Section 1: Practical project: Modeling, Building a Data Warehouse for Gravity Bookstore

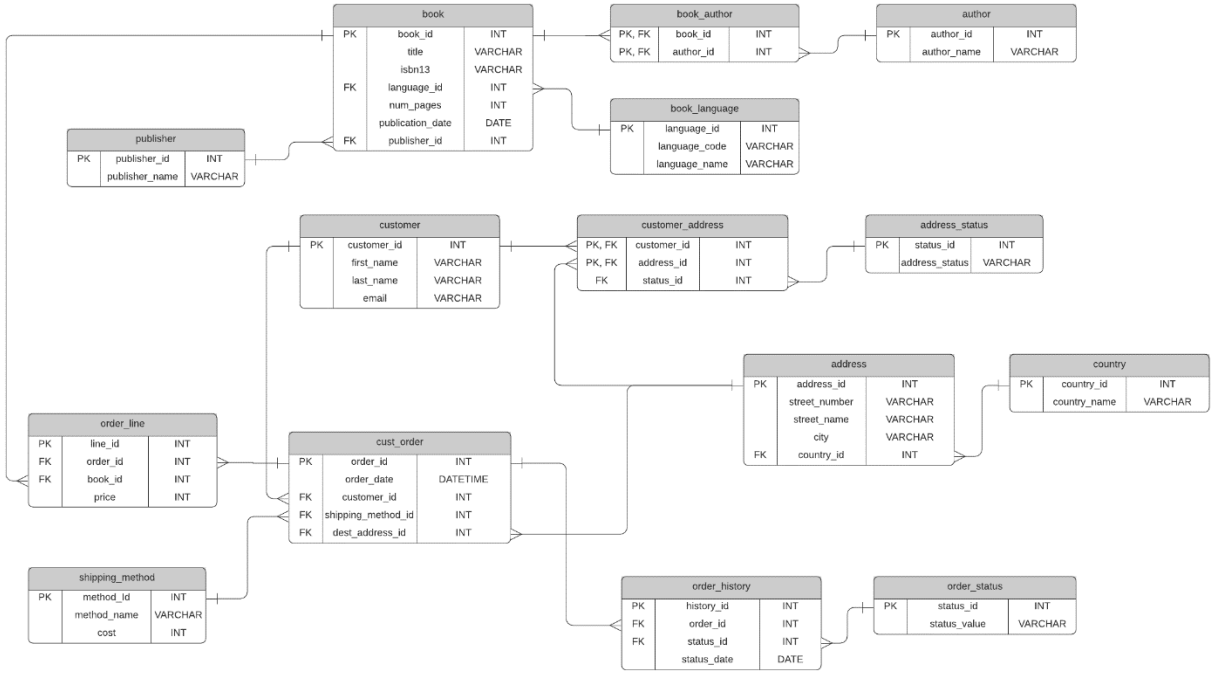
Use-case Background

Gravity Bookstore is a database for a fictional bookstore called that captures information about books, customers, and sales.

Please download the SQL files from the link below and run them in a new database named 'gravity_books' in your local SQL Server DB engine:

[databasestar/sample_databases/sample_db_gravity/gravity_sqlserver · GitHub]

ERD of 'gravity books' transactional database:



Tables description:

- **book**: a list of all books available in the store.
- **book_author**: stores the authors for each book, which is a many-to-many relationship.
- **author**: a list of all authors.
- **book_language**: a list of possible languages of books.
- **publisher**: a list of publishers for books.
- **customer**: a list of the customers of the Gravity Bookstore.
- **customer_address**: a list of addresses for customers, as a customer can have more than one address, and an address has more than one customer.
- **address_status**: a list of statuses for an address, because addresses can be current or old.
- **address**: a list of addresses in the system.
- **country**: a list of countries that addresses are in.

- **cust_order**: a list of orders placed by customers.
- **order_line**: a list of books that are a part of each order.
- **shipping_method**: the possible shipping methods for an order.
- **order_history**: the history of an order, such as ordered, cancelled, delivered.
- **order_status**: the possible statuses of an order.

Requirements

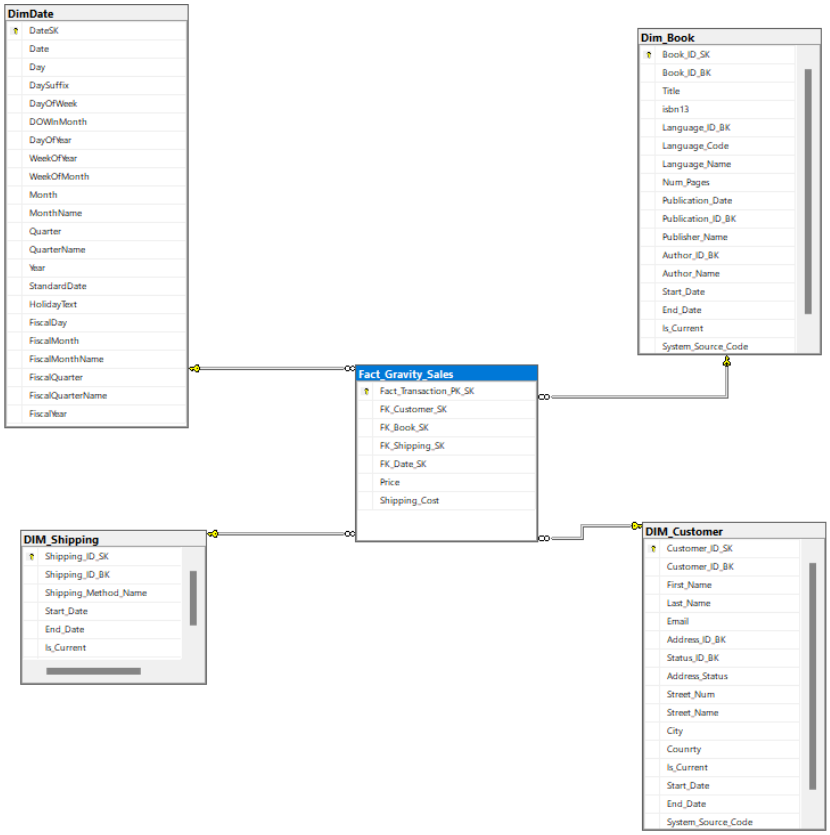
1. Model and develop ‘gravity_books_dwh’ Data Warehouse
    a. Please provide the DDL statements of table creation and a screenshot from the DWH ERD.
2. Define which approach (star schema, snowflake) of data warehouse used in your solution, and why.
3. Define a method to check and maintain the integrity between the fact and the dimensions (SQL).
4. Please, add a date dimension to the system to track the historical changes.
5. Design an SSIS project to populate the data from ‘gravity_books’ transactional database into the new target data warehouse ‘gravity_books_dwh’.
6. Design an SSAS project (multidimensional mode) and provide the main deliverables of the cube browsing.

- Please provide screenshots from the output of each point above:
- Compress the entire solution files:
    - DWH DDL statements (format . sql)
    - ETL SSIS project
    - OLAP SSAS project
    - Mapping Sheet

1. **Model and develop 'gravity_books_dwh' Data Warehouse**
   a. **Please provide the DDL statements of table creation and a screenshot from the DWH ERD.**

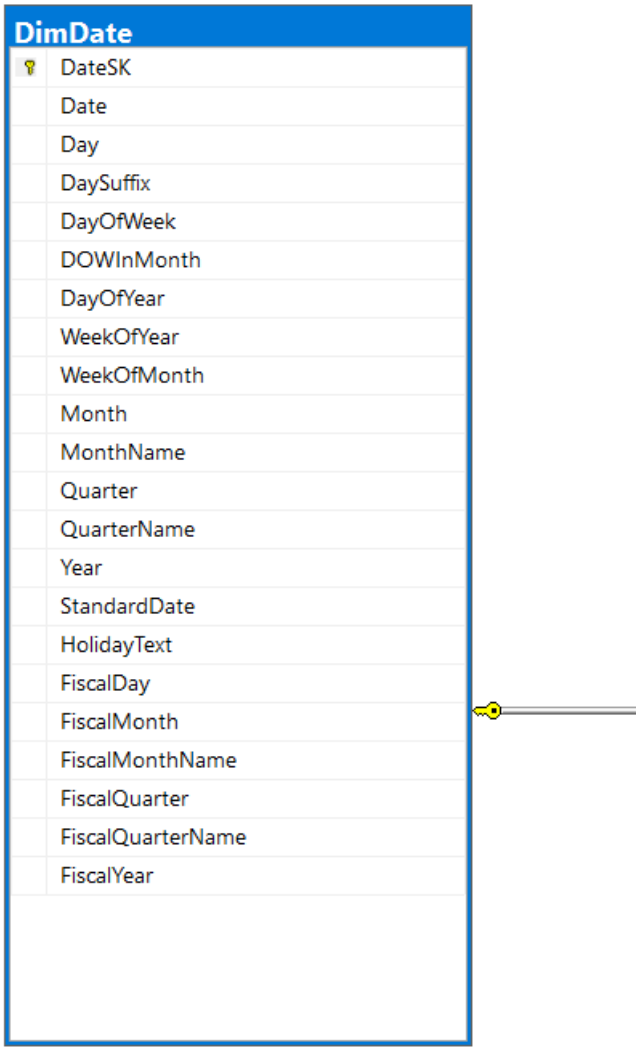
```
CREATE TABLE [dbo].[Dim_Book](
    [Book_ID_SK] [int] IDENTITY(1,1) NOT NULL,
    [Book_ID_BK] [int] NOT NULL,
    [Title] [varchar](400) NULL,
    [isbn13] [varchar](13) NULL,
    [Language_ID_BK] [int] NULL,
    [Language_Code] [varchar](8) NULL,
    [Language_Name] [varchar](50) NULL,
    [Num_Pages] [int] NULL,
    [Publication_Date] [date] NULL,
    [Publication_ID_BK] [int] NULL,
    [Publisher_Name] [nvarchar](1000) NULL,
    [Author_ID_BK] [int] NULL,
    [Author_Name] [varchar](400) NULL,
    [Start_Date] [datetime] NOT NULL,
    [End_Date] [datetime] NULL,
    [Is_Current] [tinyint] NOT NULL,
    [System_Source_Code] [tinyint] NULL,
 CONSTRAINT [PK_Dim_Book] PRIMARY KEY CLUSTERED
(
    [Book_ID_SK] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[DIM_Customer](
    [Customer_ID_SK] [int] IDENTITY(1,1) NOT NULL,
    [Customer_ID_BK] [int] NOT NULL,
    [First_Name] [varchar](200) NULL,
    [Last_Name] [varchar](200) NULL,
    [Email] [varchar](350) NULL,
    [Address_ID_BK] [int] NULL,
    [Status_ID_BK] [int] NULL,
    [Address_Status] [varchar](30) NULL,
    [Street_Num] [varchar](10) NULL,
    [Street_Name] [varchar](200) NULL,
    [City] [varchar](100) NULL,
    [Counrty] [varchar](200) NULL,
    [Is_Current] [tinyint] NOT NULL,
    [Start_Date] [datetime] NOT NULL,
    [End_Date] [datetime] NULL,
    [System_Source_Code] [tinyint] NULL,
 CONSTRAINT [PK_DIM_Customer] PRIMARY KEY CLUSTERED
(
    [Customer_ID_SK] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
```

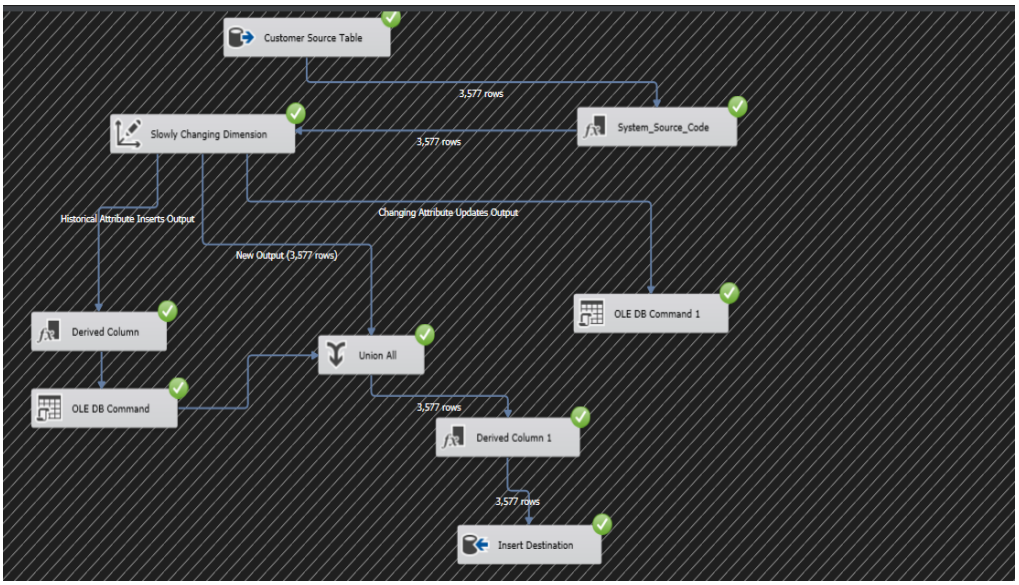


2. **Define which approach (star schema, snowflake) of data warehouse used in your solution, and why.**
   - I have used Star schema in my model design
   - In a star schema, the data warehouse is designed with one central fact table surrounded by dimension tables. The fact table contains measures (quantifiable data) and foreign keys to related dimension tables, so we found that star schema is:
     - Simple
     - High Performance
     - Denormalized
     - The query performance in reading is critical, such as business intelligence and reporting
3. **Define a method to check and maintain the integrity between the fact and the dimensions (SQL).**
   - By implementing foreign keys, constraints, and regular data validation, you can ensure the integrity of your data warehouse by maintaining consistent relationships between fact and dimension tables. This helps prevent data anomalies and inaccuracies, which are critical for accurate reporting and analysis in a data warehouse.
   - By matching the primary keys in dimensions with surrogate key in the fact table

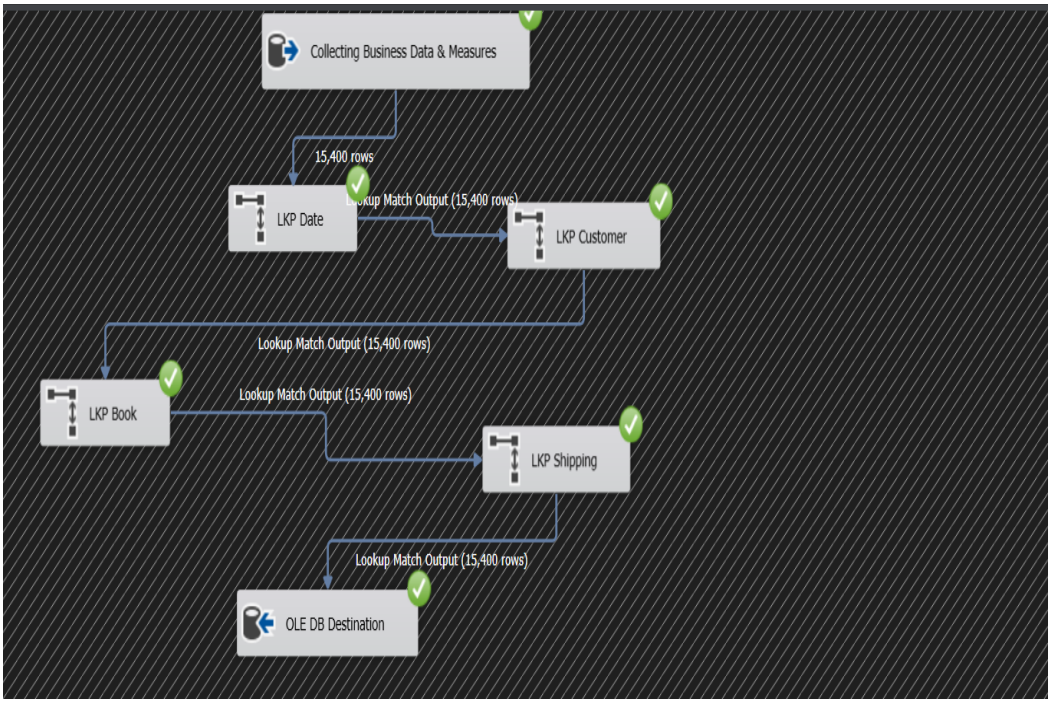
4. **Please, add a date dimension to the system to track the historical changes.**



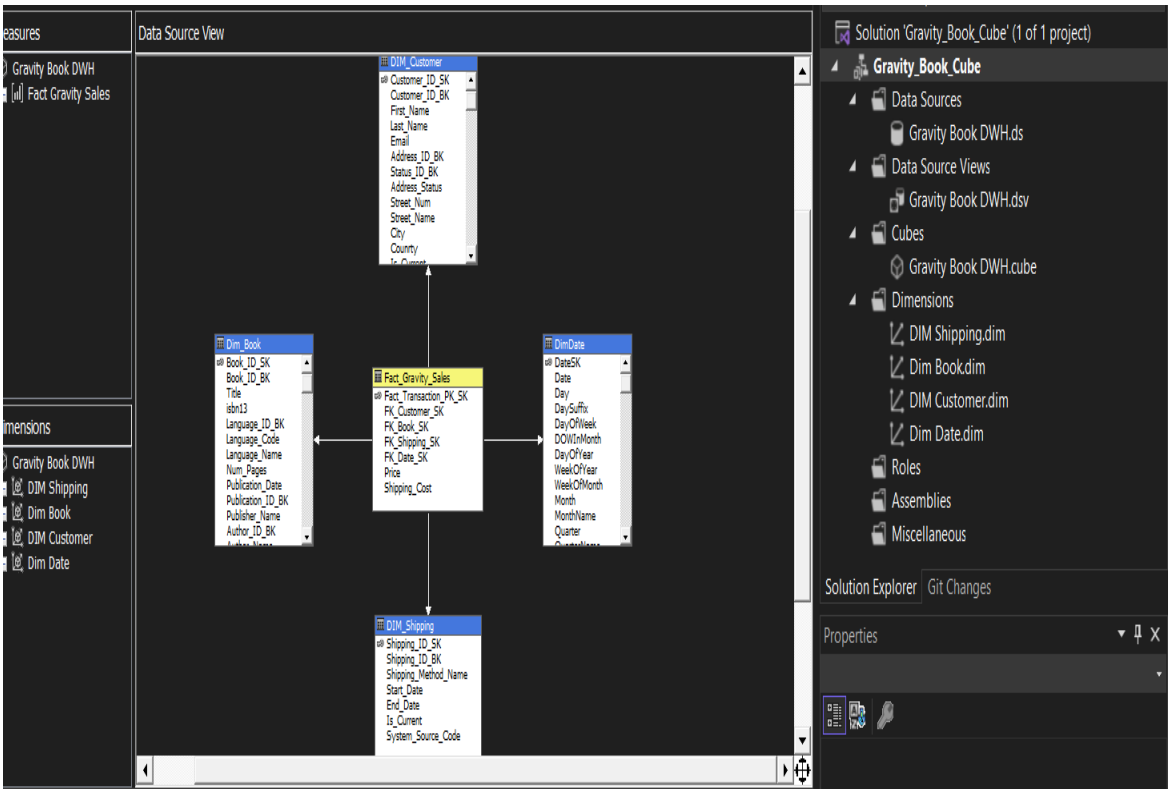
5. **Design an SSIS project to populate the data from 'gravity_books' transactional database into the new target data warehouse 'gravity_books_dwh'.**



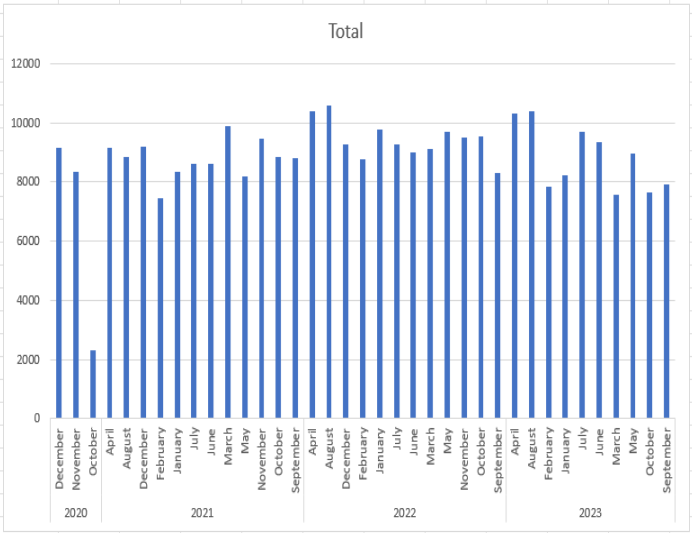

OLE DB Source
18,711 rows
Derived Column
18,711 rows
Slowly Changing Dimension
Historical Attribute Inserts Output
Changing Attribute Updates Output
Derived Column 1
OLE DB Command 1
OLE DB Command
New Output (18,711 rows)
Union All
18,711 rows
Derived Column 2
18,711 rows
Insert Destination


Collecting Business Data & Measures
15,400 rows
LKP Date
Lookup Match Output (15,400 rows)
LKP Customer
Lookup Match Output (15,400 rows)
LKP Book
Lookup Match Output (15,400 rows)
LKP Shipping
Lookup Match Output (15,400 rows)
OLE DB Destination

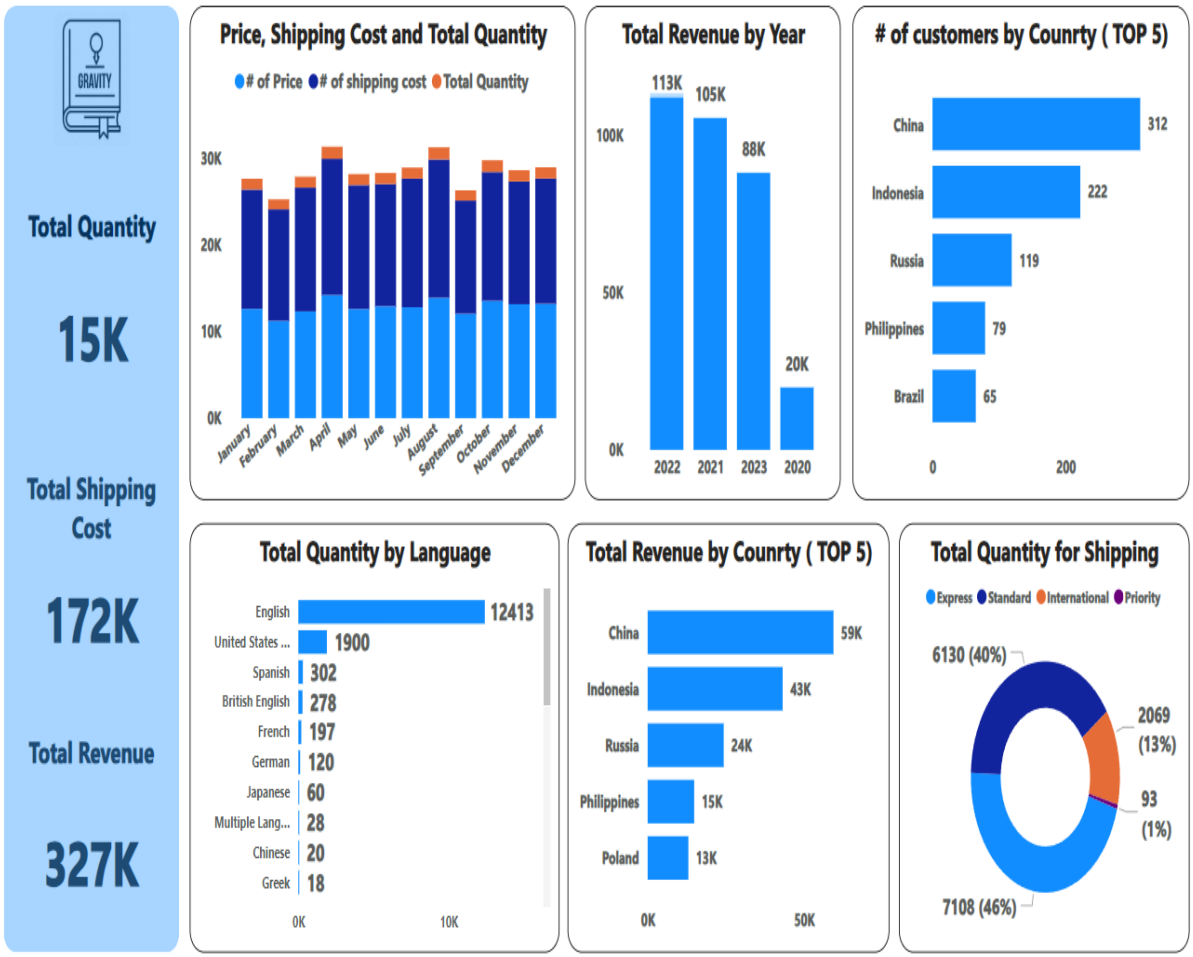
6. Design an SSAS project (multidimensional mode) and provide the main deliverables of the cube browsing.



Row Labels	Total_Revenue
2020	
December	9167.49
November	8333.88
October	2325.81
2021	
April	9174.91
August	8831
December	9179.1
February	7471.01
January	8341.39
July	8623.37
June	8627.29
March	9880.82
May	8196.66
November	9454.54
October	8839.49
September	8826.27
2022	
April	10390.09
August	10596.59
December	9280.3
February	8781.83
January	9766.57
July	9274.03
June	9012.19
March	9133.62
May	9692.44
November	9492.68
October	9554.88
September	8323.06
2023	



7. Dashboard


Total Quantity
15K
Total Shipping Cost
172K
Total Revenue
327K
Price, Shipping Cost and Total Quantity
# of Price
# of shipping cost
Total Quantity
30K
20K
10K
0K
January
February
March
April
May
June
July
August
September
October
November
December
Total Revenue by Year
113K
105K
88K
20K
100K
50K
0K
2022
2021
2023
2020
# of customers by Counrty (TOP 5)
China
312
Indonesia
222
Russia
119
Philippines
79
Brazil
65
0
200
Total Quantity by Language
English
12413
United States ...
1900
Spanish
302
British English
278
French
197
German
120
Japanese
60
Multiple Lang...
28
Chinese
20
Greek
18
0K
10K
Total Revenue by Counrty (TOP 5)
China
59K
Indonesia
43K
Russia
24K
Philippines
15K
Poland
13K
0K
50K
Total Quantity for Shipping
Express
Standard
International
Priority
6130 (40%)
2069
(13%)
93
(1%)
7108 (46%)