Mendel University in Brno

BUSINESS INTELLIGENCE AND DATA WAREHOUSE

Project:
Olympics Data Analysis

Group Members
Roshah Charisma Selasie Nomo
Eugene Adu-Gyamfi
Enoch Kpoti

INTRODUCTION

Business intelligence (BI) technology, procedures, and methods are used to convert unprocessed data into meaningful and practical information for business decision-making. It often uses tools and techniques, including data warehousing, data mining, and reporting to analyse and visualise data from various sources, including transactional databases and log files.

In the context of the Olympic Games, we used the data warehouse and business intelligence tools to store and organise the data, as well as do research and generate reports and visualisations to analyze trends and patterns in the data.

GOAL

To analyse the performance of athletes and countries and to compare the success of different countries to their economic power (GDP).

RESOURCES

Access to the completed work can be found in the google drive link here:
https://drive.google.com/file/d/1seitJs01SyeAeQUAMmmmUbmQwrKPEijj/view?usp=share_link

DATASETS

The datasets were taken from Kaggle; the main dataset is scrapped from http://www.olympedia.org/ and is in CSV format. It contains all details about the Olympics- the athlete bio, events and results about an athlete, games, medals and countries. The additional dataset contains population, yearly change and median age, region, and the GDP for each country which are also in CSV format.

Main Dataset –

Olympic Games Results from Athens 1896 to Beijing 2022-https://www.kaggle.com/datasets/josephcheng123456/olympic-historical-dataset-from-olympediaorg?select=Olympics_Games.csv

Additional Dataset-

Countries and their population-https://www.kaggle.com/datasets/muhammedtausif/world-population-by-countries

All the countries with their regions-https://www.kaggle.com/datasets/andreshg/countries-iso-codes-continent-flags-url

Country GDP in US dollars- https://www.kaggle.com/datasets/tmishinev/world-country-gdp-19602021

DATASET STRUCTURE

The dataset contains the following:

- Athlete id – primary key, numeric (e.g **5085**)
- Start Time – date the event started (e.g. 1896-04-06)
- End Time – date the event ended (e.g., 1896-04-15)
- Country- nvarchar (255) the name of a country athlete, is from
- Event - nvarchar (255), a specific event under a particular sport (eg: event title: triple jump, men, sport: athletics)

KPI

Analysis of the data can be used to:

- The number of medals won by a country in summer
- The number of medals won by a country in winter.
- The number of medals won per country. (Gold, bronze, or silver)
- The highest number of medals won by an athlete for a sport
- The average number of countries that participated per year.
- Correlation between a country's GDP and the number of medals won.
- Correlation between a country's population, median age and the number of medals won
- Height of the athletes and the high jump event
- Correlation between age and number of medals won by an athlete

METHODOLOGY

The tools we used were Draw.io for visual design of the dimensional model, Microsoft SQL management studio for the database, Microsoft visual studio to extract, transform and load(ETL) the data and Powerbi for data visualisation.

Dimensional model

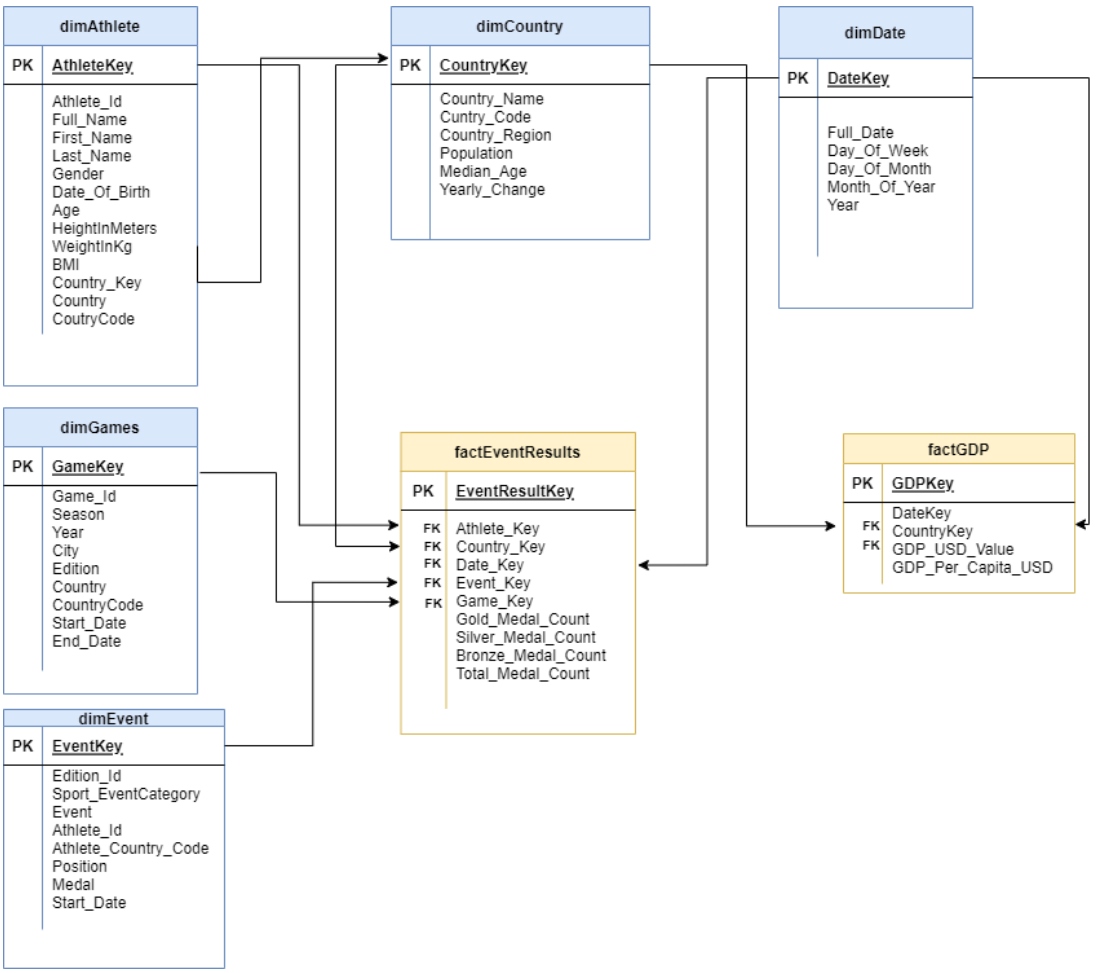


Table = dimAthelete

DIMATHELETE					
Key	Name	Data Type	Null	Attribute	Description
1. PK	AthleteKey	int		Identity	The primary key for the dimAthlete
2.	Athlete_Id	int	✔		The ID of the athlete (“136329”)
3.	Full_Name	Nvarchar(255)	✔		Full name of athlete
4.	FirstName	nvarchar(255)	✔		The first name of the athlete - “Gabriela”
5.	LastName	nvarchar(255)	✔		The last name of the athlete – “Petrova”
6.	Date_Of_birth	date	✔		The date of birth of the athlete- MM-DD-YYYY (“9/22/1966”)
7.	Age	int	✔		Athletes age -“27”
8.	Gender	nvarchar(255)	✔		The gender of the athlete (“male”)
9.	HeightInMeters	float			The height of the athlete - (“203”)
10.	WeightInKg	float	✔		The weight of the athlete - (“100”)

11.	BMI	float	✔		The body mass index of the athlete – (“24.3”)
12.	Country_Key	int			Country key from country table-(1)
13.	Country	nvarchar(255)	✔		The country where the athlete is from – (“Germany”)
14.	CountryCoode	nvarchar(255)	✔		The country code of where the athlete comes from – (“GER”)

Table = dimCountry

DIMCountry					
Key	Name	Data Type	Null	Attribute	Description
1. PK	CountryKey	Int		Identity	The primary key for the dimCountry
2.	CountryName	nvarchar(255)	✔		The name of a country – (“Germany”)
3.	CountryCode	nvarchar(255)	✔		The code of the country – (“GER”)
4.	CountryRegion	nvarchar(255)	✔		The region where the country is located - (“WESTERN EUROPE ”)
5.	Population	float	✔		The population of the country – (“12,127,071”)
6.	Median_Age	float	✔		Median age of the country (“27”)
7.	Yearly Change	float	✔		Yearly Change in Population per country in percentage (1.07%)

Table = dimGames

DIMGAMES					
Key	Name	Data Type	Null	Attribute	Description
1. PK	GameKey	int		Identity	The primary key for the dimGames
2.	GameID	int	✔		The ID of the games (“8”)
3.	Season	nvarchar(255)	✔		The season in which the game was held – “Summer”
4.	Year	int	✔		The year in which the game was held - “1997”
5.	City	nvarchar(255)	✔		The city in which the game was held – “Helsinki”
6.	Edition	nvarchar(255)	✔		The edition of the game – “1972 Summer Olympics”
7.	Country	nvarchar(255)	✔		The country where the game was held – “Finland”
8.	Country_code	nvarchar(255)	✔		The country code – “FIN”
9.	StartDate	date	✔		The starting date of the olympics – “7-29-1948”
10.	EndDate	date	✔		The ending date of the olympics – “8-14-1948”

Table = dimDate

DIMDATE					
Key	Name	Data Type	Null	Attribute	Description
1. PK	DateKey	int		Identity	The primary key for the dimDate
2.	Full_date	date			MM-DD-YYYY – 1896-01-01
3.	Day_of_week	varchar(9)			Day of the week- "Wednesday"
4.	Day_of_month	int			Day number in month- "1"
5.	Month_of_year	Varchar(9)			Month_of year – "January"
6.	year	int			Year -1896

Table = dimEvent

DIMEVENT					
Key	Name	Data Type	Null	Attribute	Description
1. PK	EventKey	int		Identity	The primary key for the dimEvent
2.	Edition_id	int	✔		The ID of the event "4"
3.	Sport_EventCategory	nvarchar(255)	✔		The category of the event "Athletics"
4.	Event	nvarchar(255)	✔		The name of the event "100 metres, Men"
5.	Athlete_id	int	✔		The ID of the athlete "56265"
6.	Athlete_Country_code	nvarchar(255)	✔		Athlete country code"GER"
7.	Position	nvarchar(255)	✔		The athlete position during the event "1"
8.	Medal	nvarchar(255)	✔		The medal the athlete won "Silver"
9.	Start_Date	date	✔		Start date of event

fact = factEventResults

FACT EVENTRESULTS					
Key	Name	Data Type	Null	Attribute	Description
1. PK	AthleteKey	Int		Identity	Primary key of the factEventResults
2. FK	CountryKey	Int			Primary key of the country dimension
3. FK	GameKey	Int			Primary key of the game dimension

4.	FK	DateKey	Int			Primary key of the date dimension
5.	FK	EventKey	Int			Primary key of the event dimension
6.		Gold_Medal_Count	Int	✔		Number of gold medals won
7.		Silver_ Medal_Count	Int	✔		Number of gold medals won by a country
8.		Bronze_ Medal_Count	Int	✔		Number of silver medals won by a country
9.		Total_ Medal_Count	int	✔		Number of bronze medals won by a country

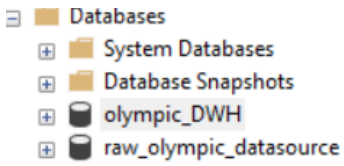
Table = factGDP

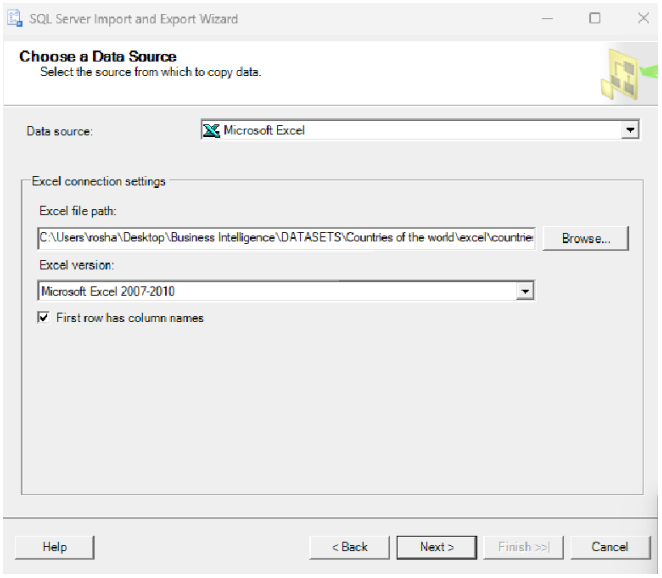
NOTE: Table below will contain GDP from 1960-2020. This is only a summary. See the dimensional model diagram for more information.

FACT GDP						
Key		Name	Data Type	Null	Attribute	Description
1.	PK	GDPKey	int		Identity	The primary key for the factGDP
2.	FK	CountryKey	int			Primary key of the country dimension
3.	FK	DateKey	int			Primary key of the date dimension
4.		GDPValue	float	✔		A country's GDP relating to a particular year
5.		GDP_per_capita_USD		✔		The gdp per capita of a country

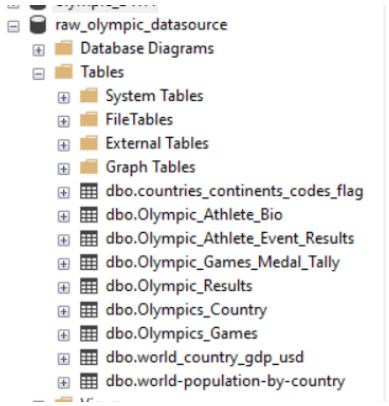
Importing the data and database design

We created a source and destination database in Microsoft management studio and uploaded the excel files as into the source database

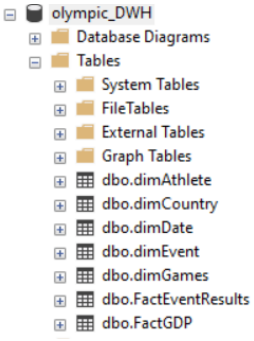




Source database tables

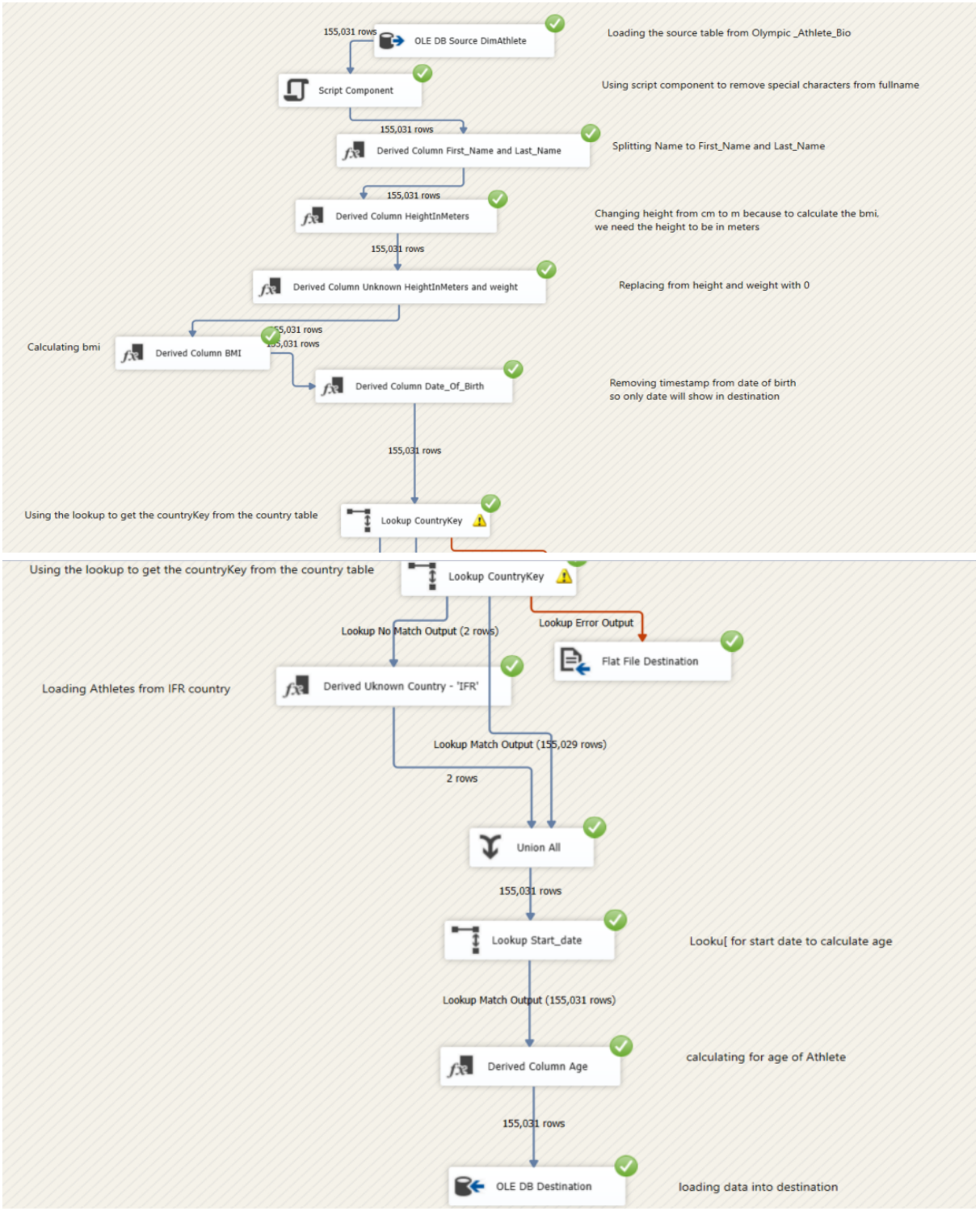


Next, we used the source tables to create scripts for our destination database, using the dimensional model tables as a guide for naming the attributes.



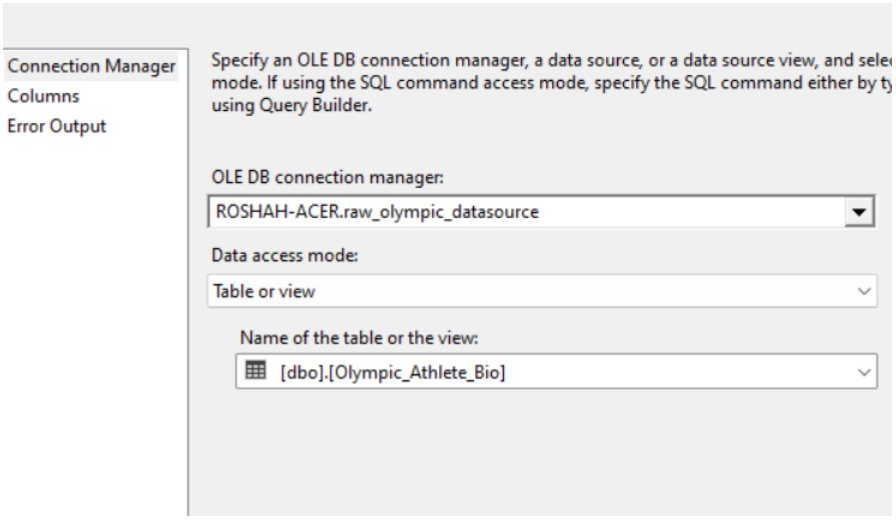
EXTRACTION, TRANSFORMATION AND LOADING (ETL)

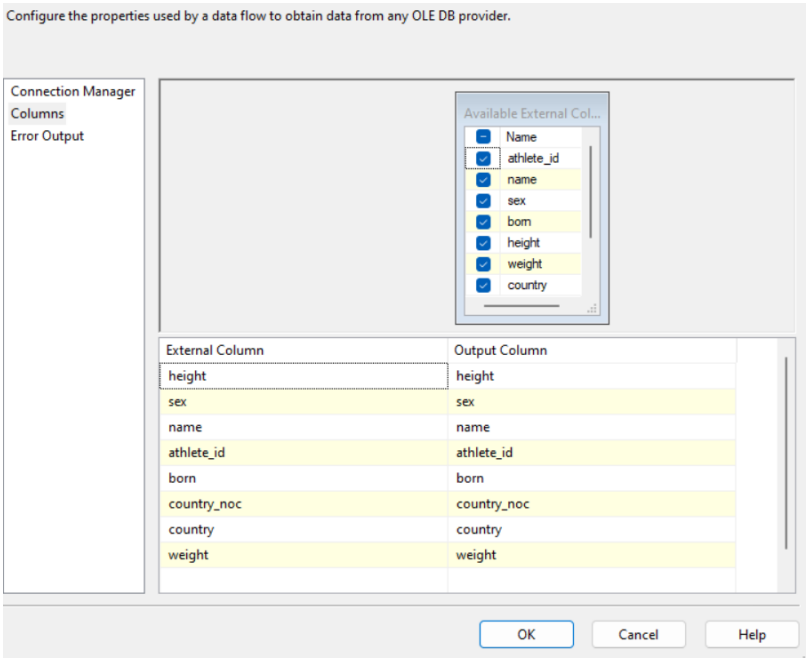
DIM ATHLETE



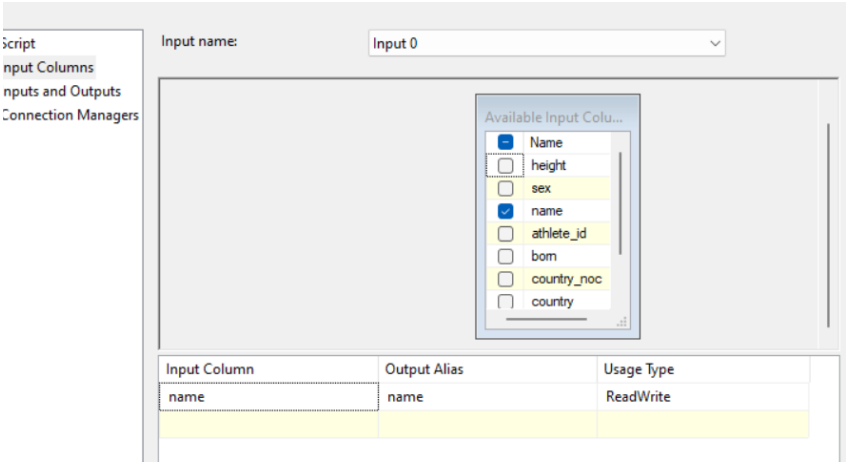
We first used the data flow component. Then used the OLEDB source component to connect to our source data(raw_olympic_datasource) and then selected the source table (raw _olympic_datasource) and added

all necessary mappings needed as shown below:





Next, we noticed the source data had special characters in the name table, so we had to take those out. We used the script component to take out these characters.



```
public override void Input0_ProcessInputRow(Input0Buffer Row)
{
    Row.name = RemoveSpecialCharacters(Row.name);


}

1 reference
public static string RemoveSpecialCharacters(string str)
{
    return Regex.Replace(str, "[^a-zA-Z_ -]+", "", RegexOptions.Compiled);
}
```

The "RemoveSpecialCharacters" method accepts as input a string that is the name property of the input buffer. A regular expression is used within the procedure to match any character that is not a letter, underscore, space, or hyphen. The characters that have been matched are then replaced with an empty string, thereby deleting them from the input string.
We added a derived component to split the name from the data source into First_Name and Last_Name in our destination table

Ref:
https://social.msdn.microsoft.com/Forums/sqlserver/en-US/8bfdd30a-526a-435e-b915-0f86645d0496/split-the-names-in-derived-column-in-ssis?forum=sqlintegrationservices

The height in the destination source was in centimetres, to get the BMI, we converted the height to meters using a derived component and used another derived component to obtain the value of the BMI

Derived Column Name	Derived Column	Expression	Data Type	Le
HeightInMeters	<add as new column>	(height / 100)	double-precision float ...	

Derived Column Name	Derived Column	Expression	Data Type
BMI	<add as new column>	(ISNULL(weight) || ISNULL(HeightInMeters) || (weight == 0) || (HeightInMeters == 0) ? (DT_R8)0 : ROUND((weight / ((HeightInMeters) * (HeightInMeters))),2))	double-precision float [DT_R8]

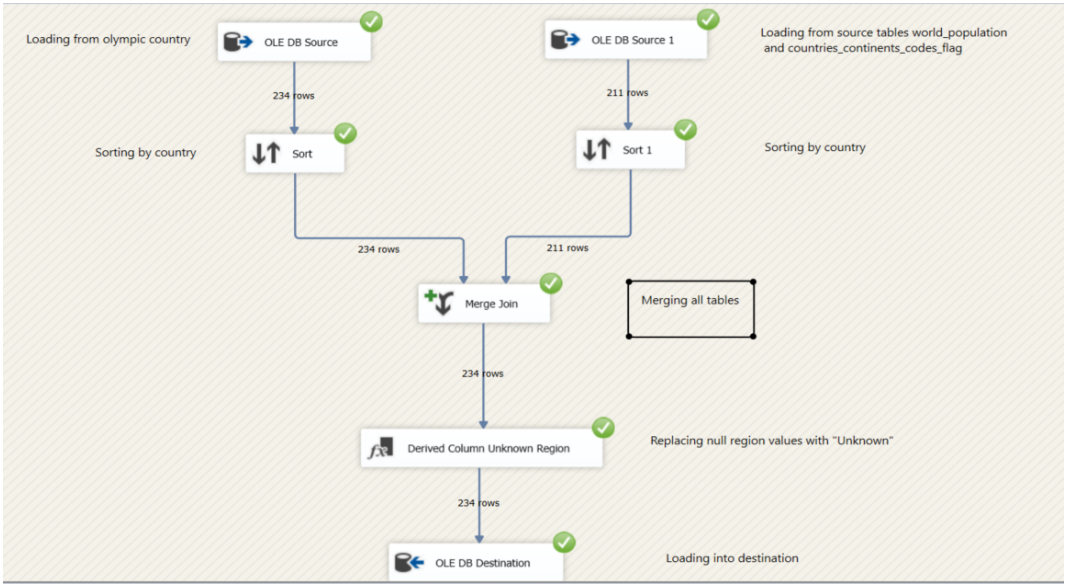
Also, we noticed the date in the destination database had a time stamp. Since our destination data needed only the date, we took out the time stamp and maintained only the date using a derived column

Derived Column Name	Derived Column	Expression	Data Type	Le
Date_Of_Birth	<add as new column>	(DT_DATE)((DT_DBTIMESTAMP)born)	date [DT_DATE]	

Next, we did a lookup into the country table to get the country key
However, after that process, two of the athletes did not have any country from the country table, so we used a derived column to redirect them into a union all component and added a flat file destination to handle the lookup error output.

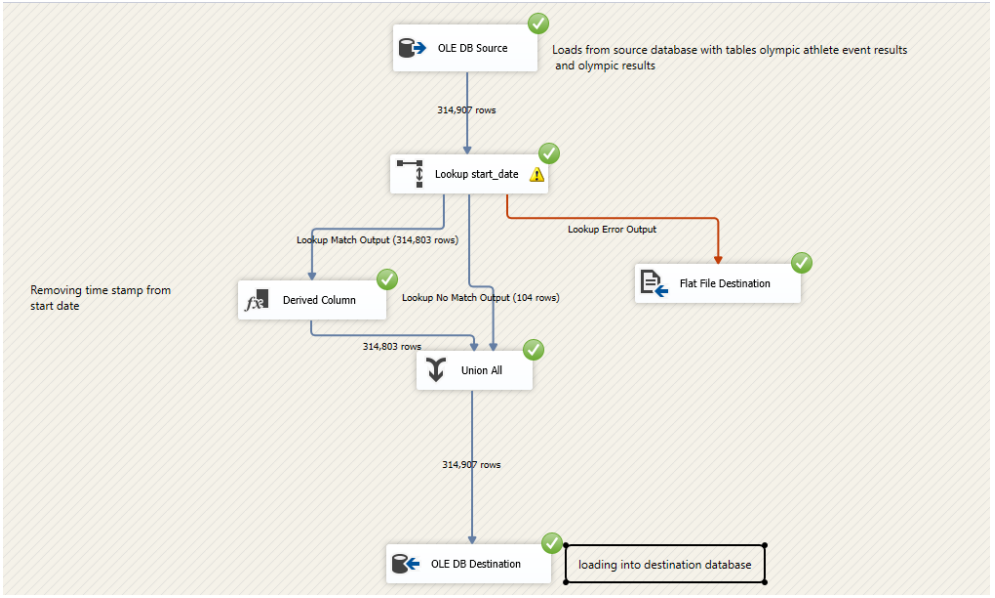
DimCountry

To load the country table, three source tables were used. The Olympic_country table, the world_population by country table and the country_flags table. We need the countries from the first table, the population, median age and yearly change from the second table and the region from the third table.We used the sort component to sort the data by country and then the merge join component to load the data. Since we mainly needed the countries from the Olympic_country tables but conne



DimEvent

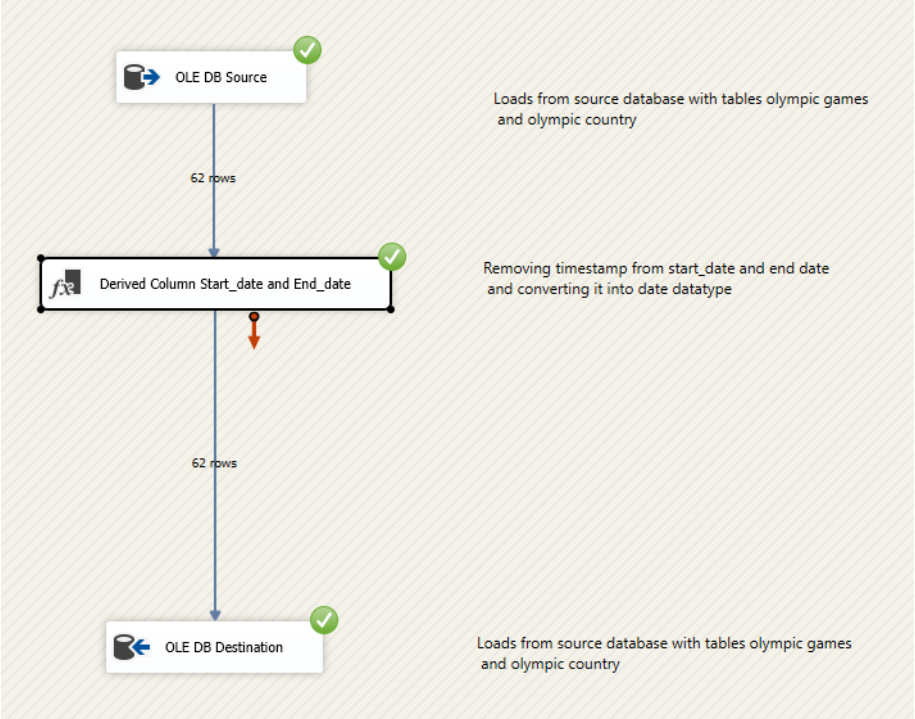
This table contains all sporting events and other details that took place at the event, as well as which medals an athlete won. The timestamp was also removed because we only needed to use the date for our analysis. The start date here refers to when the event competition began

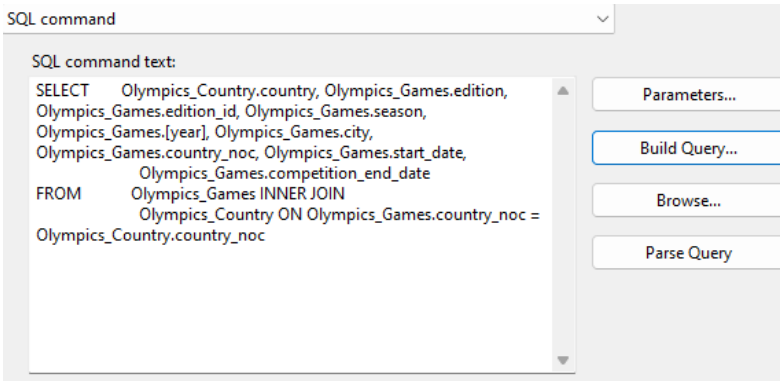


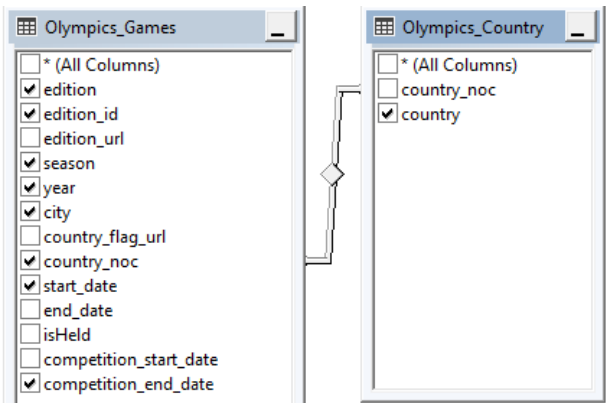
Derived Column Name	Derived Column	Expression	Data Type	L
start_date	Replace 'start_date'	(DT_DATE)((DT_DBTIMESTAMP)start_date)	database date [DT_D...	

DimGames

To load the dimGames table, the used the sql command and the build a query option to pick the details we needed to fill the destination table. Here, we combined two tables from the source database. ie: Olympic games and oympic country. The start and end date here refers to when the Olympic games started.The timestamp was also taken out,







Derived Column Name	Derived Column	Expression	Data Type	L
Start_date	<add as new column>	(DT_DATE)((DT_DBTIMESTAMP)start_date)	date [DT_DATE]	
End_date	<add as new column>	(DT_DATE)((DT_DBTIMESTAMP)competition_end_da...	date [DT_DATE]	

DimDate

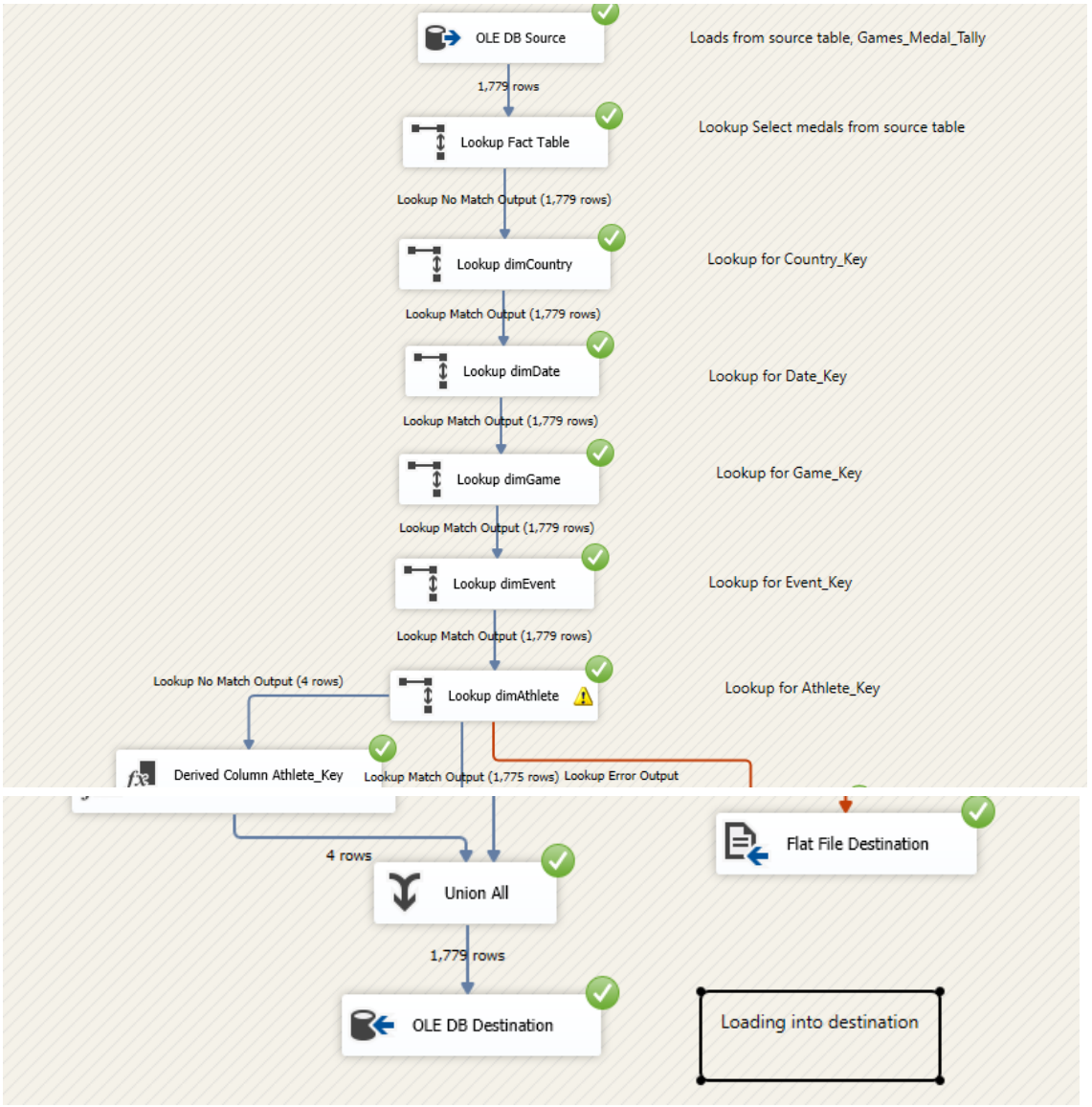
For the date dimension, we wrote an SQL script which helped to place the required values into the database

```sql
DECLARE @start_date DATE = '1896-01-01';
DECLARE @end_date DATE = '2022-12-31';

-- Create the table structure for the date dimension
CREATE TABLE dimDate (
    DateKey INT PRIMARY KEY,
    Full_date DATE NOT NULL,
    Day_of_week VARCHAR(9) NOT NULL,
    Day_of_month INT NOT NULL,
    Month_of_year VARCHAR(9) NOT NULL,
    year INT NOT NULL
);

-- Declare a loop variable and set it to the start date
DECLARE @current_date DATE = @start_date;

-- Loop through the dates from the start date to the end date
WHILE @current_date <= @end_date
BEGIN
    -- Insert a row into the date dimension table for the current date
    INSERT INTO dimDate (DateKey, Full_date, Day_of_week, Day_of_month, Month_of_year, year)
    SELECT
        DATEDIFF(day, @start_date, @current_date) + 1, -- Generate a unique ID for the date
        @current_date,
        DATENAME(weekday, @current_date), -- Get the day of the week
        DAY(@current_date), -- Get the day of the month
        DATENAME(month, @current_date), -- Get the month of the year
        YEAR(@current_date) -- Get the year
    ;

    -- Increment the loop variable to the next date
    SET @current_date = DATEADD(day, 1, @current_date);
END;
```

Source: https://drive.google.com/drive/folders/11LIhwr51sLowlt9KxVoNum93Zpwgliwx

FACT EVENTRESULTS

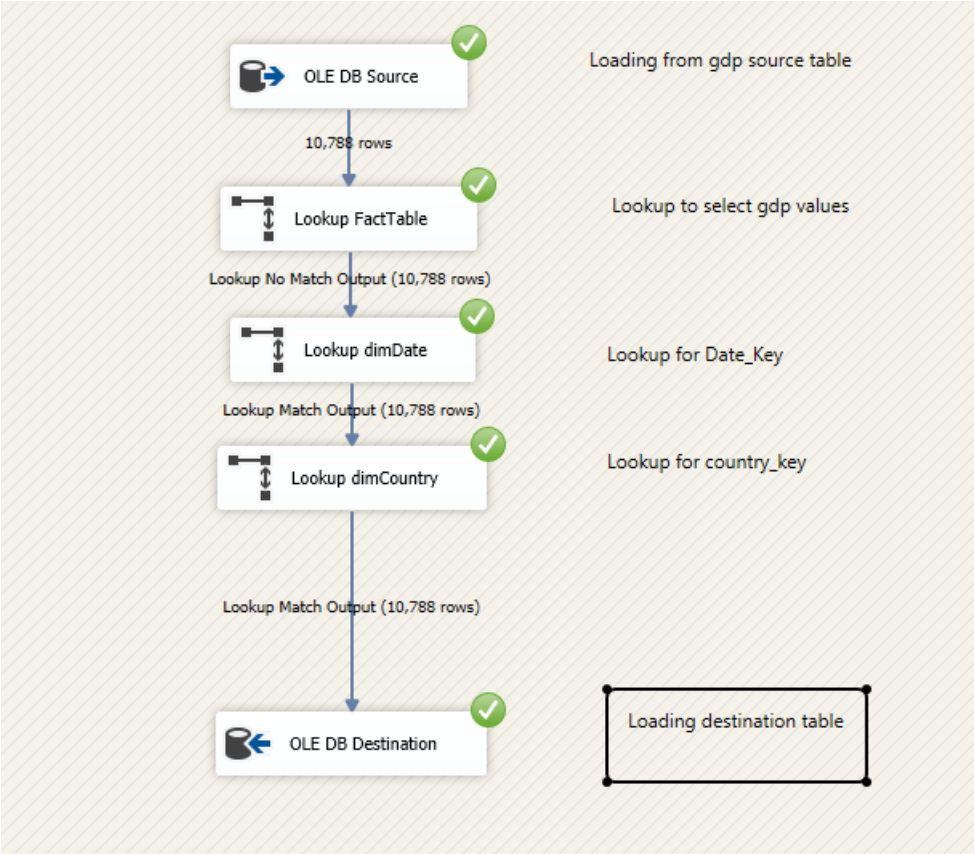
In this table, we used the lookup into all our dimensional tables and also added the medals received by each country. We noticed that four records were unable to enter the destination database. This happened because the four did not have athlete ids and since we were making a lookup into the athlete table, there was no match. We use the derived column to provide athlete keys for this table




```
1896 Summer Olympics,1,1896,Mixed team,MIX,1,0,1,2,130,32
1900 Summer Olympics,2,1900,Mixed team,MIX,1,2,3,6,130,17
1904 Summer Olympics,3,1904,Mixed team,MIX,1,0,0,1,130,56
1924 Winter Olympics,29,1924,Mixed team,MIX,1,0,0,1,130,16
```

FACTGDP

The ETL for the table below contains the gdp and gdp per capita for each country. We made a lookup into the dim date and dim country tables to obtain their keys

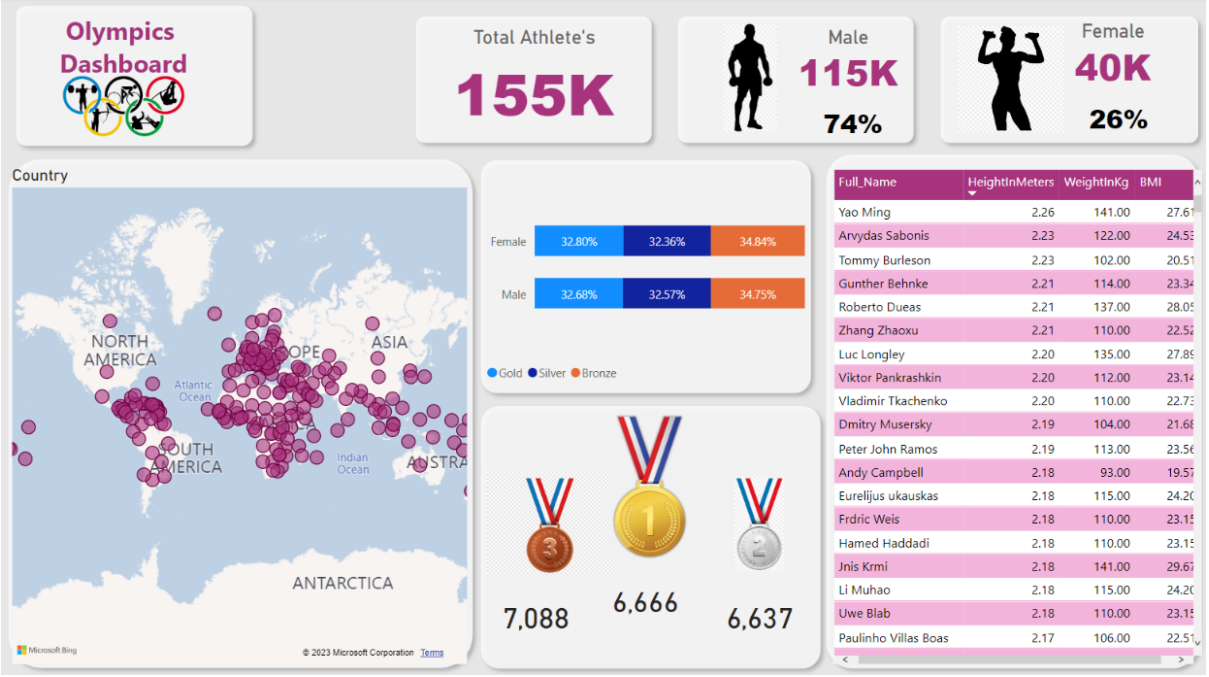


RESULTS

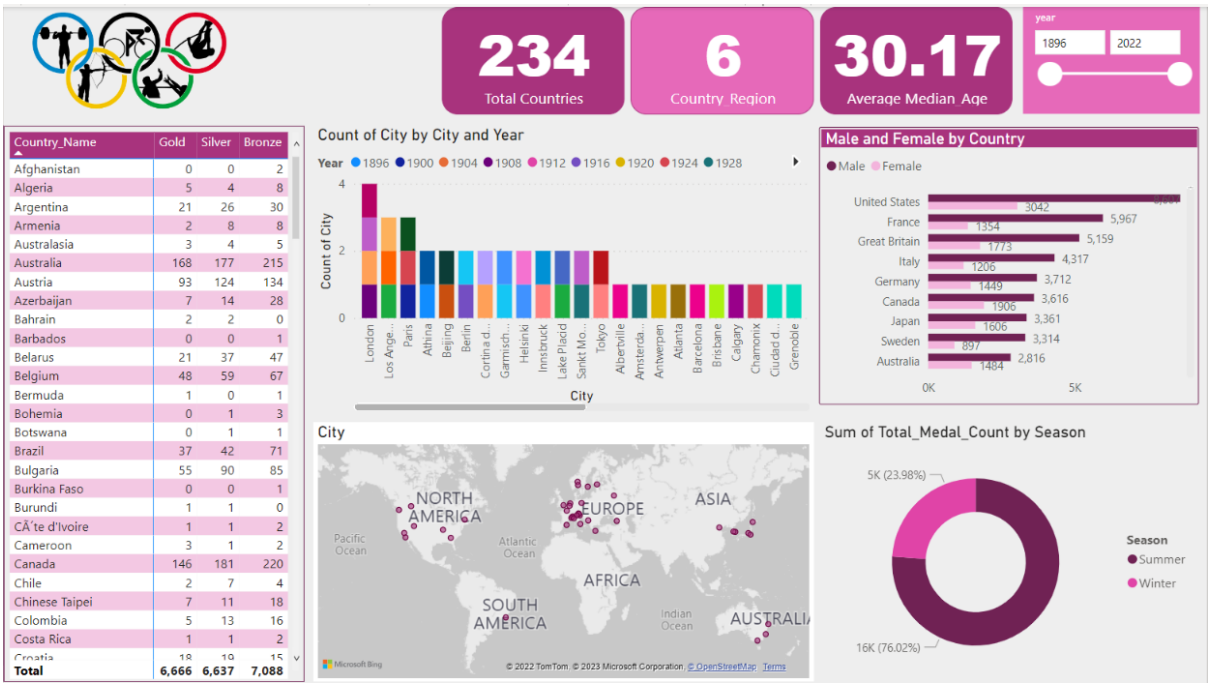
ANALYSIS AND DATA VISUALIZATION WITH POWER BI

From the diagrams below, we can interpret and analyze the data we loaded.

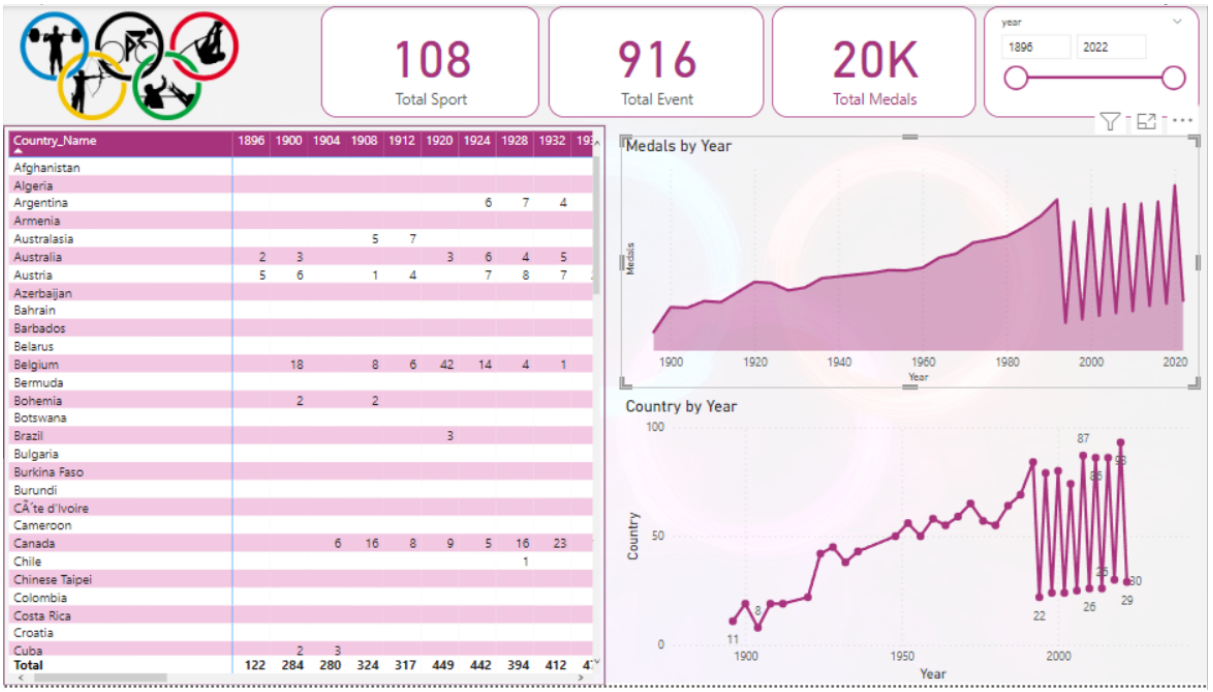
The image below gives a general overview of the data we have. Here, we can see the total number of athlete, the number of females and males, as well as the percentage, the countries the athletes belong to, the number of gold, bronze and silver medals per country And the percentage of percentage of medals obtained per male and female. We are also able to tell which athletes are the tallest.



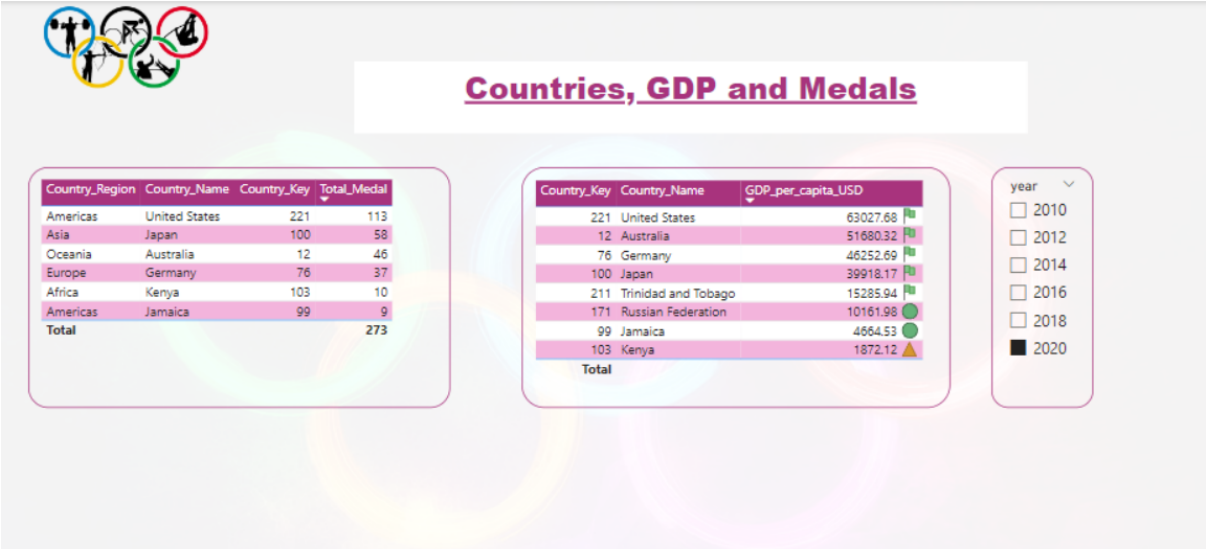
This table the diagram below also gives a general overview but also has two of the kpis.i.e the number of medals percountry and in summer. The data shows that more medals are usually won in summer thank in winter. From the image, we are also able to see the number of regions we have in the data, average median age per the countries, the total number of countries that have participated in the Olympics. Also, the countries that the Olympics have been played in and the number of times a city has hosted the Olympics. From this, we can tyell that London has hosted the Olympics more than any other city.



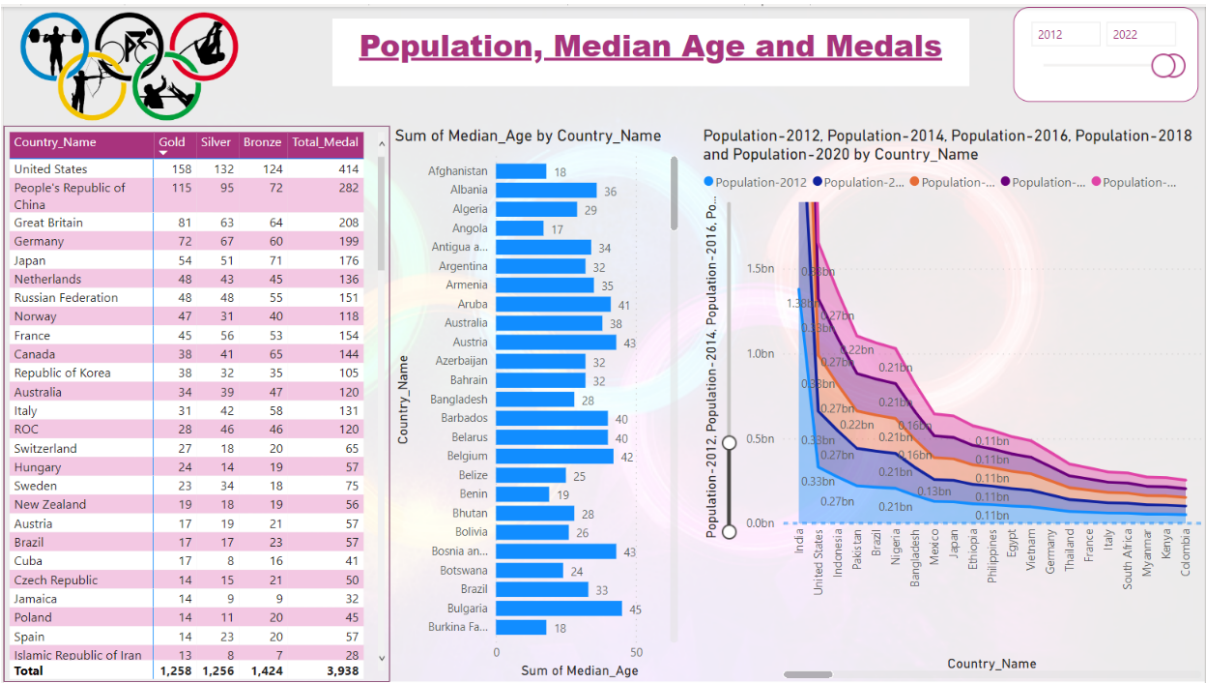
Another KPI in this diagram is the number of countries participating per year. We have a slider that makes it easier to analyse the years. The table on the left shows the number of medals presented to a country per year. The area chart shows the number of medals presented per year, and the line chart shows the number of countries that competed. From the diagrams below, the number of medals reduced in 1994. Still, it increased in 1996, and the inconsistency continues, however, to understand this, we see that the number of countries that competed could have been more consistent. This informs us that the medals increased when the countries increased and decreased just as the countries did.



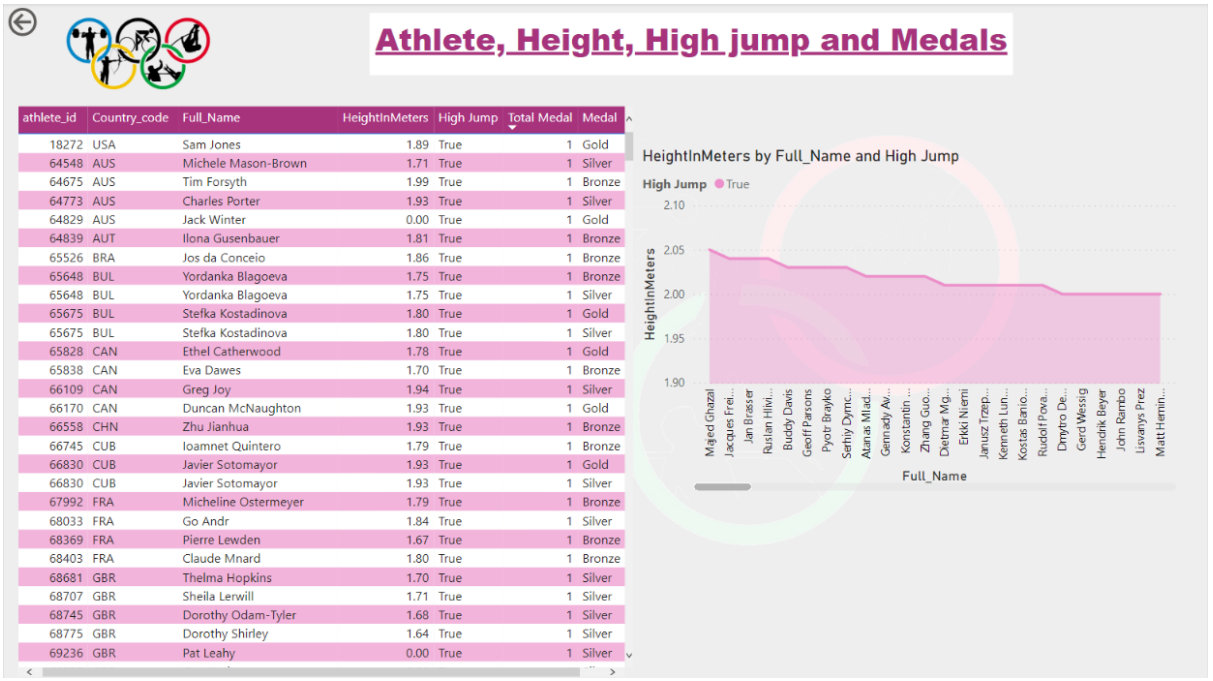
The table below also shows the countries and their GDP. We are able to analyse whether a country with a high gdp would win more medals than a country with a low one. Also, we chose to pick one country from each continent to help with this analysis



We were also able to analyse the influence of a country's population and median age against the number of medals won. For instance, a country like India has a very high population and would therefore be able to present many athletes, however, they do not have many medals. This means quantity doesn't always mean winning more medals, but the quality of the athletes you present as a country.



Also, we analysed the high jump and realised that not all the tallest athletes can win MUCH.



athlete_id	Country_code	Full_Name	HeightInMeters	High Jump	Total Medal	Medal
18272	USA	Sam Jones	1.89	True	1	Gold
64548	AUS	Michele Mason-Brown	1.71	True	1	Silver
64675	AUS	Tim Forsyth	1.99	True	1	Bronze
64773	AUS	Charles Porter	1.93	True	1	Silver
64829	AUS	Jack Winter	0.00	True	1	Gold
64839	AUT	Ilona Gusenbauer	1.81	True	1	Bronze
65526	BRA	Jos da Conceio	1.86	True	1	Bronze
65648	BUL	Yordanka Blagoeva	1.75	True	1	Bronze
65648	BUL	Yordanka Blagoeva	1.75	True	1	Silver
65675	BUL	Stefka Kostadinova	1.80	True	1	Gold
65675	BUL	Stefka Kostadinova	1.80	True	1	Silver
65828	CAN	Ethel Catherwood	1.78	True	1	Gold
65838	CAN	Eva Dawes	1.70	True	1	Bronze
66109	CAN	Greg Joy	1.94	True	1	Silver
66170	CAN	Duncan McNaughton	1.93	True	1	Gold
66558	CHN	Zhu Jianhua	1.93	True	1	Bronze
66745	CUB	Ioamnet Quintero	1.79	True	1	Bronze
66830	CUB	Javier Sotomayor	1.93	True	1	Gold
66830	CUB	Javier Sotomayor	1.93	True	1	Silver
67992	FRA	Micheline Ostermeyer	1.79	True	1	Bronze
68033	FRA	Go Andr	1.84	True	1	Silver
68369	FRA	Pierre Lewden	1.67	True	1	Bronze
68403	FRA	Claude Mnard	1.80	True	1	Bronze
68681	GBR	Thelma Hopkins	1.70	True	1	Silver
68707	GBR	Sheila Lerwill	1.71	True	1	Silver
68745	GBR	Dorothy Odam-Tyler	1.68	True	1	Silver
68775	GBR	Dorothy Shirley	1.64	True	1	Silver
69236	GBR	Pat Leahy	0.00	True	1	Silver

Finally, Micheal Phelps is one of the athletes winning the most medals in the Olympics, age is can be seen as a factor in his winning strike, the older, the more experience, the higher the chance of winning, however, this isn't always the case but the image below clearly shows that an average number of people above 24 can win more medals

Country	athlete_id	Full_Name	Medals	Sport_EventCategory	Age
United States	93860	Michael Phelps	28	Swimming	31
Soviet Union	29198	Larisa Latynina	18	Artistic Gymnastics	21
Soviet Union	31235	Nikolay Andrianov	15	Artistic Gymnastics	19
Norway	101008	Marit Bjrgen	15	Cross Country Skiing	37
Italy	22677	Edoardo Mangiarotti	13	Fencing	17
Japan	30611	Takashi Ono	13	Artistic Gymnastics	20
Soviet Union Ukraine	31267	Boris Shakhlin	13	Artistic Gymnastics	24
Norway	84154	Ole Einar Bjrndalen	13	Biathlon	20
Netherlands	109767	Ireen Wst	13	Speed Skating	19
East Germany Germany	9879	Birgit Fischer-Schmidt	12	Canoe Sprint	18
Germany	11503	Isabell Werth	12	Equestrian Dressage	23
Japan	30593	Sawao Kato	12	Artistic Gymnastics	22
Russian Federation	30978	Aleksey Nemov	12	Artistic Gymnastics	20
United States	51217	Jenny Thompson	12	Swimming	27
United States	51218	Dara Torres	12	Swimming	33
Finland	67728	Paavo Nurmi	12	Athletics	
Norway	85378	Bjrn Dhlie	12	Cross Country Skiing	24
United States	103301	Natalie Coughlin	12	Swimming	22
United States	103585	Ryan Lochte	12	Swimming	32
Czech Republic Czechoslovakia	29115	Vra slavsk	11	Artistic Gymnastics	18
Soviet Union	31270	Viktor Chukarin	11	Artistic Gymnastics	30
United States	44730	Carl Osburn	11	Shooting	
United States	51266	Matt Biondi	11	Swimming	18
United States	51572	Mark Spitz	11	Swimming	18
United States	105293	Allyson Felix	11	Athletics	35
Italy	109931	Arianna Fontana	11	Short Track Speed Skating	15
Australia	129061	Emma McKeon	11	Swimming	22
Belgium	333	Hubert Van Innis	10	Archery	
Hungary	22451	Aladr Gerevich	10	Fencing	22
Hungary	28630	gnes Keleti	10	Artistic Gymnastics	31
Soviet Union	29175	Polina Astakhova	10	Artistic Gymnastics	20
Total			**44571**		

Discussion and Conclusion

In the study, we can see that the kpis does play a role in an athlete or a country exceling at the Olympics. Even though the analysis was based on these factors, other factors can also determine whether a country win such as health, education, an athletes rec.

If there are future analyses on the Olympics in the other factors can be included to obtain

Problems Encountered

Microsoft Access Database Engine 2016 Redistributable to re

References

https://www.youtube.com/watch?v=wTVoSX37ols&t=262s

https://social.msdn.microsoft.com/Forums/sqlserver/en-US/8bfdd30a-526a-435e-b915-0f86645d0496/split-the-names-in-derived-column-in-ssis?forum=sqlintegrationservices

*Loading date

https://www.youtube.com/watch?v=5OieIJeNXZA

Connection Managers
https://www.youtube.com/watch?v=L1MSU0XjV3U&list=PLRZy31u3pia7Hq85lwGaobA85JMKSDyvS&index=8

Removing special characters
https://www.youtube.com/watch?v=wTVoSX37ols&t=262s

https://learn.microsoft.com/en-us/sql/integration-services/expressions/replacenull-ssis-expression?redirectedfrom=MSDN&view=sql-server-ver16

null values
https://www.youtube.com/watch?v=GDBnYL958c4

https://learn.microsoft.com/en-us/sql/integration-services/expressions/isnull-ssis-expression?view=sql-server-ver16

merge join , join
https://www.youtube.com/watch?v=P3s48WDNVzE

merge
https://www.youtube.com/watch?v=JuHcoEvIwQk

https://learn.microsoft.com/en-us/sql/integration-services/lesson-1-6-adding-and-configuring-the-lookup-transformations?view=sql-server-ver16

https://www.mssqltips.com/sqlservertip/1322/merge-multiple-data-sources-with-sql-server-integration-services/

dim date
https://www.youtube.com/watch?v=5OieIJeNXZA

date time conversion

https://www.youtube.com/watch?v=L4tbtZDJqfE
lookup

https://www.sqlshack.com/an-overview-of-the-lookup-transformation-in-ssis/

https://www.learnmsbitutorials.net/ssis-lookup-transformation-example.php

Power bi

https://www.wallstreetmojo.com/power-bi-related/

https://learn.microsoft.com/en-us/dax/or-function-dax