Roll up performs aggregation on the OLAP cube. It can be done by, climbing up in the data hierarchy or removing an existing dimension. Therefore, when Roll up is performed one or more dimensions have to be removed.

- **Slice**: Selects a single dimension from the cube which results in a new sub-cube creation.

- **Dice**: Selects a sub-cube from the OLAP cube by selecting two or more dimensions.

OLAP Calculated Measures

An OLAP (Online Analytical Processing Server) Calculation is a Multidimensional Expressions (MDX) expression or script that is used to define a calculated member in a cube in Visual Studio and more specifically in Multidimensional Analysis Service and Data Mining Project.

Calculations let you add objects that are defined not by the data of the existing cube, but by expressions that can reference other parts of the cube (other measures). Calculations let you extend the capabilities of a cube, adding flexibility and power to business intelligence applications.

Therefore, we choose to create two calculated measures, the Avg_Price, to calculate the average price of the paid apps, which later be used for the reports to the stakeholders and the Installation_Profit, which multiply the sum of the total installation of all the paid apps multiplied with the average price of the apps of a specific genre, this way it will help our reports regarding which of the pre-defined by the stakeholder seven app's genre will eventually be chosen.

Finally, more calculated measures will be created to the PowerBI app, since we found it out more useful for our analysis reports.

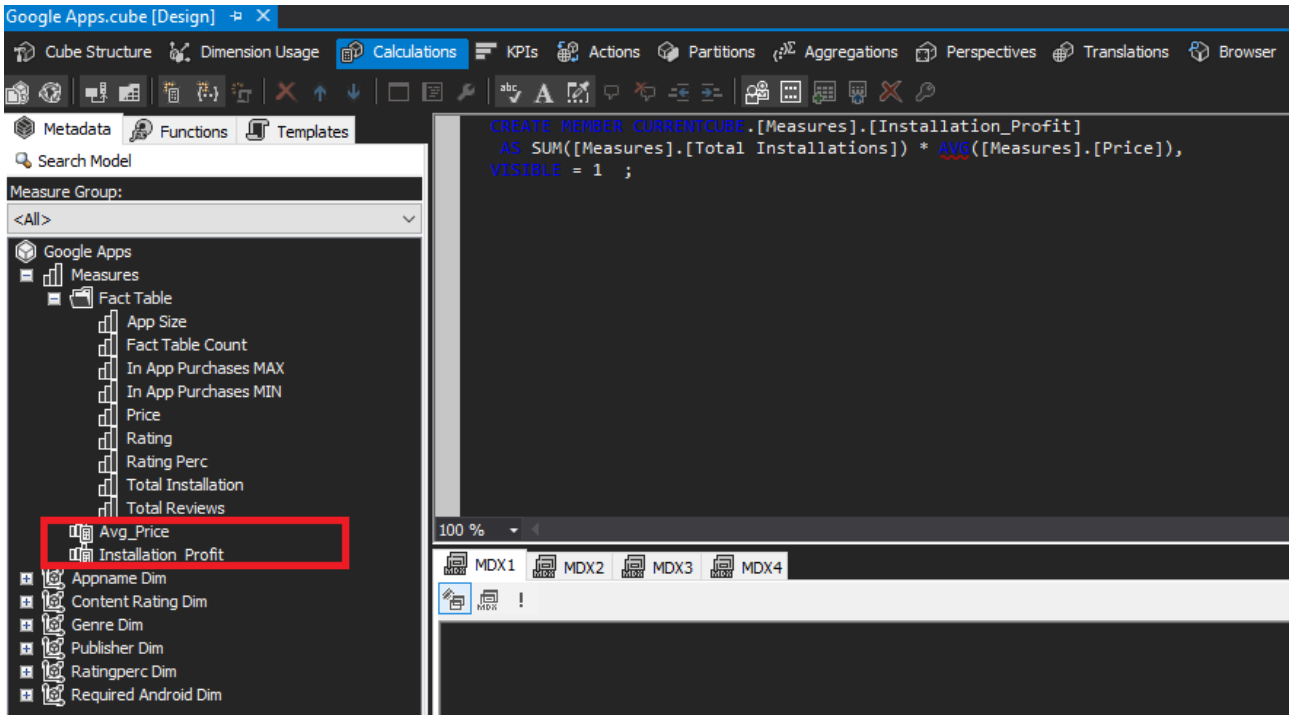
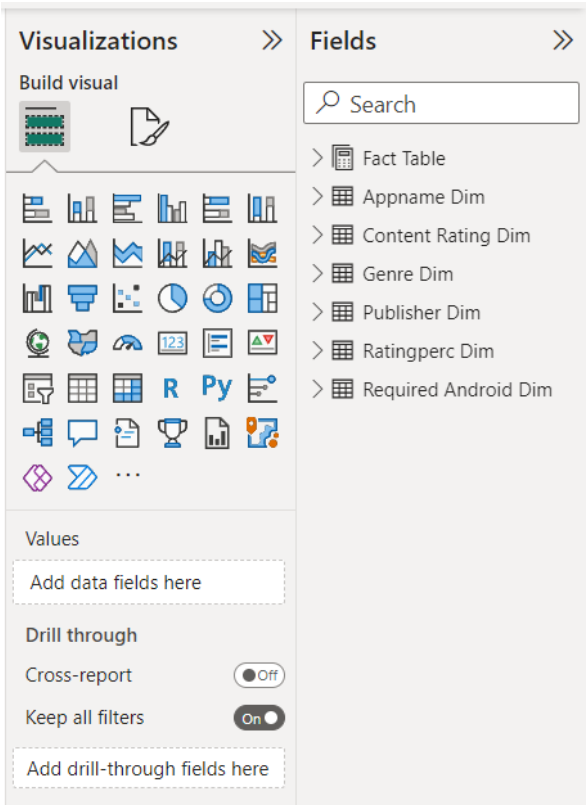


Figure 21: Calculated Measures

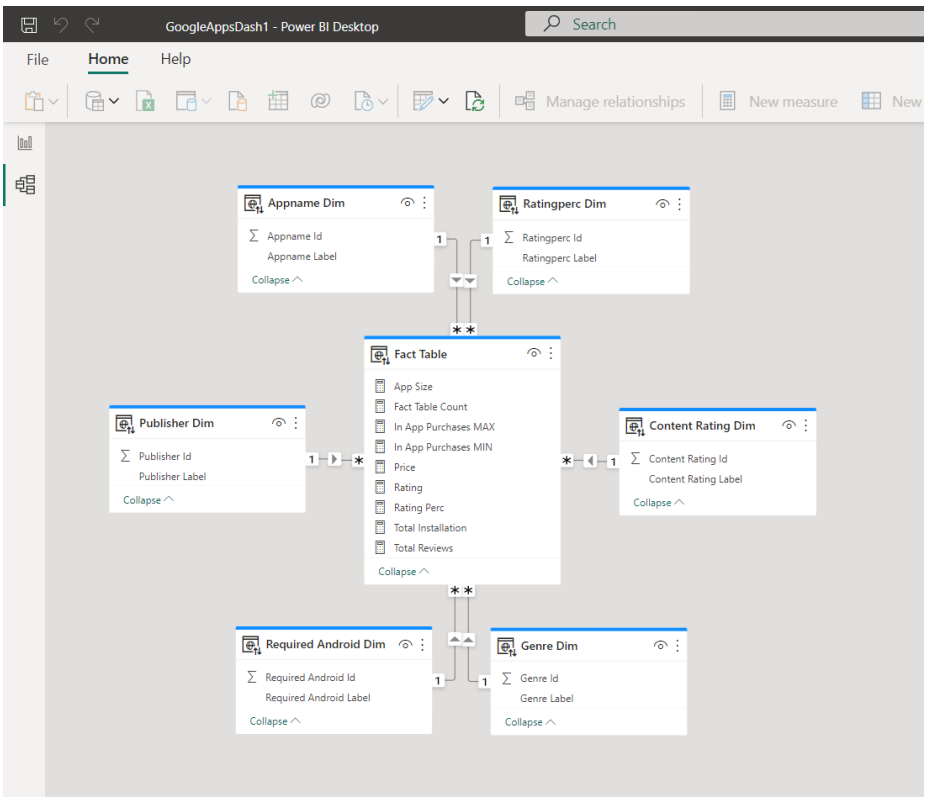
Data Visualization (MS Power BI)

After the completion of the OLAP services and the creation of some new measures that represent crucial deducted information incorporated in the analysis we proceed in the procedure of building in top of the cube the Power BI desktop app that is connected to the final data analysis views created in the OLAP. The data warehouse schema is inserted into the Power BI and the tables of the star schema are identified with their foreign keys' connections to the Fact table in the Power BI environment without any intervention.

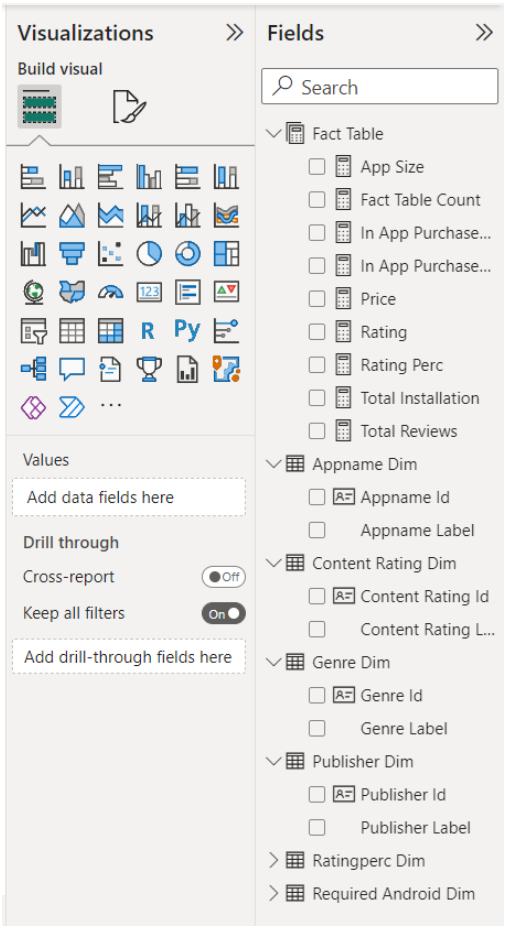


Data Visualization 1: Tables in Power BI

The schema is presented in the right column menu of the screen where the Fact table with the observations of each distinct application along with all its quantitative variables (metrics) is associated with the table's dimensions (the categorical variables). In every dimension table the id of the particular variable exists, and it is accompanied by its label price. In all the produced graphs in each report the actual variable used is the label variable which presents the qualitative information needed to be presented in the report. The diagram below shows the star schema data model in the environment of the Power BI.



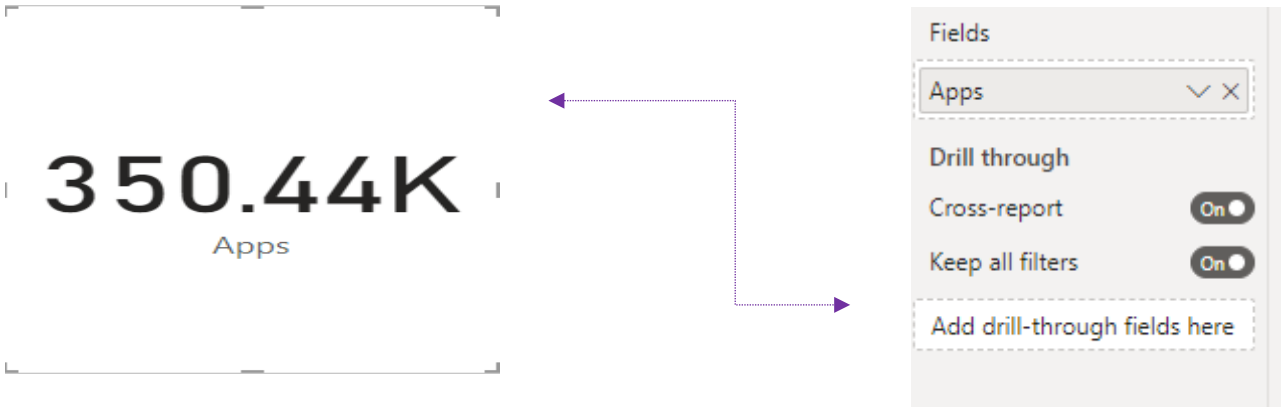
Data Visualization 2: Star Schema in Power BI



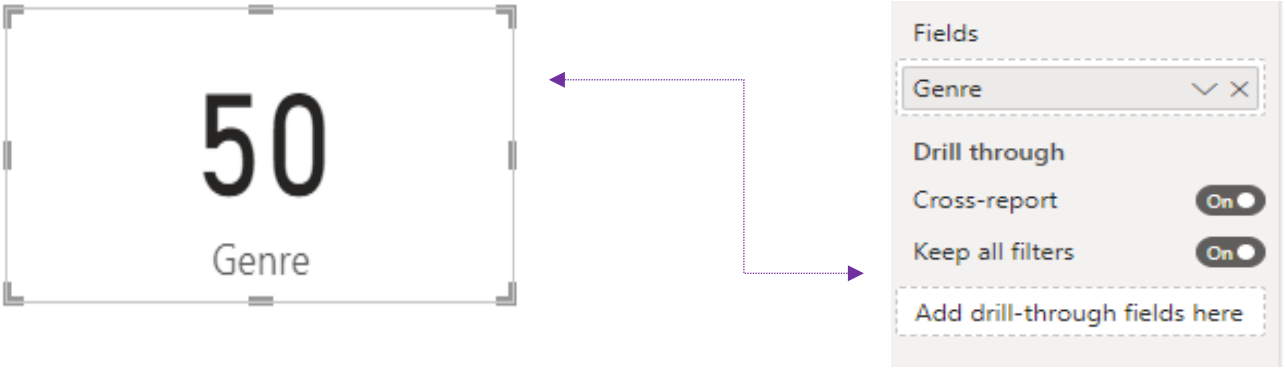
Data Visualization 3: Tables' Variables in Power BI

First Dashboard - Popularity Analysis

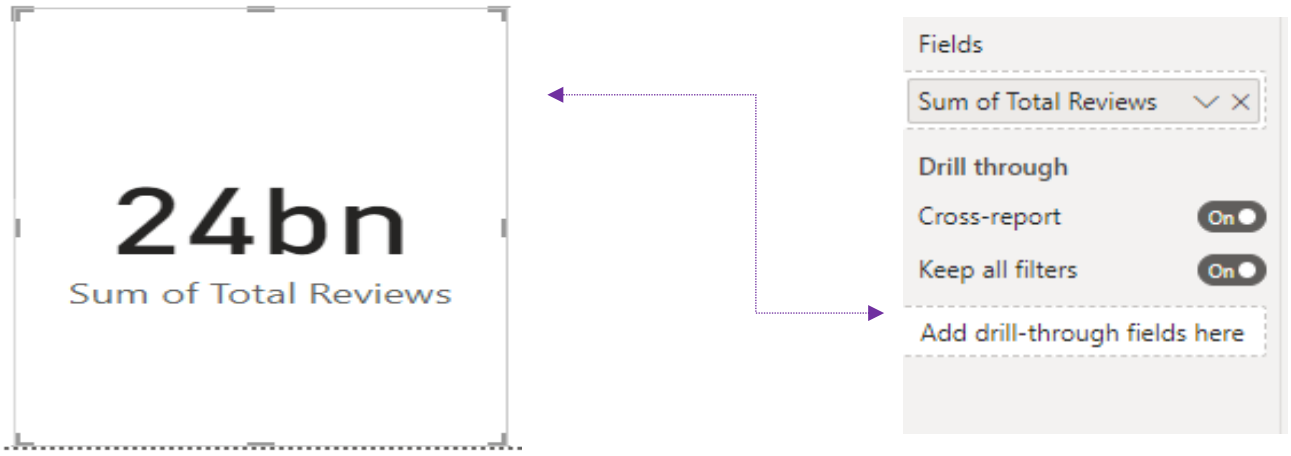
In the next steps the different dashboards for each distinct report are created. Depending on our business case it is important to explore the data and confirm that the seven categories of interest are actually valid choices for such a business investment move or another alternative should be proposed. In the first approach an overview of the main information of the dataset is created. To begin with, the single value cards show the different number of applications available for the analysis showing that 350.44K applications of all various genres that reaches 50 different categories are contained in the fact table and analysed. The total amount of installations of all these applications reaches the number of 24bn, while the public ratings average score for the rated ones is 4.2 up to 5 and this depicts a favourable public opinion of the public proportion which actually rated an app. It is of interest that the visual representation of the ratings graph in the single card does not belong in the basic visual's graphic library of Power BI but it is an added visual from initially created form MAQ software. Finally, the number of publishers is 102.87K. The created visuals for these overview data are presented below.



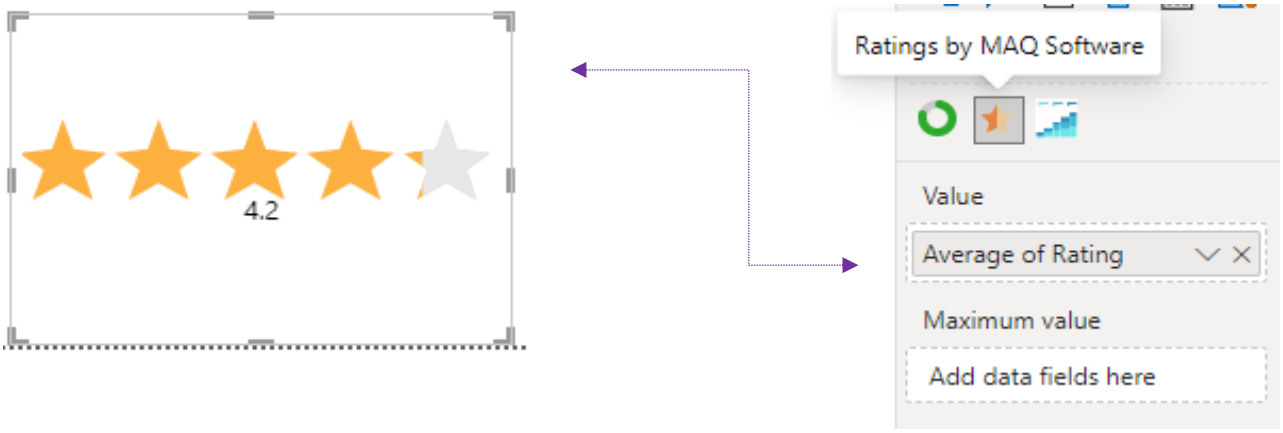
Data Visualization 4: Data fields



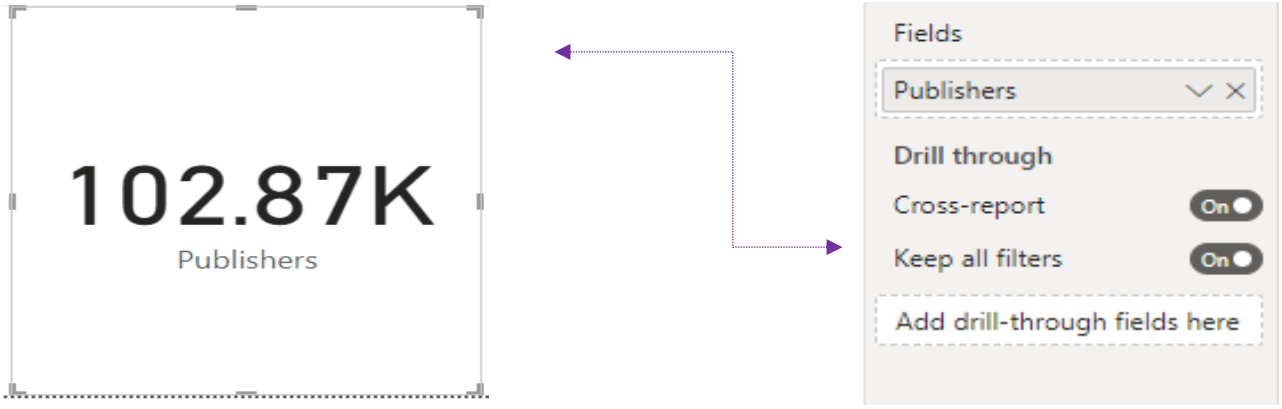
Data Visualization 5: Data fields



Data Visualization 6: Data fields

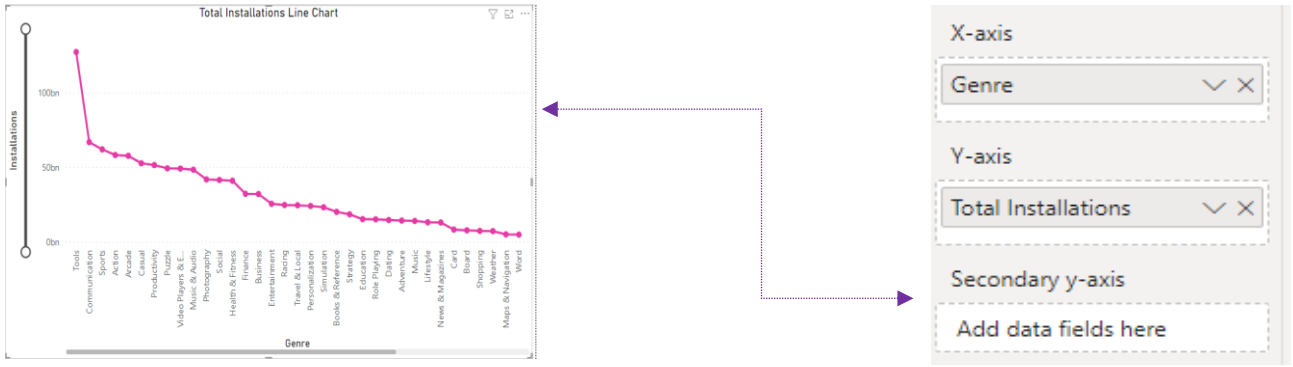


Data Visualization 7: Data fields



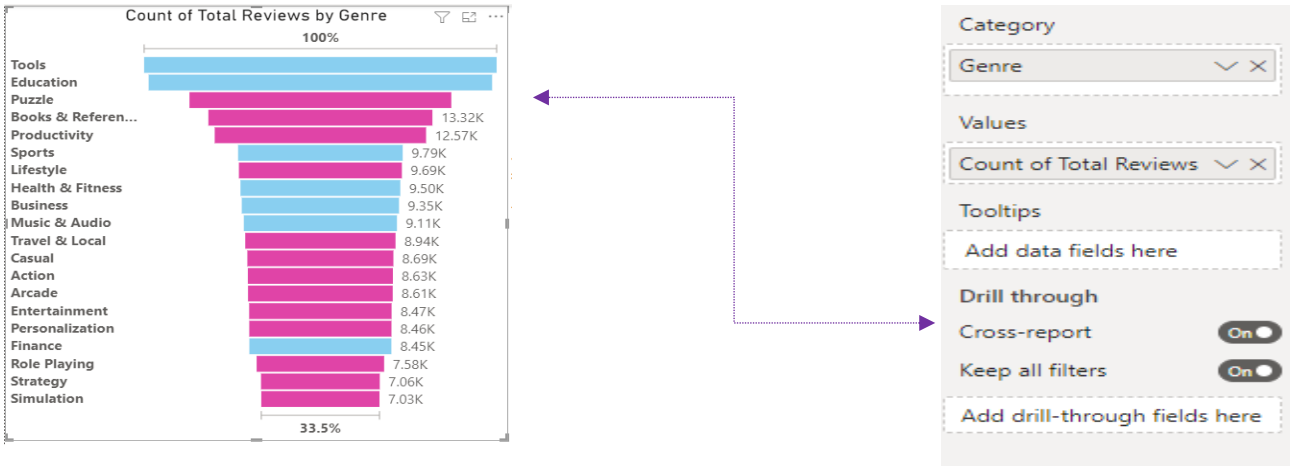
Data Visualization 8: Data fields

One fundamental explanatory analysis of the different genres' popularity is the line chart that shows fluctuation of the total amount of the installations. Every genre and the number of the total installations that corresponds to the sum of the total installations for every applications belonging to a particular genre is put on the particular graph.



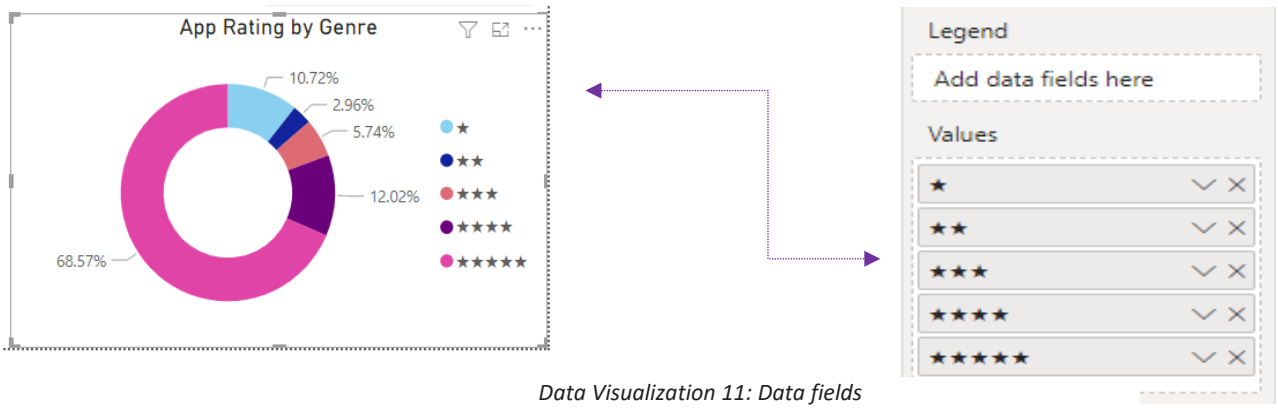
Data Visualization 9: Data fields

Another intriguing feature that expresses the popularity of an application genre is the number of reviews in comment forms, that all the applications of the particular genre have actually received. The funnel visual is selected for presenting the number of total reviews spotted for all applications of a particular genre in descending order with the total number of reviews in each category and the relative percentage of the smallest in terms of number of reviews genre in comparison with the genre with the biggest number of reviews.



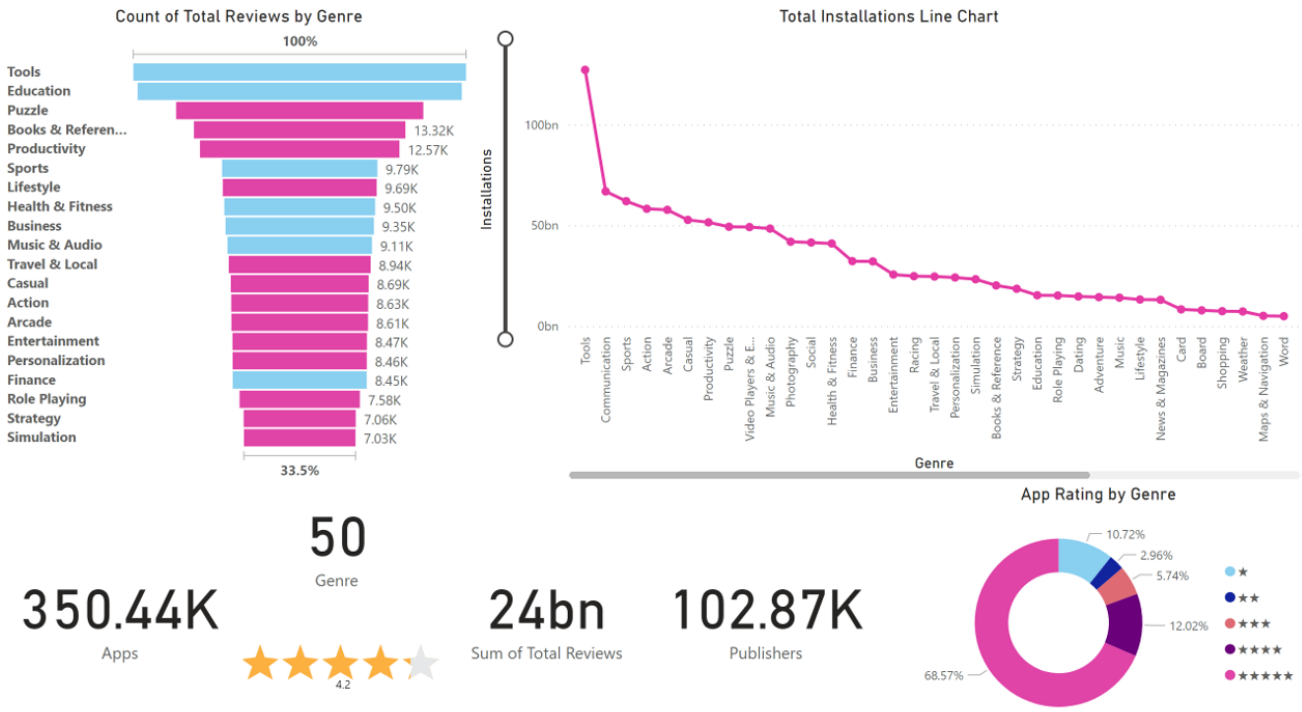
Data Visualization 10: Data fields

The last graph of the introductory dashboard is a donut chart that presents the percentages of the number of applications that have been rated with each distinct available grade in the scale of one to five stars.



Data Visualization 11: Data fields

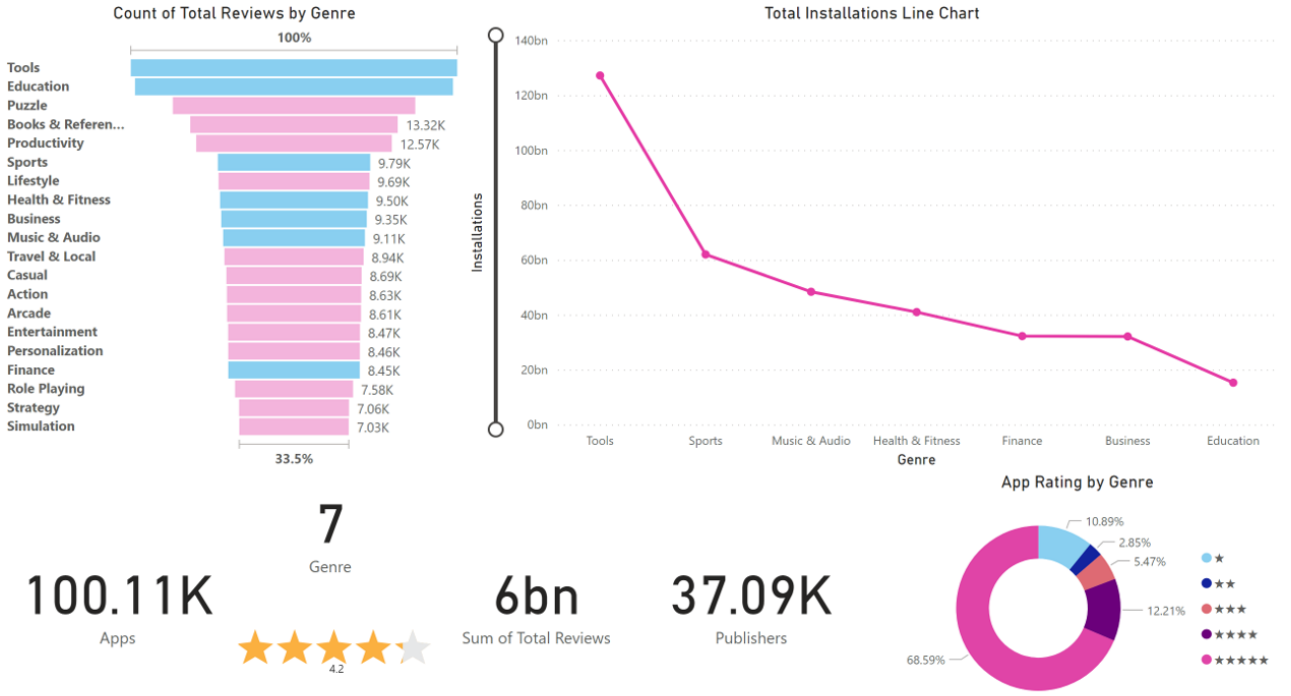
The first report is presented below:



Data Visualization 12: First Report of Genres' Popularity Analysis

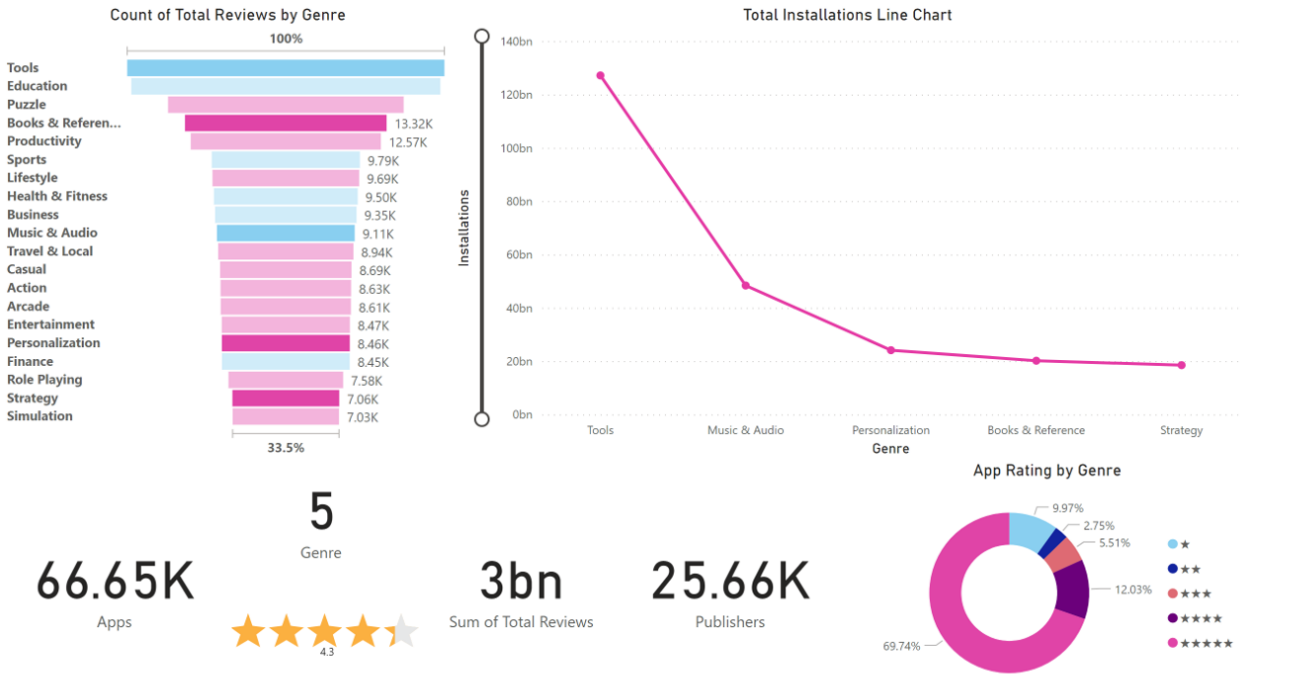
It is obvious that the five categories with the most installations of their applications are the *Tools, Communication, Sports, Action* and *Arcade* genre and they do not actually match with the proposed ones from our business case and generally most commonly known categories. On the other hand, the indicator of the number of the total reviews attributed to its genres reveals that the proposed categories of *Tools, Sports* are indeed important in the public's view but instead of *Action* and *Arcade,* the *Education* and the *Puzzle* gain ground and popularity. Inside the ten most popular ones in public reviews rests the categories of *Health & Fitness, Business* and *Music & Audio* that were the initial goals and assures that these genres are the most useful in public's opinion and could present the fields that could host a new application. The *Finance* category is picked as a complementary and directly linked category of the *Business* category.

So, in the next dynamic view the seven categories that were proposed and actually confirmed from the explanatory analysis to be popular and interesting applications genres are presented below. The number of apps in these categories is just 100.11K though the average rate in reviews is 4.2 (top score) and their installation's number is 6bn. Finally, it is profound that the *Tools* category is the first in the number of total installations with a major difference from the second one that of *Sports*. After taking into consideration the donut chart as well it is obvious that the general trend of proportionally high ratings is preserved among these categories as well as in the whole data set.



Data Visualization 13: First Report of Genres' Popularity Analysis Seven Categories

In the following snapshot the combination of five other application genres along with their general characteristics.

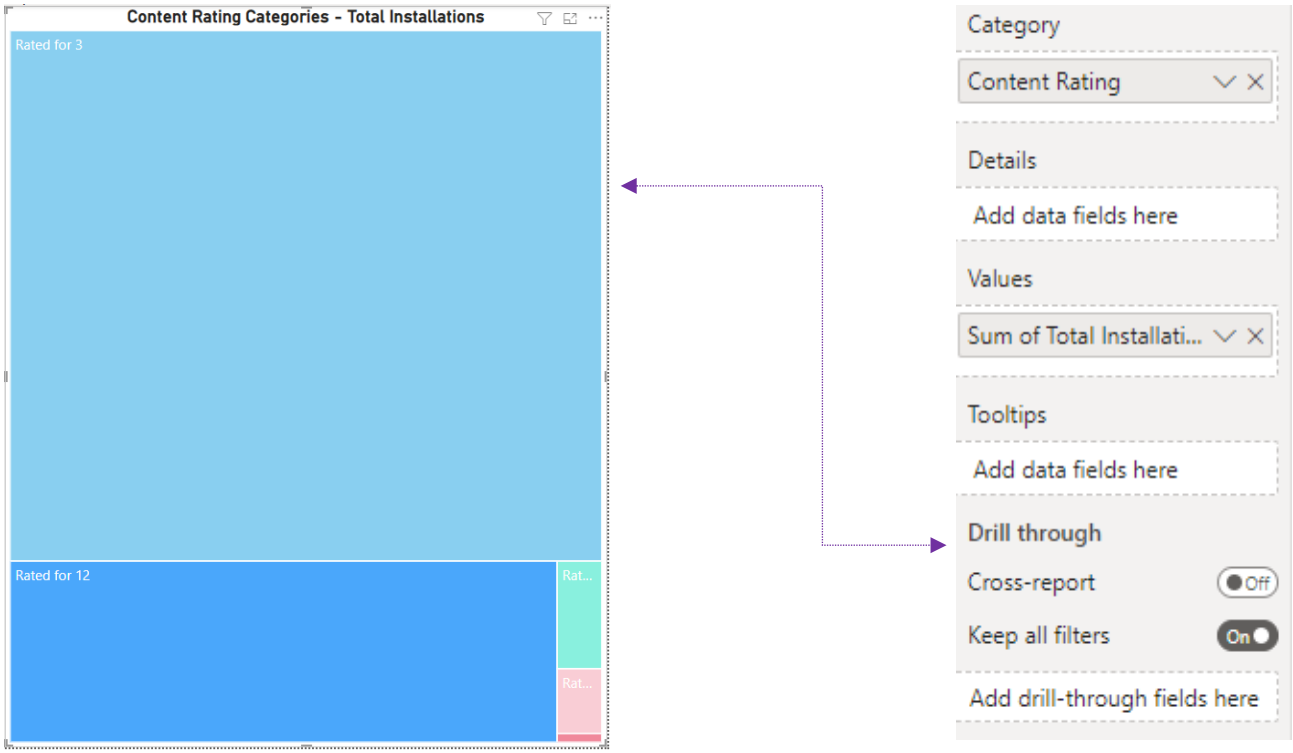


Data Visualization 14: First Report of Genres' Popularity Analysis Five Categories

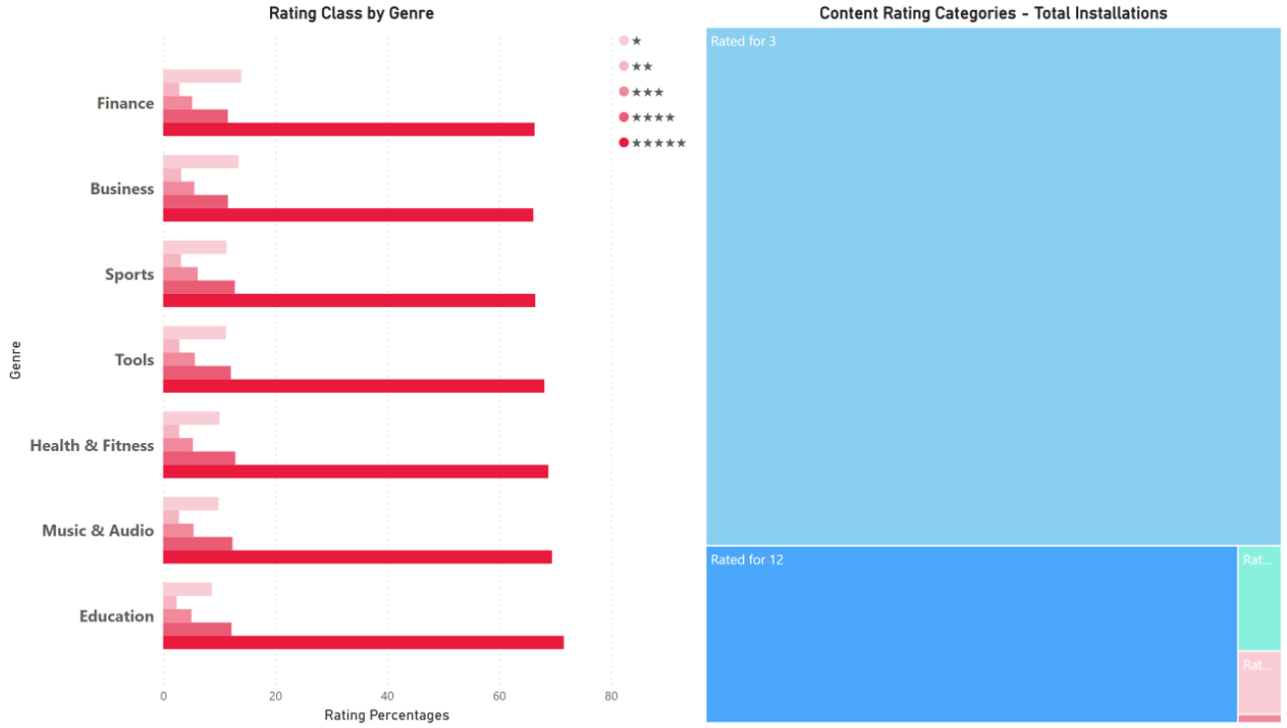
Second Dashboard - Rating Performance

In the second dashboard the different ratings are presented for each one of the seven categories along with their content ratings that reveals the percentages of each genre's applications that are appropriate for different age groups. To be more specific a clustered bar chart is created with clusters of the seven different genres examined and bars for each rating scale. Secondly, a tree map is created in order to depict the proportions of each content rating age group depending on the total installations number for each category.

Data Visualization 15: Data fields



Data Visualization 16: Data fields

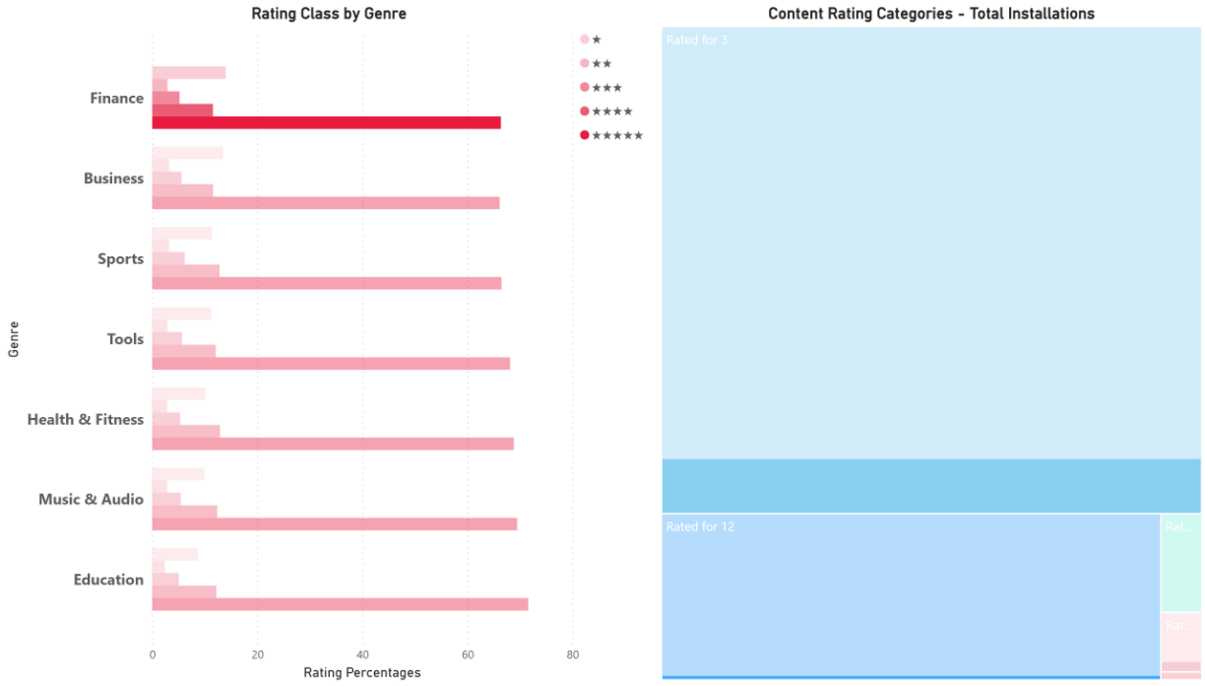


Data Visualization 17: Second Report of Genres' Rating & Content Rating Analysis

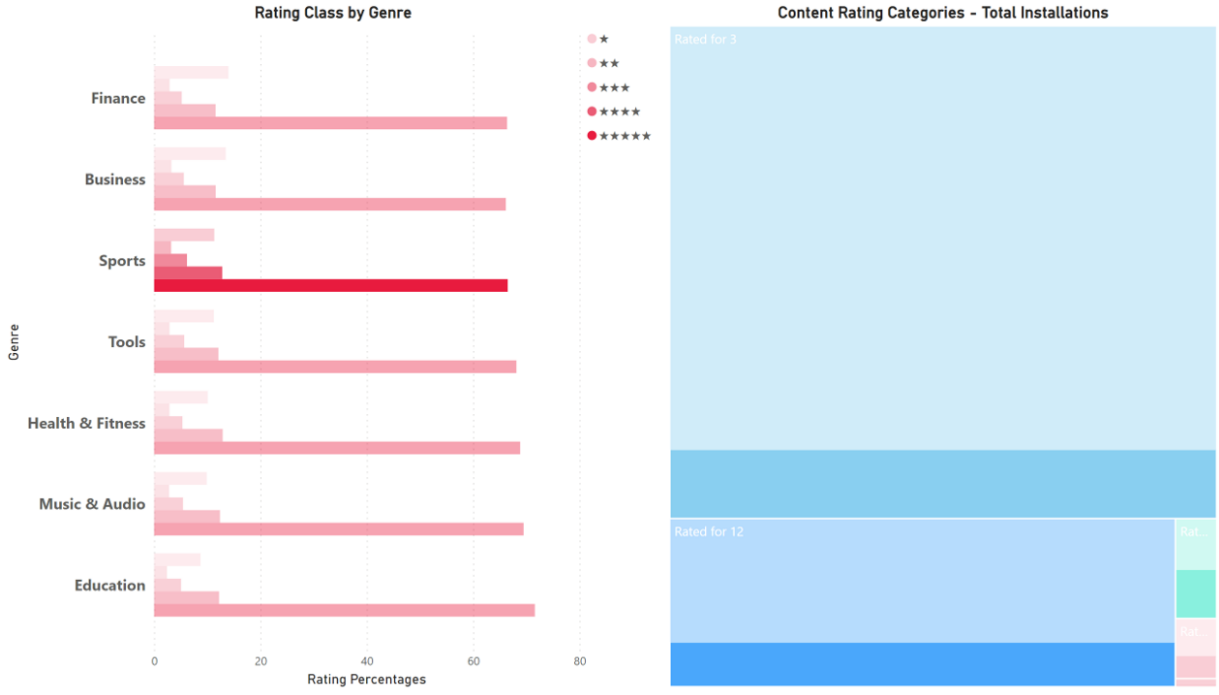
In all genre clusters the ratings applied follow the same trend. The rating of five stars (top score) represents the majority of the ratings for all genres and actually reaches the same proportion among the genre clusters. For the rest of the rating classes the same order is presented for the clusters. In detail the rating of four stars is the second in applications' number for each category with significant difference from the five stars rated applications of the category. One exception of this conclusion is the *Business* and *Finance* genre in which the second rating class is the rating of one star (basic score) with slight difference of the third group which is the rating class of 4 stars. Furthermore, the third bigger number of applications on each category (for the categories of *Sports, Tools, Health & Fitness, Music & Audio, Education)* is the rating class of one star (basic score) and then follows the rating class of three stars and finally the rating class of two stars.

In addition, the tree map shows the number of total installations of applications belonging in one of the seven aforementioned genres and simultaneously have a particular content rating for the appropriate applications for the age groups above 3 years old, 7 years old, 12 years old, 16 years old and 18 years old. Approximately, the 80% of the installations concerns applications for users above 3 years old while almost 15% of the installations are applications for users above 12 years old and the other age groups are extremely small minorities comparatively. From this overall view it is concluded that for all seven categories of interest the age factor does not limit the target audience of the categories.

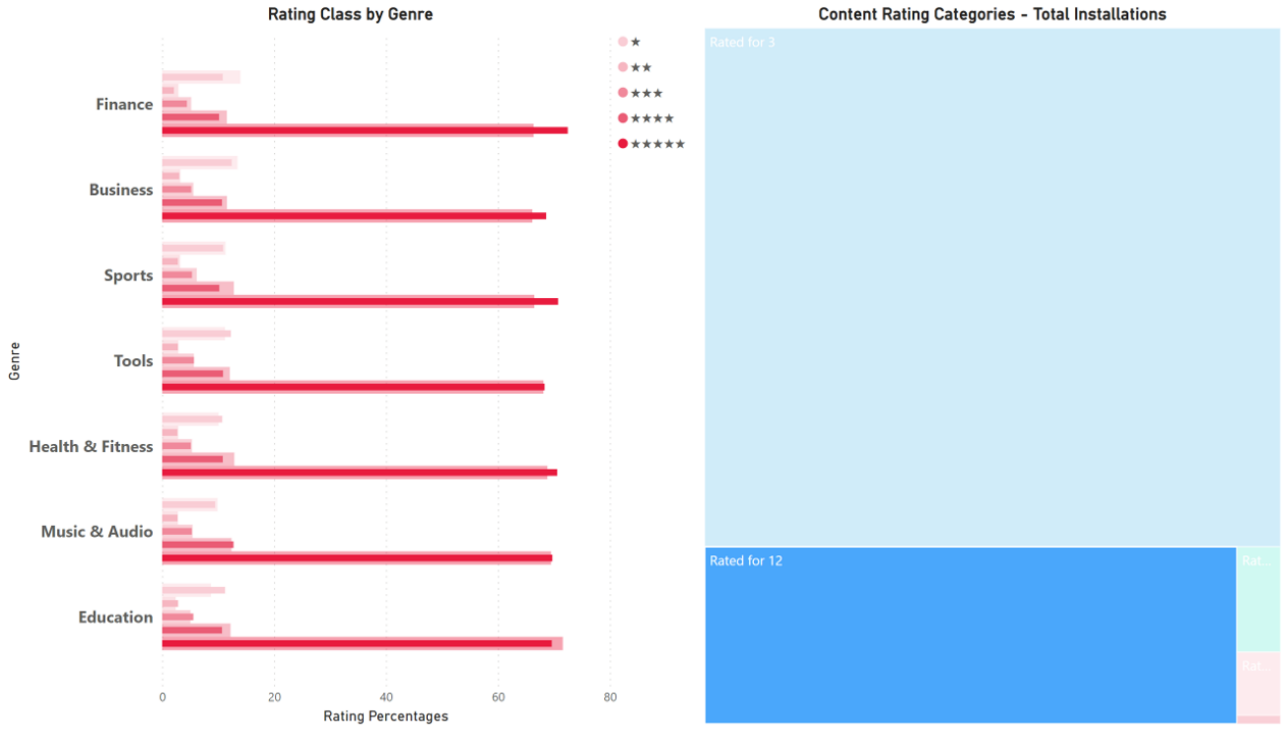
The observation above is applied clearly in the case of *Finance* genre but in case of the *Sports* genre a small difference is detected concerning the distinct content rating percentages where the *Rated for 3* present a significantly lower percentage while the *Rated for 12* presents a higher one as well as the *Rated for 7* and the *Rated for 16* in comparison with the results for each age category that are induced in the case of the *Finance* genre. When the aggregation factor per category is the Content Rating class, we see that the Rating classes for each genre cluster keeps the same proportion between the five, four, three, two and one star ratings as it has for the overall slides (figures for data visualization 20 & 21).



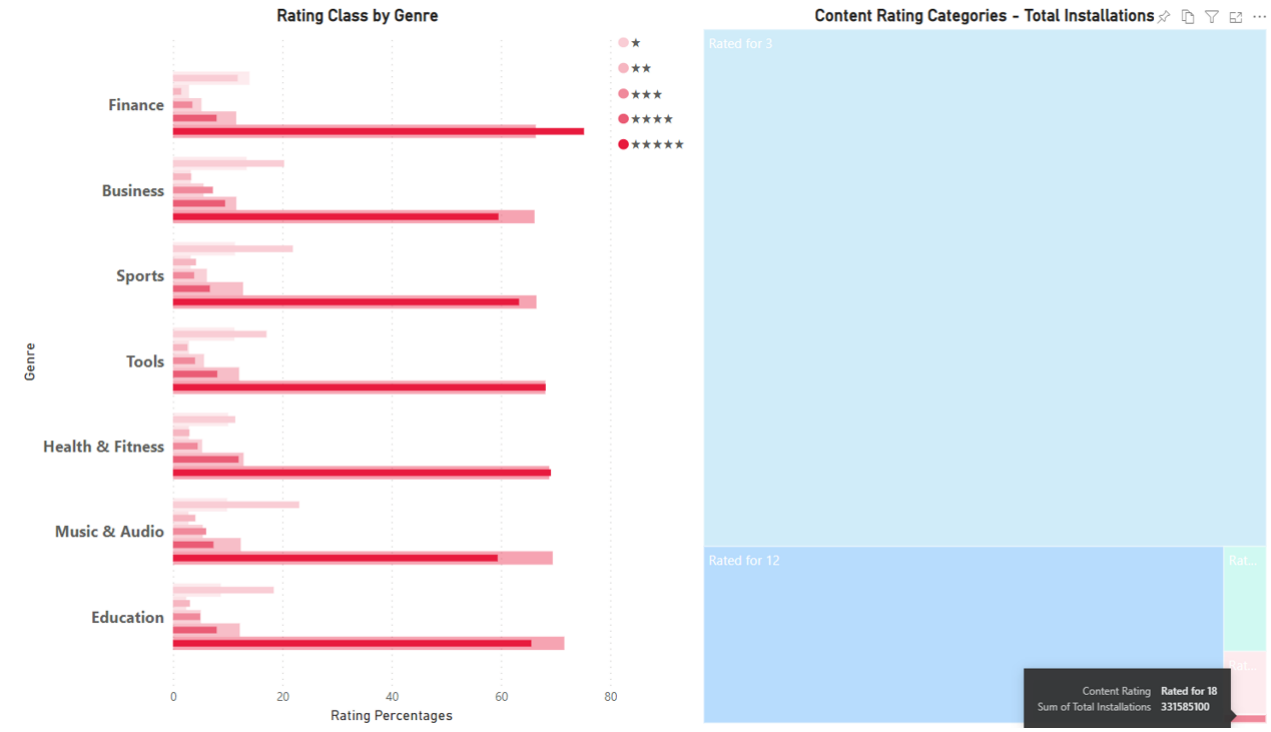
Data Visualization 18: Second Report of Genres' Rating & Content Rating Analysis - Finance



Data Visualization 19: Second Report of Genres' Rating & Content Rating Analysis - Sports



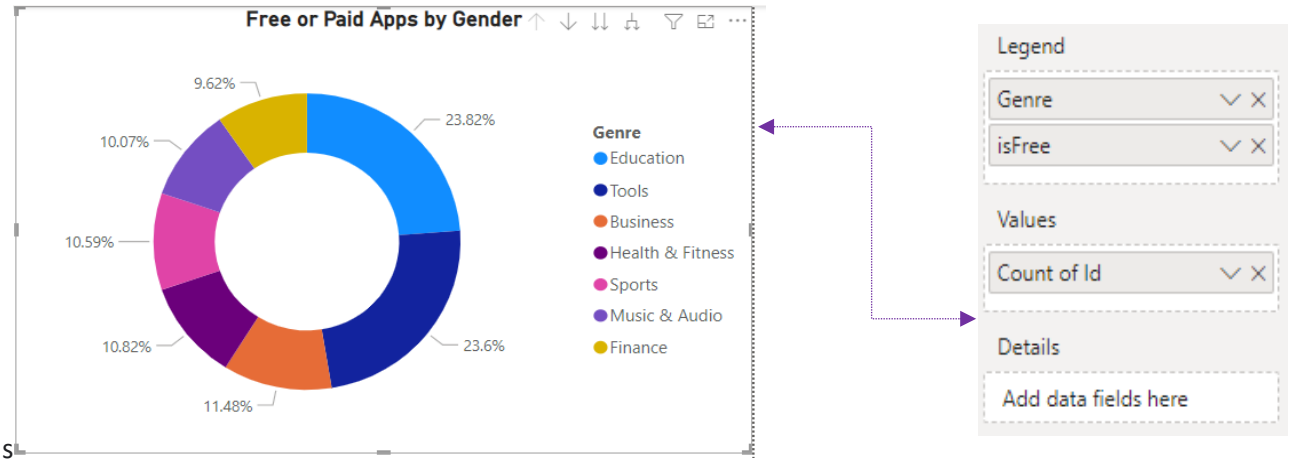
Data Visualization 20: Second Report of Genres' Rating & Content Rating Analysis - Rated for above 12



Data Visualization 21: Second Report of Genres' Rating & Content Rating Analysis - Rated for above 18

Third Dashboard - Profit Sources' Analysis

While proceeding in the analysis step, one essential aspect is the evaluation of the profit aspects and capabilities each genre can offer in new potential applications that could be launched in its category. The applications are distinguished by one fundamental factor that affects the profit sources that an application can offer and it reprsents if the application can be installed and acquired by the user for free or the user should pay an installation fee. To map the applications examined in our data set a donut chart is created with three levels of information. The first level is the proportions of free and non-free applications belonging in the seven genres, the second level presents the proportions of the applications of the seven different genres, and the third level of detail is the one of the combination of the previous two that presents the free apps of each category and the non-free apps of each categroy.



Data Visualization 22: Data fields

The first evaluation process contains the review of the profits from applications of installation for the paid apps. In this framework a new measure is created that is called *Average Installation Price (Avg_Price)* that calculates the average installation price for each genre by ignoring the zero prices that represent the free applications.

```
1  Avg_Price = CALCULATE(AVERAGE('appfinalmaybe'[Price]), FILTER('appfinalmaybe', 'appfinalmaybe'[Price]<> 0 ))
```

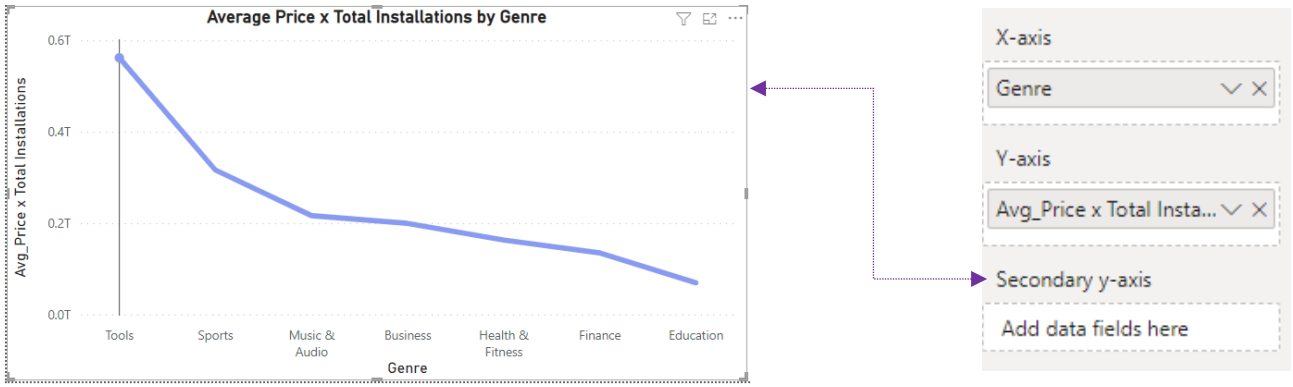
Data Visualization 23: Measure for Average Installation Price

Data Visualization 24: Data fields

The total profit from installations for the paid apps is depicted in a line chart for all of the seven categories and it is calculated using the new metric of *Avg_Price* and the *Total_Installations* of applications belonging in each different genre.

```
1 Avg_Price x Total Installations =
2 [Avg_Price] * SUM('appfinalmaybe'[Total Installations])
```

Data Visualization 25: Measure for Average Installation Profit for non-free apps



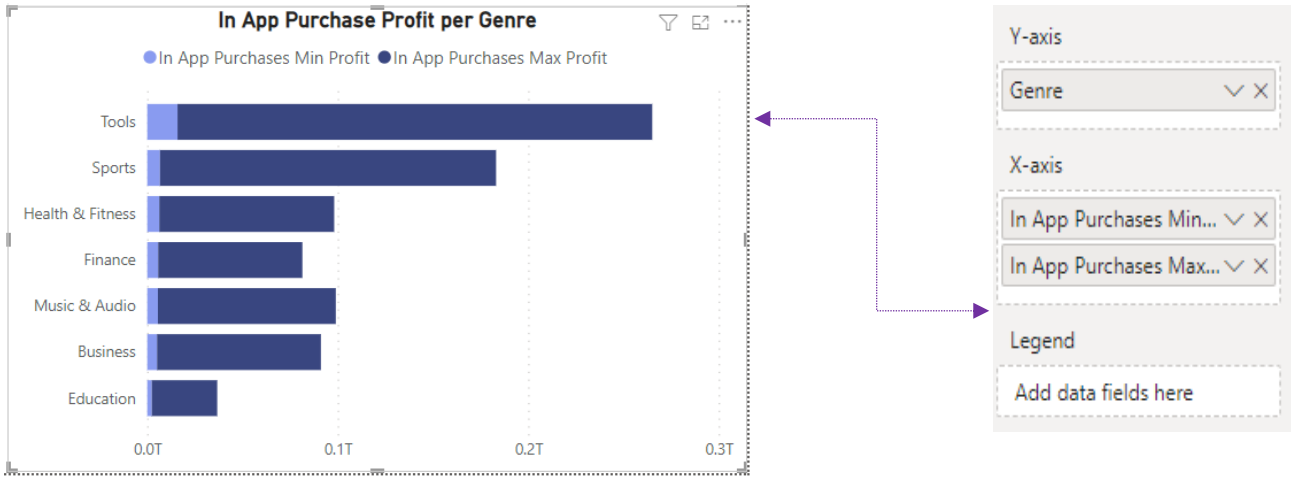
Data Visualization 26: Data fields

The second profit source of an application usage that corresponds in both free and non-free apps is the purchases of additional applications features and capabilities that the app can offer by purchases made form users after the application's installation and usage of the basic package. Every application offer additional functionalities in a minimum price that a user can pay and the premium functionalities that corresponds to the maximum price a user can spend in application purchases. In order to calculate the profits from the basic packages and the premium packages we create four new measures in the dataset depending on two initial metric of the dataset the *In App Purchases Min* & *In App Purchases Max.* Firstly, we creat the average measures for these variables (and for this computation the zero prices are excluded from the computation). In the next step the profit from the minimum and maximum application purchases are calculated for the number of users, otherwise per installation, that actually end up buying such an application additional

service. The number of installation that will end up to in app purchases is estimated by the statistical research on app revenue statistics by Connor McMahon of the Zippia website[14] for 2022 to regarded to 5% of the total number of users that have installed the application. To complete this profit cacluation for two new measures are created one for the In App Purchases Profit and one for the In App Purchases Max Profit using the average prices for min and max purchases and the estimation of total installation that end up to attain such purchases from an application after installing it. These profits are separetely presented for each distinct application genre.

```
1 Inst_App_Purch = SUM('appfinalmaybe'[Total Installations]) * 0.05 * ('appfinalmaybe'[Avg_InApp_Min] + 'appfinalmaybe'[Avg_InApp_Max])
```

Data Visualization 27: Measure for In App Purchases Profit of applications

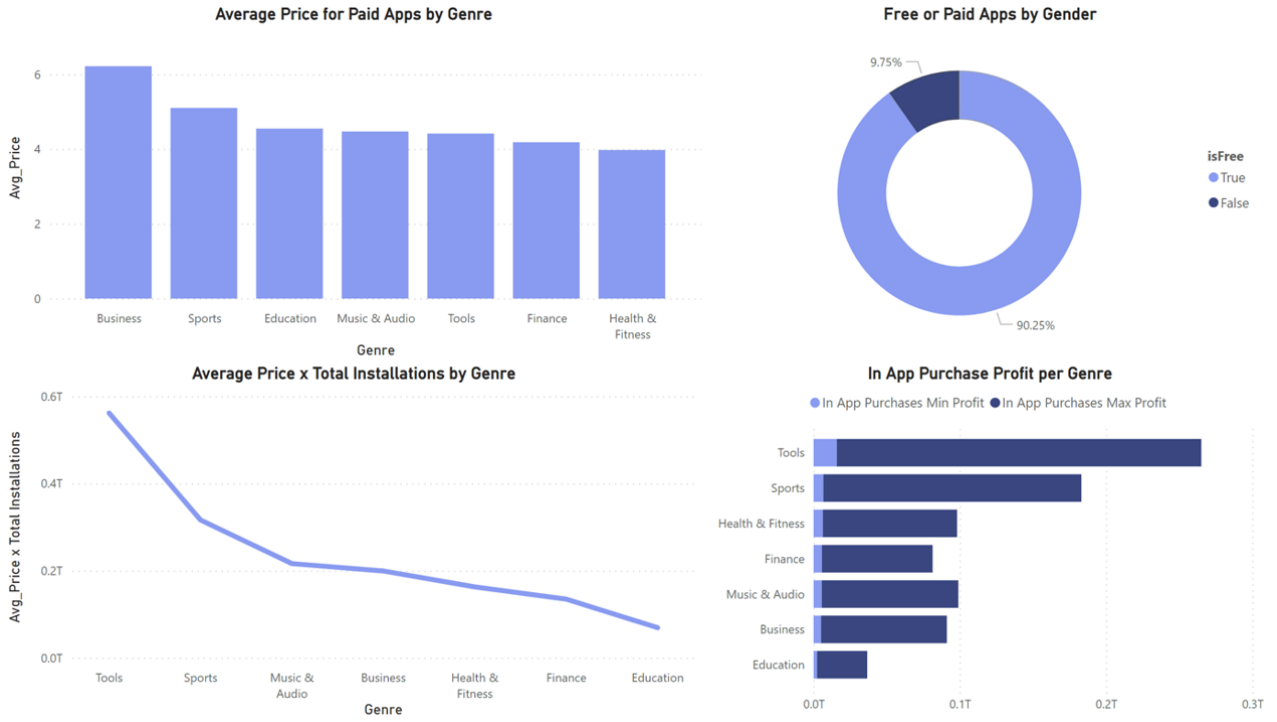


Data Visualization 28: Data fields

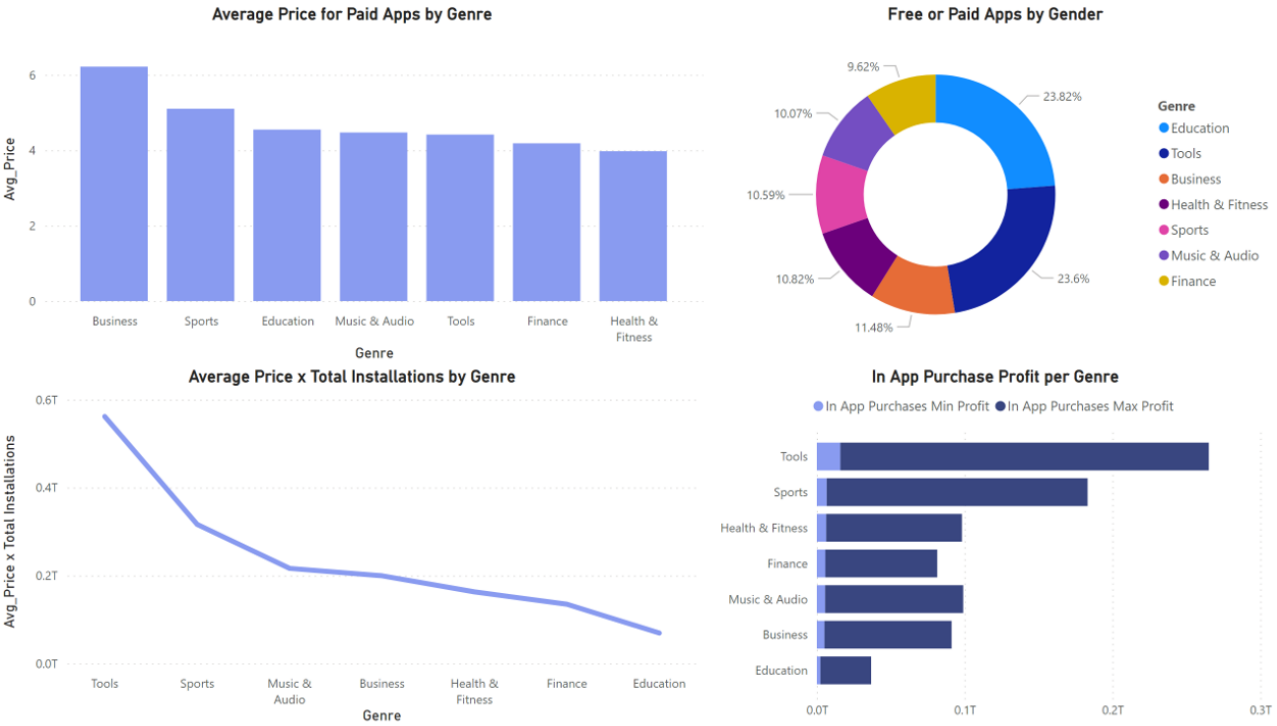
The first view shows that the 90.25% of the applications are free of installation cost and represent the predominant market share that reveals that an application adresses better to the audience. The non-free apps represent the 9.75% of the total applications. The installations profit for the non-free applications shows a range of average installation profits that varies among the genres. The Business apps presents the maximum price with Sports coming directely afterwards while in the third positions with the highest average installation prices comes the Education genre. Although it is shown that the profit form installations from the non-free apps is bigger for the Tools, Sports and Music & Audio genres that shows that these genres are considered more popular and considerably have more installations despite the fact that they are not the most expensive ones. The Tools genre reaches the 0.6T euros in installation profit.

Regarding the profits from In App Purchases it is obvious that most apps offer a wide ragne of additional features capabilities and the minimum price can differ essentially from the maximum one offered in applications of the same genre let alone in the same application. The profits from these additional purchases can reach 0.3T euros for Tools cataegory, 0.2T for Sports category and in the tird place of this profit source category comes the Health & Fitness genre with total revenues of 0.1T euros. The dashboard is presented below. While we drill down to the next level of the donut chart we see that Education is the genre that hosts tha most apps (although in the profit line chart it is shown that it does not have the bigger number of installations) and then follows the Tools genre and in third place of apps naumber in its cataegory comes the

Business apps where we can see from the profits line chart and bar chart that it actually has the smallest number of installations.



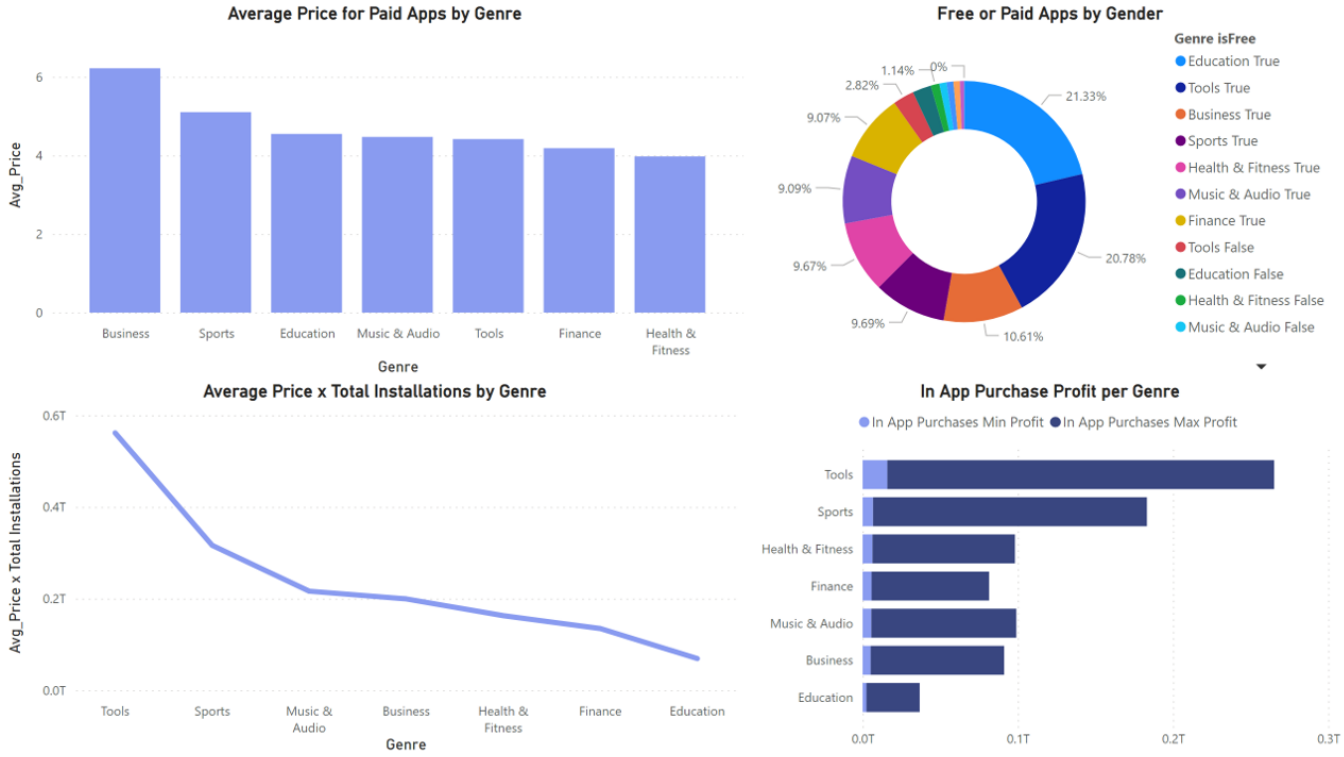
Data Visualization 29: Third report on Applications Profit Sources - Level 1



Data Visualization 30: Third report on Applications Profit Sources - Level 2

The general trend between the percentages of free and non-free apps is retained inside its genre with absolute compliance. In all the genres the vast majority is free apps. The Education, Tools, Business are the

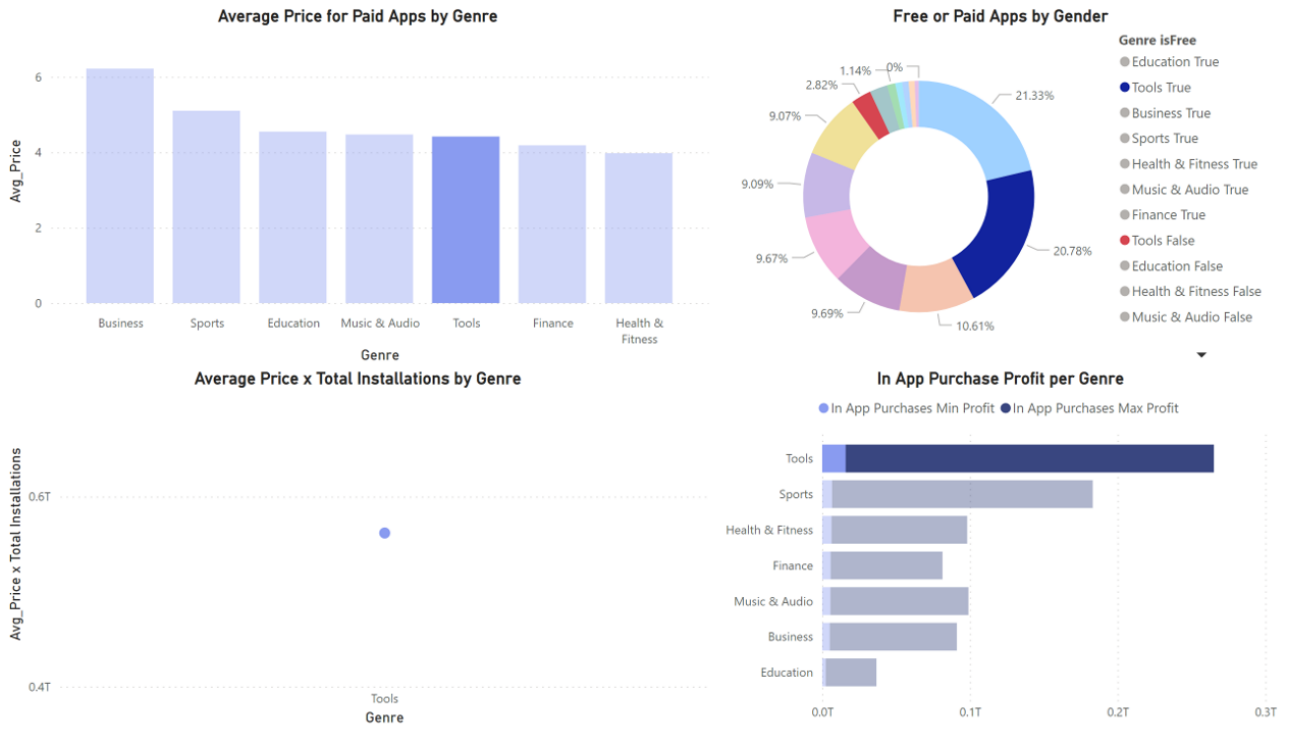
genres with the most free apps and then it is shown that Tools, the Education and the Health & Fitness have the biggest number of non-free apps.



Data Visualization 31: Third report on Applications Profit Sources - Level 3

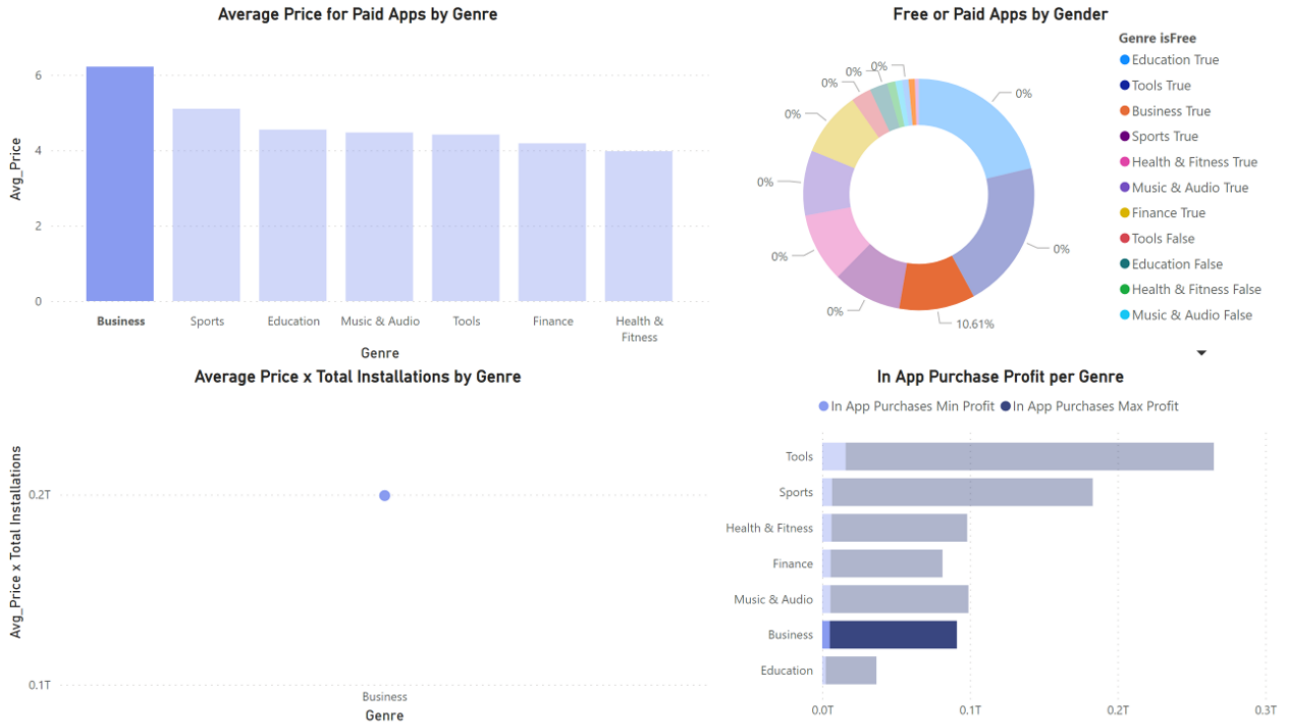
A further examination is made for the genre of Tools and Business. The first one is chosen because it represents the category with the most actual installations as it is a general qualitative category, the Business genre presents the most expensive category as it is shown in the bar chart for its non -free applications and finally both of them are considered to have the highest percentages of different applications belonging to its category (the Tools have 23.6% of the total number of available applications among the seven categories and the Business 10.61%).

The free Tools apps category has the 20.78% of all applications and the non-free apps category has the 2.82% of all applications, the fifth average installation price of almost 4 euros per installation of a Tool app and the first position in profits from both installations (for non-free apps) and app purchases, almost 0.6T in the first case and 0.3T in the second case of app purchases.



Data Visualization 32: Third report on Applications Profit Sources - Tools genre

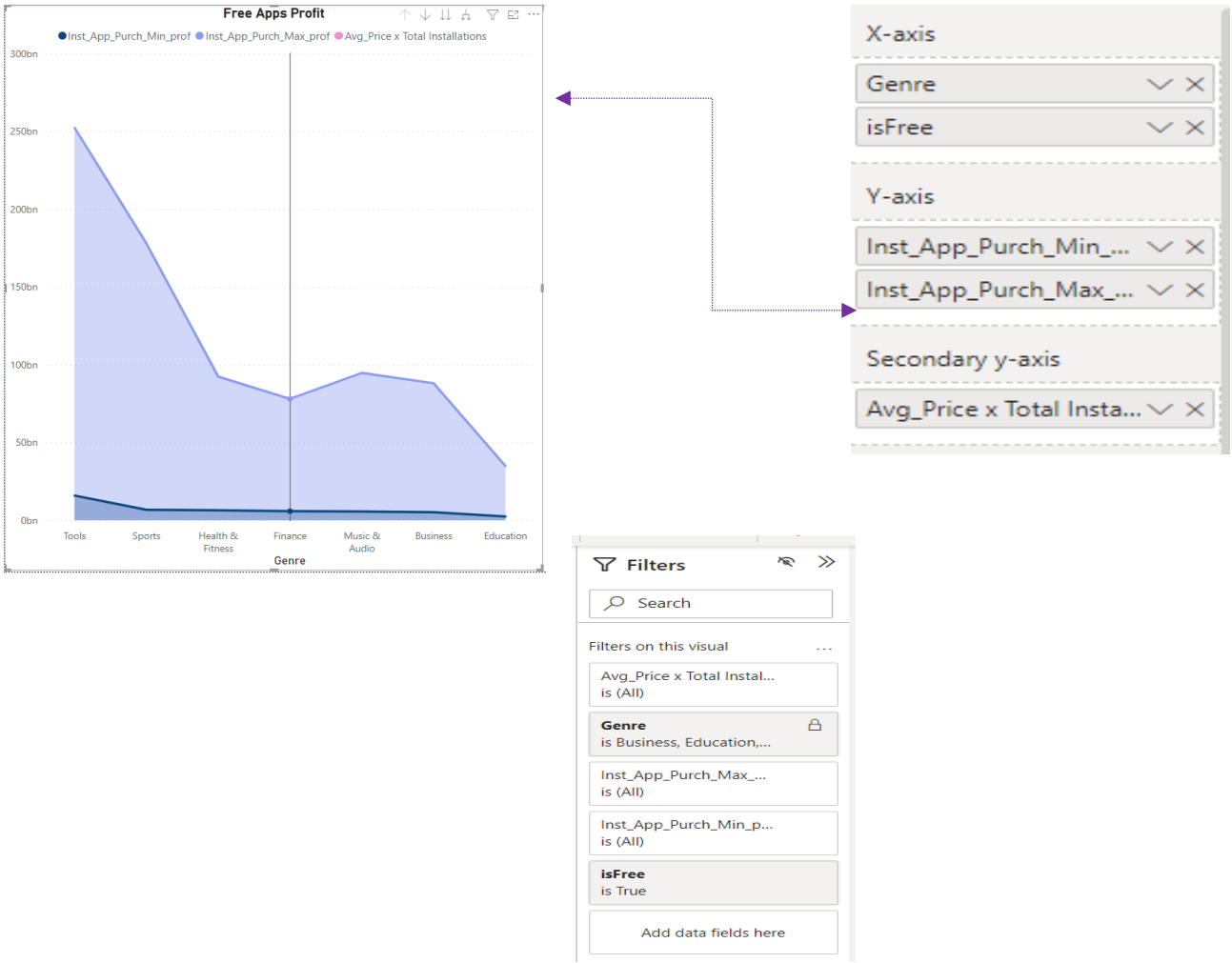
The free Business apps category has the 10.61% of all applications and the non-free apps category has the 0.86% of all applications, the first average installation price of almost 6 euros per installation of a Business app and the fourth position in profits from installations (for non-free apps) and the sixth position in app purchases, almost 0.2T in the first case and 0.1T in the second case of in app purchases.



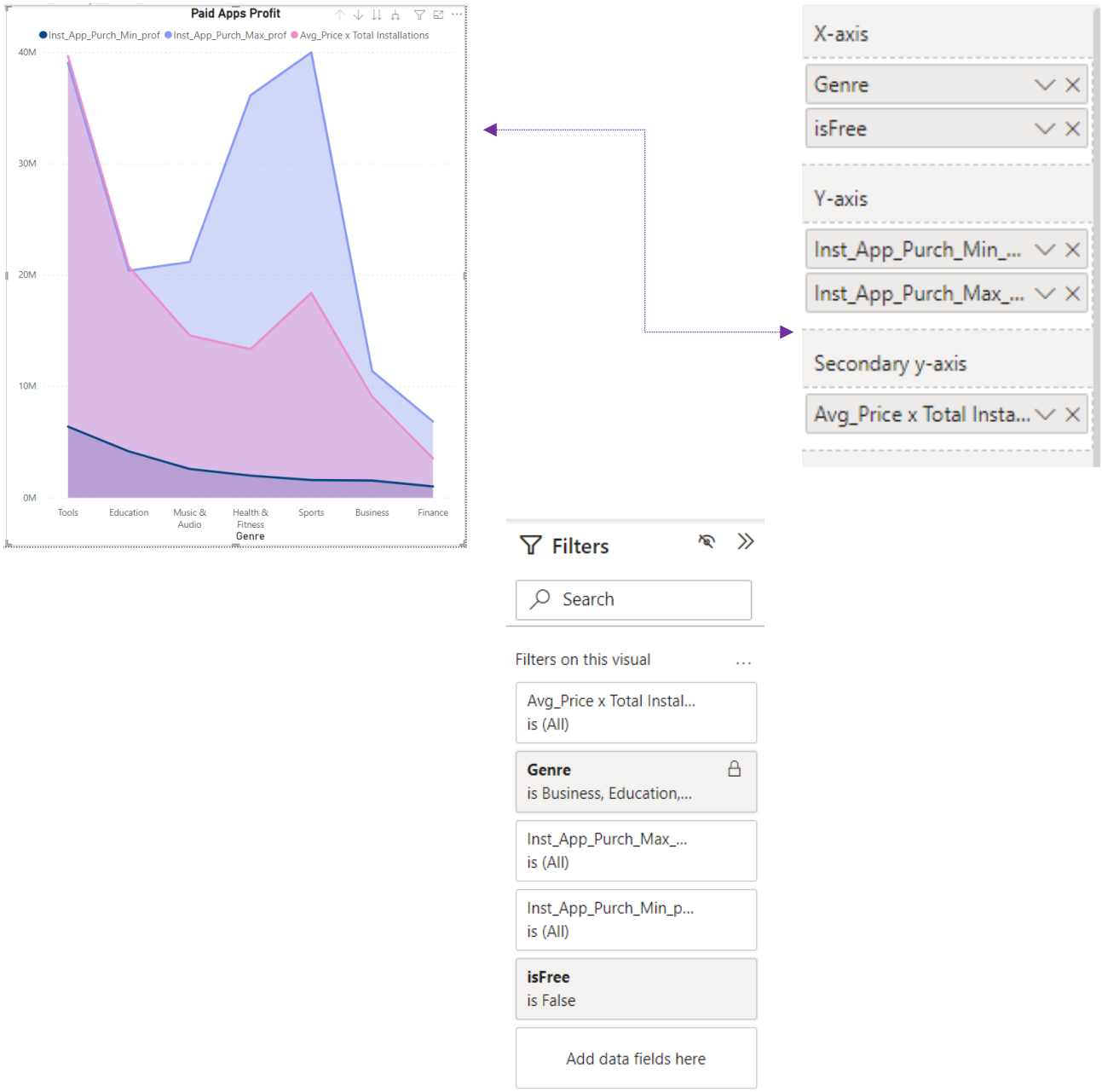
Data Visualization 33: Third report on Applications Profit Sources - Business genre

Fourth Dashboard - Free and Paid Apps Profits

In order to reach conclusions, it is useful to analyse the profits for the distinct categories of free and non-free apps of its genre and a clearer representation of profit efficiencies would be presented. Two area charts are created, one for the paid apps and one for the free apps. In each graph the profit sources are presented and simultaneously compared. Both area graphs are created with genre and free status categorical variables in the x-axis and In App Purchases Min Profit and In App Purchases Max Profit in y-axis and the Avg. Installation Profit in secondary y-axis. The first graph is filtered in order to present the free apps (Data Visualization 29) and the second one the paid apps (Data Visualization 30).



Data Visualization 34: Data fields and filtering

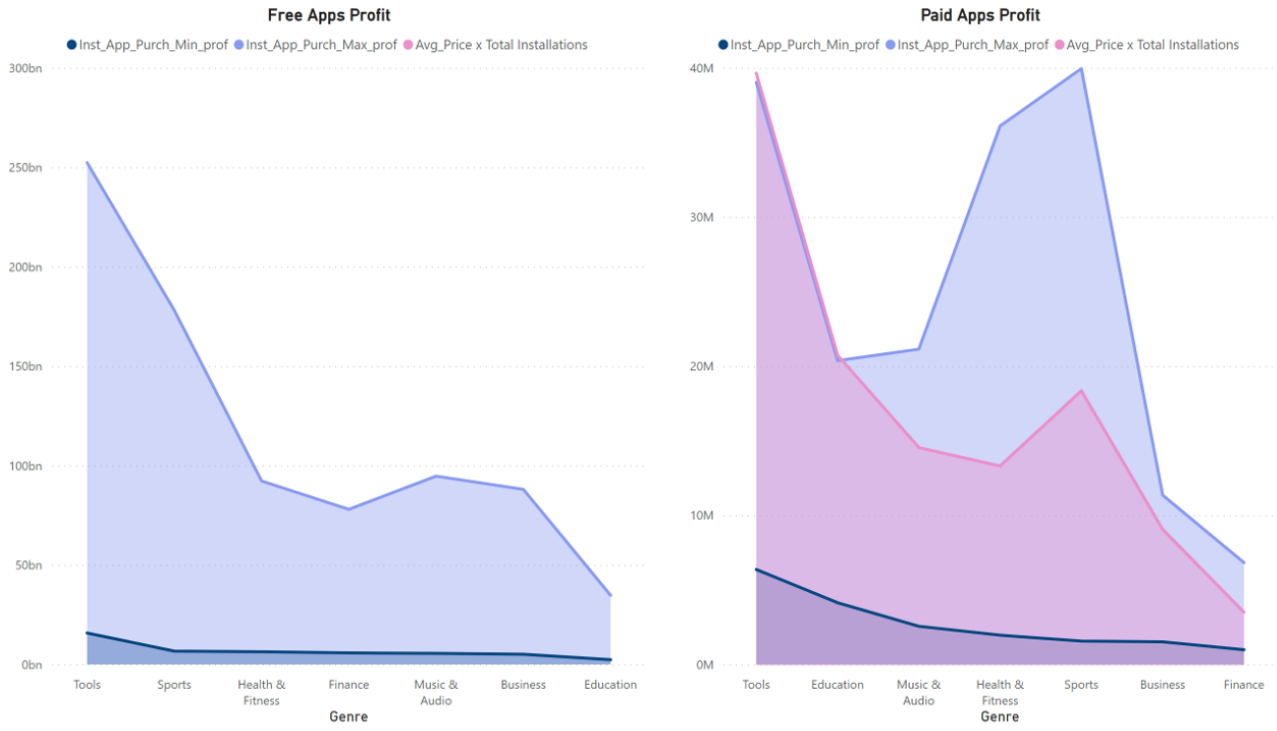


Data Visualization 35: Data fields and filtering

For the category of the free apps the profit sources are the In App Purchases with prices for premium and most expensive added functionalities being a present choice for all apps' genres as we see that the profit from the max priced purchases is significantly bigger from the profit attained by the less expensive available In App Purchases. The two genres with higher profits from In App Purchases are the Tools with 250 bn euros and then Sports with considerably smaller profit value. The genres with the smallest comparatively profits are the Business and the Education (with 88bn and 34 bn respectively).

For the paid apps category, it is generally concluded form the overview that the overall profit is much smaller in the class of millions. This is fully justified by the fact that the proportion of paid apps are one-tenth of the free apps, so the profit margins on absolute scale are much more limited. Tools and Sports are the genres with the highest overall profits, with 200M in the Tools genre and 100M in the Sports genre. Although the In App Purchases profit for both genres is 40M, the Average Installation profit is double for the Tools genre, a fact that was expected as the Tools is the dominant category in terms of Installations. In the last two positions - genres, we find the Business with its Installation profit being very close to the In App Purchases profit which

actually means that this genre contains apps that users are willing to pay to acquire them and they probably contain a lot of functionalities in their basic editions. In contrary to the free apps the Education has the second place in terms of total profit as the average installation profit is considerably high for Education as it is one genre with expensive installation cost in comparison with the others and as the Business genre the total profit from installations is almost equal to the profit from the in app purchases and this indicates that this genre contains apps that offer in their basic editions already enough functionalities and so they offer services that are useful to a particular audience.



Data Visualization 36: Fourth report on Applications Profits for Free & Non-free apps

Fifth Dashboard - Key Profitability Indicators

Despite the indicators of the total profit corresponding in a certain genre and its free status for the final decision concerning the possible genre that an upcoming investment would have potential, one profit Key Performance Indicator is measured and evaluated as factor that will lead to reach a conclusion of the genre area that could be proposed for a new application to be launched. The Key Performance Indicator is the average profit per installation of application belonging in a particular genre and it is measured separately for the free and non-free apps as the profit sources and the target audiences for these subcategories is actually differentiated.

The profitability KPI for the non-free apps is measured by using the three new metrics previously created form the initial variables of the dataset. The sum of the profit sources and especially the average installation profit for all apps in the genre is added to the In app purchase profit and then divided by the total number of installations of paid apps of the particular genres. On the other hand, for free apps the profit KPI is measured by adding the in app purchase profit for all installations of all apps of a genre and divide with the total number of installations of the free apps belonging to a category.

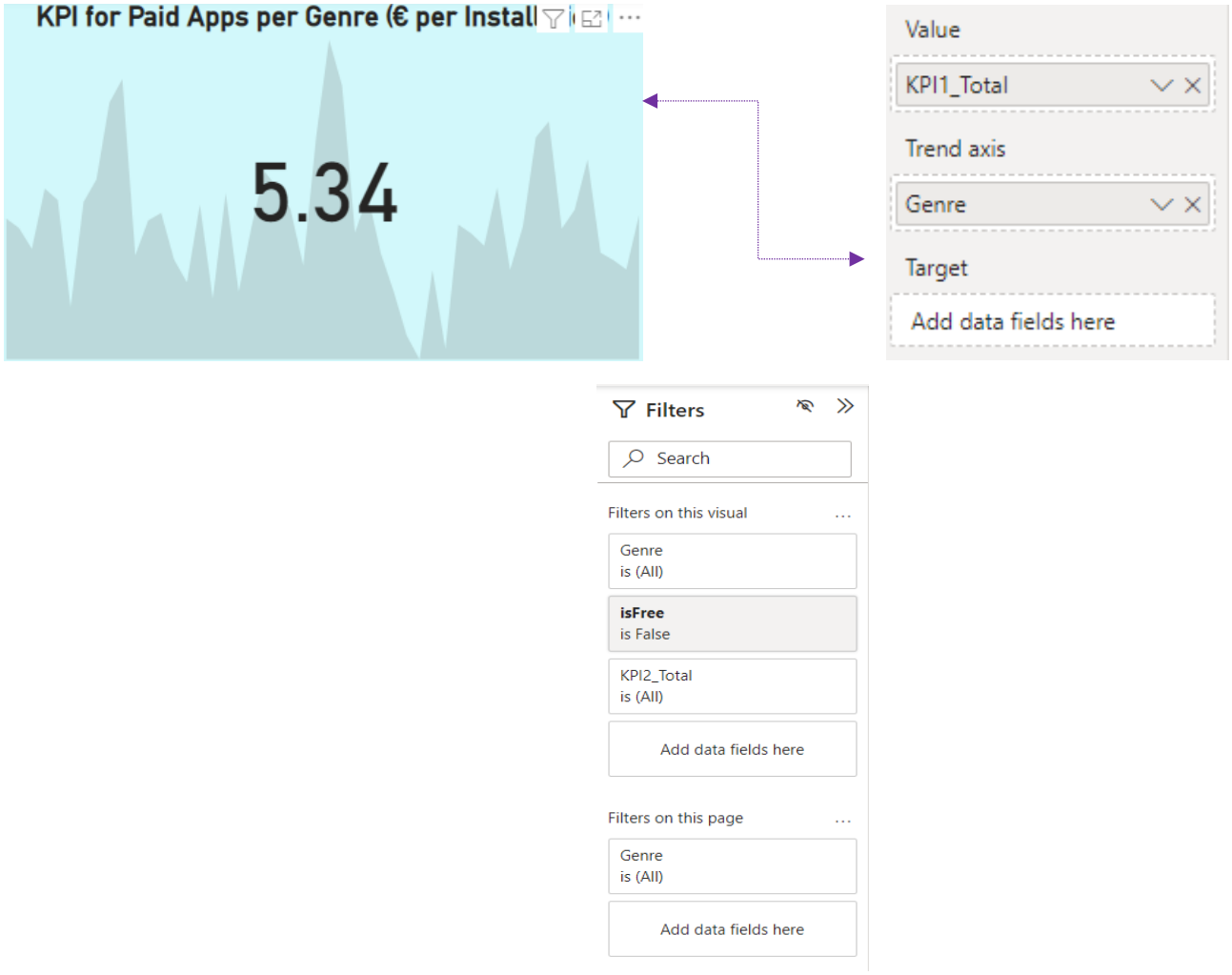
```
KPI1_Total =
DIVIDE(
    ([Avg_Price x Total Installations] + [Inst_App_Purch]),
    SUMX(FILTER( 'appfinalmaybe', 'appfinalmaybe'[isFree]=TRUE()), 'appfinalmaybe'[Total Installations])
)
```

Data Visualization 37: Measure for profitability KPI - for paid apps

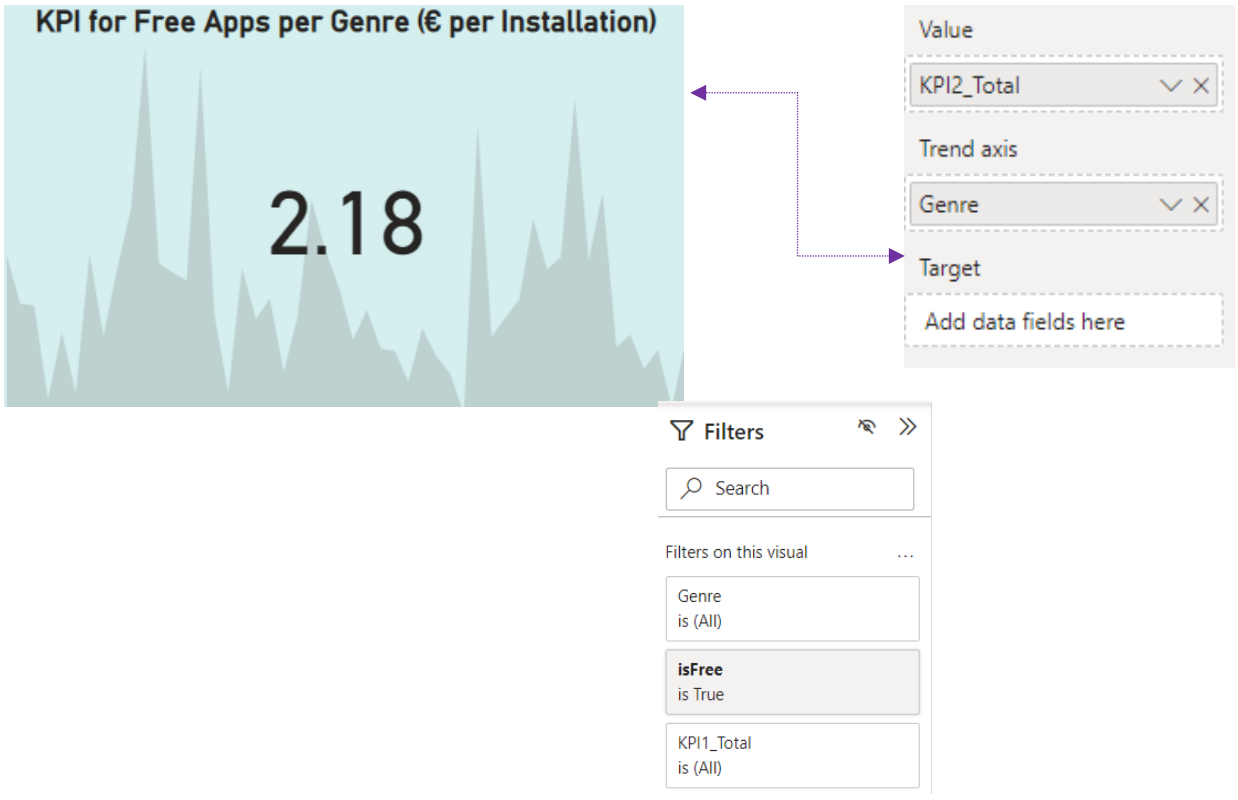
```
KPI2_Total =
DIVIDE(
    ([Avg_Price x Total Installations] + [Inst_App_Purch]),
    SUMX(FILTER( 'appfinalmaybe', 'appfinalmaybe'[isFree]=FALSE()), 'appfinalmaybe'[Total Installations])
)
```

Data Visualization 38: Measure for profitability KPI - for free apps

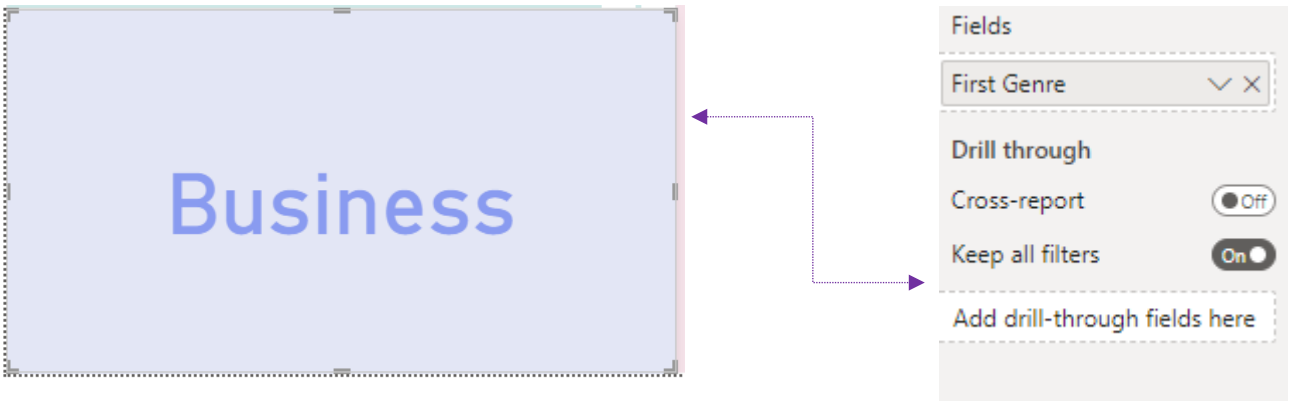
The KPIs are presented in single cards among with a single card presenting the genre's name. The dashboard present a bar chart that shows the total installation per genre in order to depict the popularity of each genre.



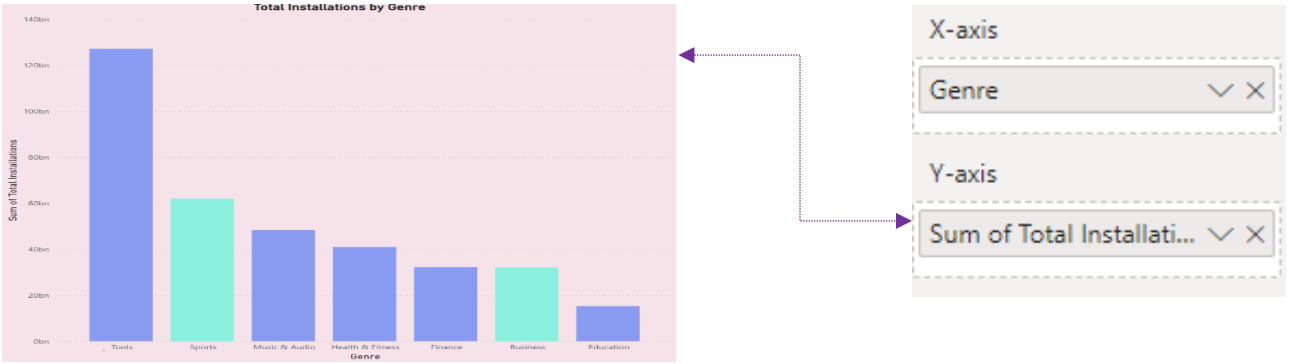
Data Visualization 39: Data fields & filtering



Data Visualization 40: Data fields & filtering

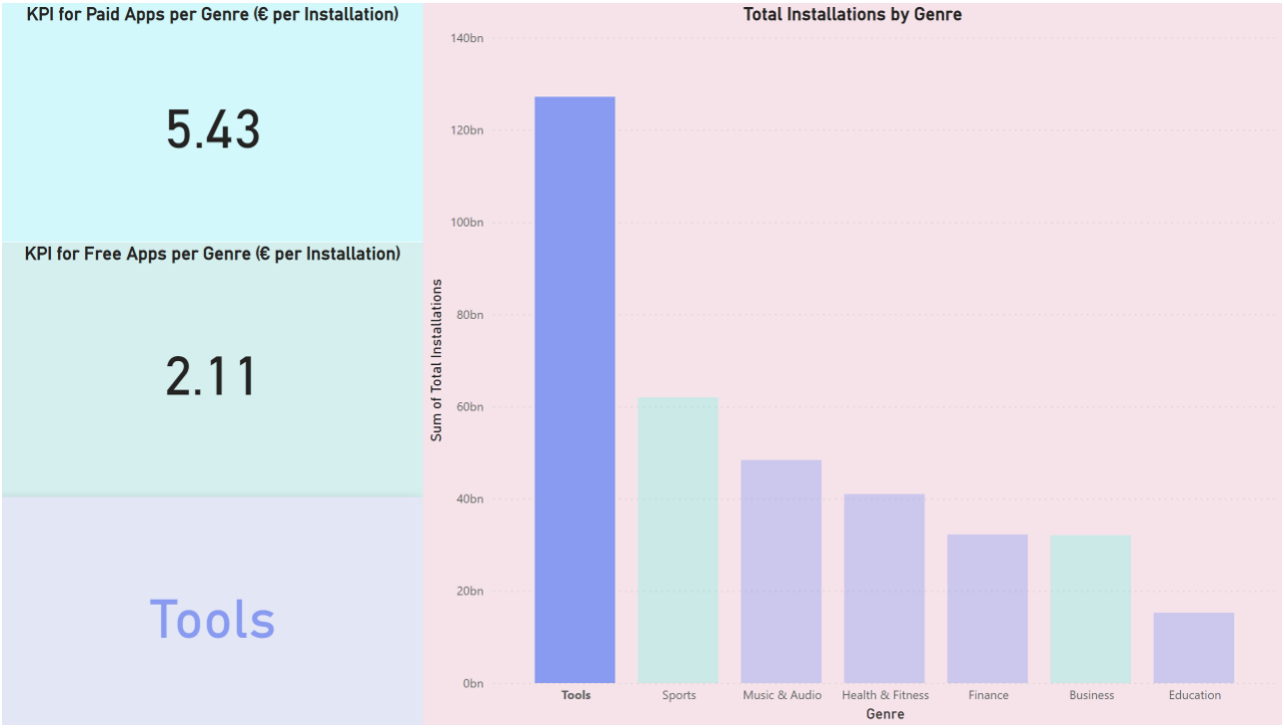


Data Visualization 41: Data fields



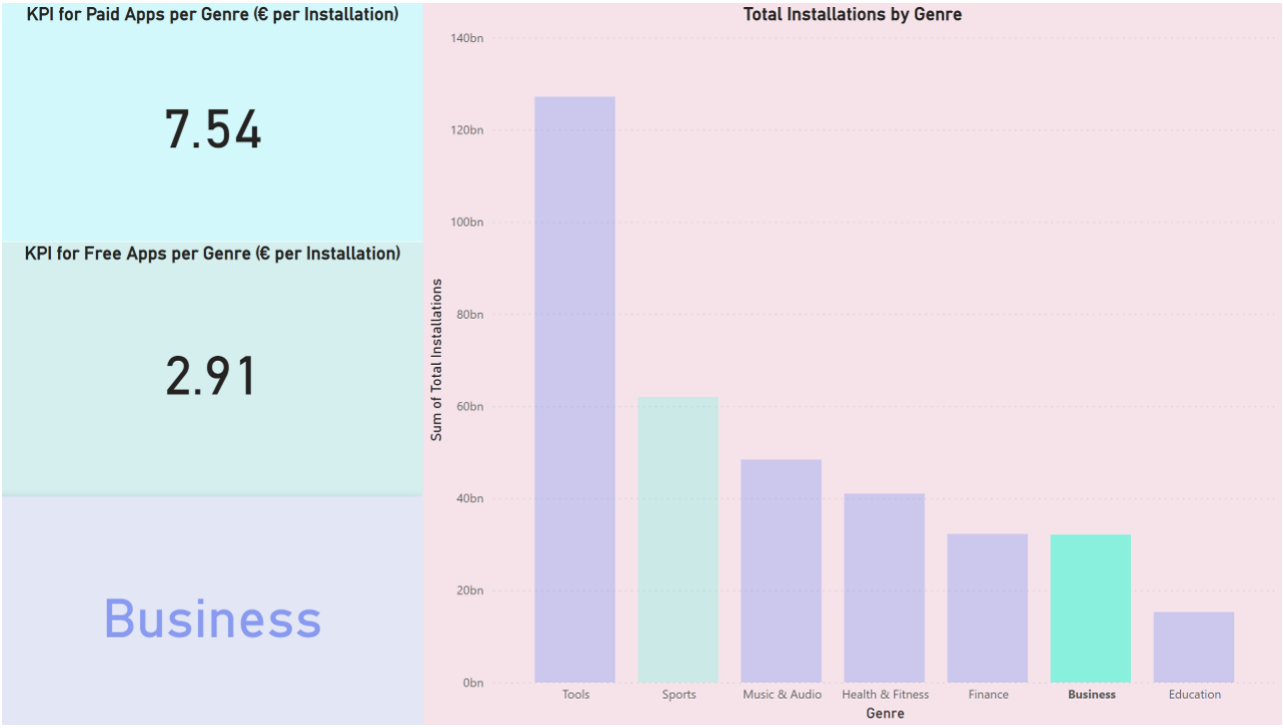
Data Visualization 42: Data fields

The overall dashboard contains the profit KPI for the paid and free apps along with the total installations of the particular genre. The free Tools profit KPI is 2.11 and for the paid apps is 5.43 while the total installations reach the 130bn installations and present by far the first genre in total installations. The Sports genre presents a profitability KPI that reaches the 2.91 euros per installation for the free apps and 7.17 euros for the non-free apps and almost 60bn total installations. And then the Business genre follows with the profitability KPI for free apps reaches the 2.98 and 7.54 for the non-free apps and almost 38 bn installations.

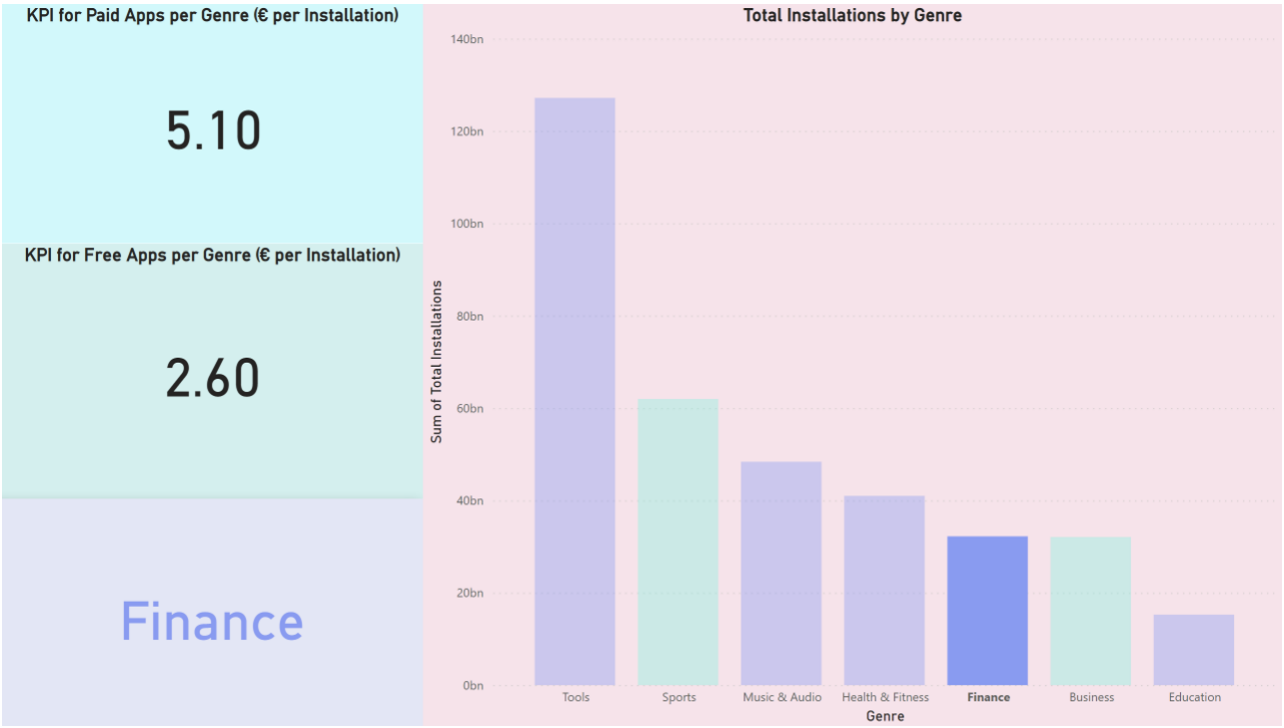


Data Visualization 43: Fifth report of Profitability KPI - Tools genre

Data Visualization 44: Fifth report of Profitability KPI - Sports genre

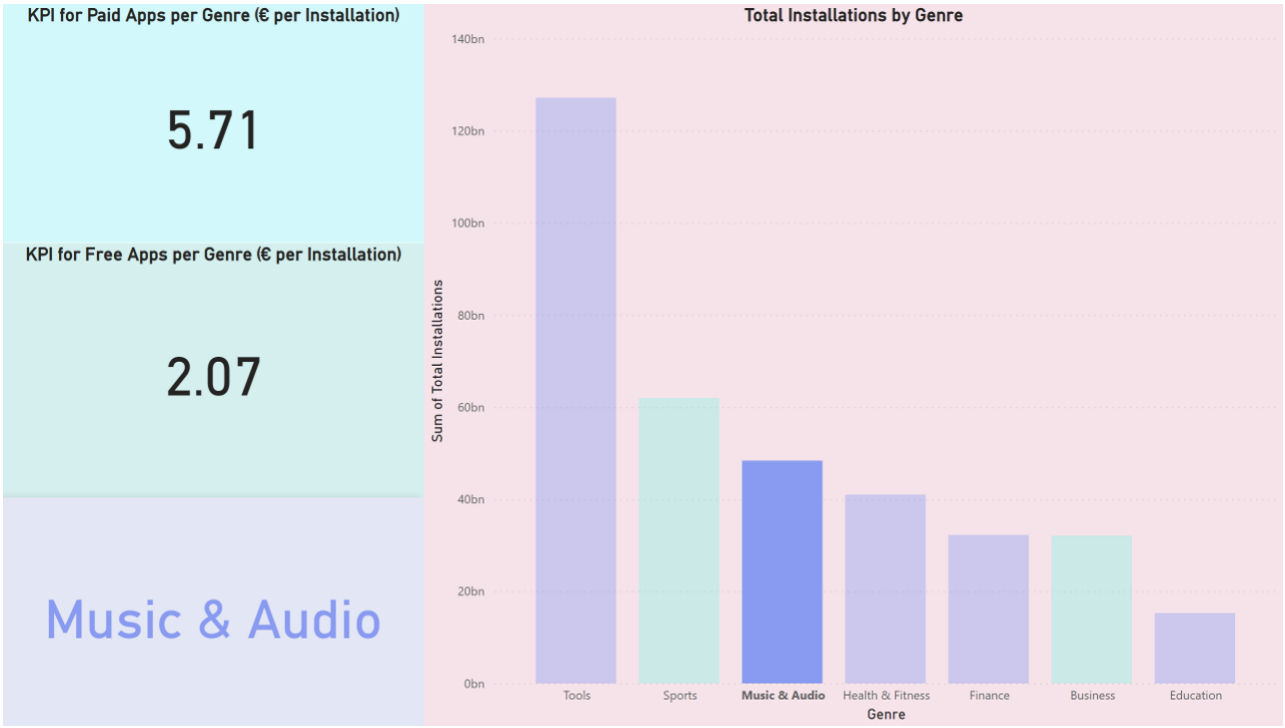


Data Visualization 45: Fifth report of Profitability KPI - Business genre



Data Visualization 46: Fifth report of Profitability KPI - Finance genre

Data Visualization 47: Fifth report of Profitability KPI - Health & Fitness genre



Data Visualization 48: Fifth report of Profitability KPI - Music & Audio genre

Summary of the Analysis

After taking into deeply consideration all the aforementioned reports and the overall information extracted from them the conclusion that is reached is following:

- The genres of Tools, Sports, Education, Music & Audio, Health & Fitness, Business and Finance are in the first ten genres in total installations and in total number of reviews among 50 distinct genres. The Finance genre is far behind in both installations and reviews so this is an indicator that the popularity and usage of this genre is not remarkable.
- Regarding the applications' ratings we found out that all seven understudy genres are rated with 4.2 out of 5 points and thus they are application genres are positively reviewed for the value they add to their actual users, so they are genres of public interest.
- All the seven genres are actually rated in the majority of their applications with five stars out five with substantial difference from the number of applications rated with four stars out of five. Additionally, the vast majority of applications in all genres are content rating for the age above 3 years old that actually contain all age groups possible users without any users quantitative limitations because of age constraints.
- The percentages of free and non-free applications of each genre substantially differ with the vast majority of applications having a free installations status. For this reason, the possible profitability of for free and non-free apps are examined separately.
- For non-free apps the Business and Sports genres present the highest installation prices, and the Tools and Sports genres present the highest installations profit as they are the dominant genres in terms of total installations number.
- For the free applications in each genre the fundamental profit source is the in app purchases either the smallest priced additional features that an application can offer to its users or the highest prices and premium additional functionalities an application offers to its users. Tools and Sports genres have the highest profits in in app purchases as expected because of their big number of installations.
- From the fourth dashboard we see that for the non-free apps the Education and Business genres are interesting options as the profits from basic installations and from in app purchases are very close so they are application genres that in their basic version offer enough functionalities that the users pay for them so they contain applications of with specified services in the respecting fields.
- In conclusion the profitability KPI emerges the Sports and Business genres as the most profitable ones in both free and non-free categories and the finally proposed one is the Sports genre as it is higher than the Business one in total installations and reviews and it is a genre that generally addresses in big and general portion of users.

References

1. Kaggle (last indexed 22/11/2022)
2. Google Play Store (last indexed 24/11/2022)
3. Microsoft SQL Server Management Studio (SSMS) (last indexed 26/11/2022)
4. Microsoft SQL Server Integration Services (SSIS) (last indexed 26/11/2022)
5. Microsoft SQL Server Analysis Services (SSAS) (last indexed 26/11/2022)
6. Microsoft Visual Studio (last indexed 28/11/2022)
7. Microsoft Excel (last indexed 26/11/2022)
8. Microsoft PowerPoint (last indexed 05/12/2022)
9. Microsoft Power BI (last indexed 04/12/2022)
10. Python (last indexed 23/11/2022)
11. Python's library Pandas (last indexed 23/11/2022)
12. R Programming language (last indexed 23/11/2022)
13. R's library Tidyverse (last indexed 29/11/2022)
14. App Revenue Statistics 2022 - Zippia.com (last indexed 29/11/2022)