

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Практикум №1

з курсу «Аналіз даних в інформаційних системах»

на тему: «Створення сховища даних»

Викладач:
Олійник Ю.О.

Виконав:
студент 2 курсу
групи ІП-14 ФІОТ
Шляхтун Денис

Київ-2023

1) Для виконання лабораторної роботи було обрано 3 джерела відкритих даних на сайті <https://www.kaggle.com/>. А саме:

1. Top Video Games 1995-2021 Metacritic:
<https://www.kaggle.com/deepcontractor/top-video-games-19952021-metacritic>
2. Game Recommendations on Steam:
<https://www.kaggle.com/antonkozyriev/game-recommendations-on-steam>
3. Video Game Sales: <https://www.kaggle.com/ibriee/video-games-sales-dataset-2022-updated-extra-feat>

Вигляд Stage зони:

Metacritic top videogames	
Text	Name
Text	Platform
Date	Release_date
Int	Meta_score
Float	User_score

Steam recommendations	
Int	app_id
Text	title
Date	date_release
Text	rating
Int	positive_ratio
Int	user_reviews
Money	price_final

Sales	
Text	Name
Text	Platform
Int	Year_of_release
Text	Genre
Text	Publisher
Float	NA_Sales
Float	EU_sales
Float	JP_Sales
Float	Other_Sales
Float	Global_Sales
Text	Developer

Скрипти створення таблиць Stage зони знаходяться у додатку А.

2) Предметною областю лабораторної роботи є комп'ютерні ігри. У моделі сховища за типом сніжинка спроектовано дві таблиці фактів та десять таблиць вимірів.

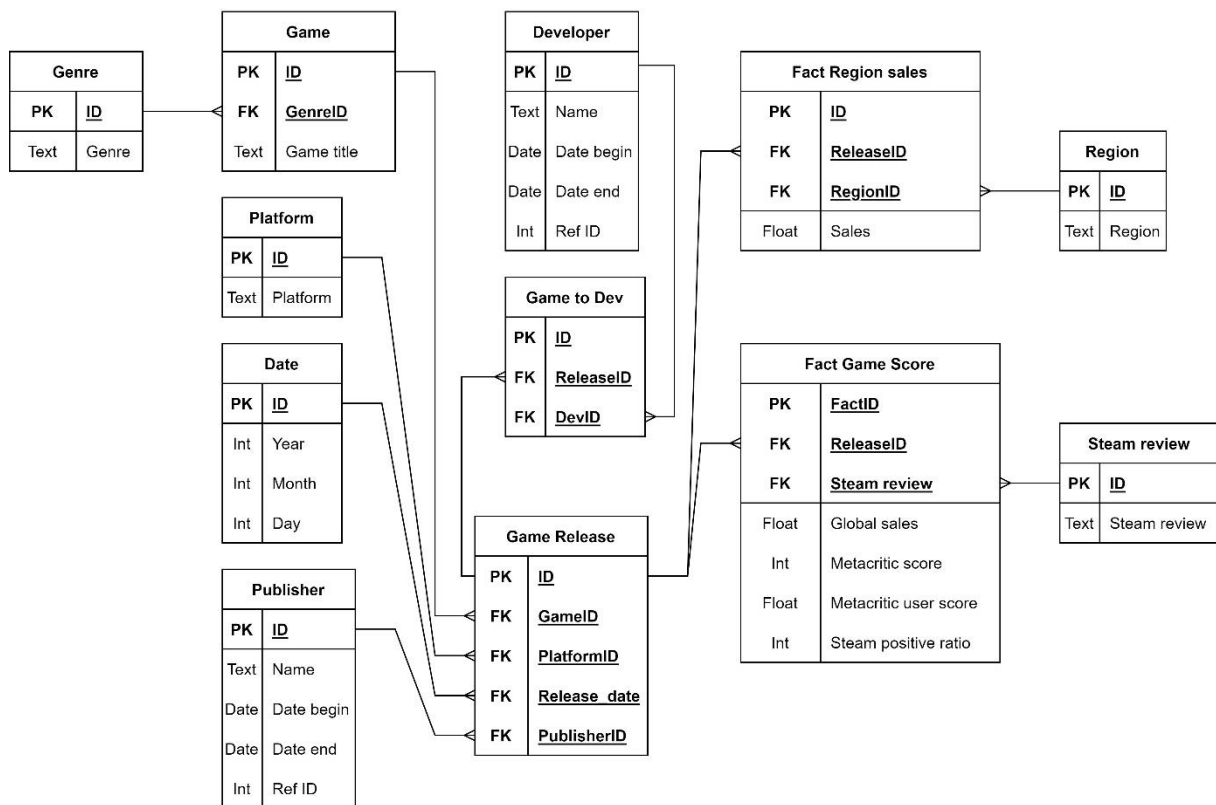


Рисунок 2.1 Проектування моделі сховища за типом сніжинка

Таблиця 1 – Таблиця виміру жанрів Genre

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор запису жанру
Genre		NVAR CHAR	Назва жанру

Таблиця 2 – Таблиця виміру розробників Developer

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор запису розробника
Developer		NVAR CHAR	Назва розробника

Таблиця 3 – Таблиця виміру ігор Game

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор запису гри
GenreID	FK	INT	Ідентифікатор запису жанру
Title		NVAR CHAR	Назва гри

Таблиця 4 – Таблиця виміру платформ Platform_

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор запису платформи
Platform_		NVAR CHAR	Назва платформи (скорочена)
Alt_name		NVAR CHAR	Альтернативна (повна) назва платформи

Таблиця 5 – Таблиця виміру видавців Publisher_

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор запису видавця
Publisher		NVAR CHAR	Назва видавця

Таблиця 6 – Таблиця виміру дати Date_

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор запису дати
Year_		INT	Рік
Month_		INT	Місяць

Day_		INT	День
------	--	-----	------

Таблиця 7 – Таблиця виміру ігрових релізів Game_Release

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор релізу гри
GameID	FK	INT	Ідентифікатор гри
PlatformID	FK	INT	Ідентифікатор платформи
Release_date	FK	INT	Ідентифікатор дати
PublisherID	FK	INT	Ідентифікатор видавця

Таблиця 8 – Таблиця зв'язку ігрових релізів і розробників Game_to_Dev

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор зв'язку
ReleaseID	FK	INT	Ідентифікатор релізу гри
DeveloperID	FK	INT	Ідентифікатор розробника

Таблиця 9 – Таблиця виміру відгуків Steam Steam_Review

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор відгуку
Steam_Review		NVAR CHAR	Відгук Steam

Таблиця 10 – Таблиця виміру регіонів Region

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор регіону
Region		NVAR CHAR	Назва регіону

Таблиця 11 – Фактова таблиця продажів ігор за регіонами
Fact_Regional_Sales

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор запису продажів
ReleaseID	FK	INT	Ідентифікатор релізу гри
RegionID	FK	INT	Ідентифікатор регіону
Sales		REAL	Кількість проданих копій гри

Таблиця 12 – Фактова таблиця оцінок ігор Fact_Game_Score

Назва поля	Ключ	Тип	Опис поля
ID	PK	INT	Унікальний ідентифікатор запису продажів
ReleaseID	FK	INT	Ідентифікатор релізу гри
Steam_ReviewID	FK	INT	Ідентифікатор відгуку Steam
Global_sales		REAL	Кількість проданих копій гри
Metacritic_score		INT	Оцінка від Metacritic
Metacritic_user_score		REAL	Оцінка від користувачів сайту Metacritic
Steam_positive_ratio		INT	Відсоток позитивних відгуків Steam

Скрипти створення основного сховища за типом «зірка» знаходяться у додатку Б

4) Імпорт даних у Stage зону здійснювався засобами SQL Server Management Studio. Для перенесення даних у таблиці сховища даних з Stage зони були розроблені збережені процедури:

1. Add_genres – додає з Stage зони унікальні значення для таблиці Genre;
2. Add_developers – додає з Stage зони унікальні значення для таблиці Developer, розробники вказані через кому, тому їх потрібно розділяти;
3. Add_games – додає з Stage зони унікальні значення назви гри та зв'язані значення жанру для таблиці Game;
4. Add_platforms – додає з Stage зони унікальні значення для таблиці Platform_;
5. Add_publishers – додає з Stage зони унікальні значення для таблиці Publisher;
6. Add_dates – додає з Stage зони унікальні значення для таблиці Date_;
7. Add_releases – додає з Stage зони інформацію про реліз гри у таблицю Game_Release, що зв'язана з таблицями Game, Platform_, Publisher, Date_, тобто автоматизовано підставлення ідентифікаторів цих таблиць;
8. Add_Game_to_Dev – зв'язує релізи ігор з розробниками з інформації Stage зони, підставка ідентифікаторів з таблиць Game_Release та Developer;
9. Add_steam_review – додає з Stage зони унікальні значення для таблиці Steam_Review;
10. Add_regional_sales – додає з Stage зони інформацію про продажі ігор для таблиці Fact_Regional_Sales, відбувається підставка ідентифікаторів з таблиць Game_Release, Region;
11. Add_game_scores – додає з Stage зони інформацію про оцінки ігор для таблиці Fact_Game_Score, відбувається підставка ідентифікаторів з таблиць Game_Release, Steam_Review, для зв'язування даних різних таблиць Stage зони назва гри була переведена в нижній реєстр і були вилучені символи «™» та «®».

Повний текст процедур знаходиться у додатку В.

5) Приклад SELECT запиту для таблиці Fact_Game_Score з поєднанням інформації з таблиць Game, Genre, Platform_, Publisher

Title	Genre	Platform_	Publisher	Global_sales	Metacritic_score	Metacritic_user_score	Steam_Review	Steam_positive_ratio
Earth Defense Force: Insect Armag...	Shooter	PS3	D3Publisher	0.2	63	7,4	Very Positive	89
Earth Defense Force: Insect Armag...	Shooter	X360	D3Publisher	0.19	68	8	Very Positive	89
Edna & Harvey: Harvey's New Eyes	Adventure	PC	Rondomedia	0.08	75	7,6	Very Positive	89
Exit	Puzzle	PSP	Ubisoft	0.08	77	7,6	Positive	89
Guild Wars 2	Action	PC	NCSoft	2,31	90	8	Very Positive	89
Guilty Gear X2 #Reload	Fighting	XB	Zoo Digital Publishing	0.15	86	7,3	Very Positive	89
Goodbye Deponia	Adventure	PC	Daedalic Entertainment	0.07	80	7,9	Very Positive	89
Grand Theft Auto 2	Action	PS	Take-Two Interactive	3,42	70	7,7	Very Positive	89
Final Fantasy IV	Simulation	DS	Square Enix	1,21	85	8,4	Very Positive	89
Killing Floor 2	Shooter	PS4	Tripwire Interactive	0.05	75	7,1	Very Positive	89
Wolfenstein: The Old Blood	Action	XOne	Bethesda Softworks	0.15	75	7,6	Very Positive	89
Wolfenstein: The Old Blood	Action	PS4	Bethesda Softworks	0.43	76	7,7	Very Positive	89
Wolfenstein: The Old Blood	Action	PC	Bethesda Softworks	0.09	76	7,6	Very Positive	89
Tropico 3	Strategy	PC	Kalypso Media	0.04	79	8,1	Very Positive	89
Tropico 3	Strategy	X360	Kalypso Media	0.22	75	6,9	Very Positive	89
Vanquish	Shooter	PS3	Sega	0.61	84	9,1	Very Positive	89
Vanquish	Shooter	X360	Sega	0.42	84	9,2	Very Positive	89
Homeworld Remastered Collection	Strategy	PC	N/A	0.01	86	8,2	Very Positive	88
Final Fantasy VIII	Role-Playing	PS	SquareSoft	7,86	90	8,7	Very Positive	88

Висновок.

При виконанні даної лабораторної роботи було розглянуто проектування моделі сховища даних за типом сніжинка. Спроектовано модель Stage зони для ETL процесів. Було створено функції для перетворення та завантаження даних до основного сховища.

Додаток А. Створення таблиць Stage зони

```
CREATE TABLE Metacritic
(
    Title NVARCHAR(200),
    Platform_ NVARCHAR(50),
    Release_date DATETIME2,
    Meta_score INT,
    User_score REAL
)
```

```
CREATE TABLE Sales
(
    Title NVARCHAR(200),
    Platform_ NVARCHAR(50),
    Year_of_release INT,
    Genre NVARCHAR(50),
    Publisher NVARCHAR(50),
    NA_Sales REAL,
    EU_Sales REAL,
    JP_Sales REAL,
    Other_sales REAL,
    Global_sales REAL,
    Developer NVARCHAR(100)
)
```

```
CREATE TABLE Steam
(
    ID INT,
    Title NVARCHAR(200),
    Release_date DATETIME2,
    Rating NVARCHAR(50),
    Positive_ratio INT,
    User_reviews INT,
    Price MONEY
)
```

Додаток Б. Створення таблиць сховища типу «зірка»

```
CREATE TABLE Genre
(
    ID INT IDENTITY(1, 1) PRIMARY KEY,
    Genre NVARCHAR(50) NOT NULL UNIQUE
)

CREATE TABLE Developer
(
    ID INT IDENTITY(1, 1) PRIMARY KEY,
    Developer NVARCHAR(100) NOT NULL
)

CREATE TABLE Game
(
    ID INT IDENTITY(1, 1) PRIMARY KEY,
    GenreID INT REFERENCES Genre(ID),
    Title NVARCHAR(200) NOT NULL
)

CREATE TABLE Platform_
(
    ID INT IDENTITY(1, 1) PRIMARY KEY,
    Platform_ NVARCHAR(50) NOT NULL unique
)

CREATE TABLE Publisher
(
    ID INT IDENTITY(1, 1) PRIMARY KEY,
    Publisher NVARCHAR(50) NOT NULL unique
)

CREATE TABLE Date_
(
    ID INT IDENTITY(1, 1) PRIMARY KEY,
    Year_ INT NOT NULL,
    Month_ INT,
    Day_ INT
)

CREATE TABLE Game_Release
(
    ID INT IDENTITY(1, 1) PRIMARY KEY,
    GameID INT REFERENCES Game(ID) NOT NULL,
    PlatformID INT REFERENCES Platform_(ID),
    Release_date INT REFERENCES Date_(ID),
    PublisherID INT REFERENCES Publisher(ID)
)

CREATE TABLE Game_to_Dev
(
    ID INT IDENTITY(1, 1) PRIMARY KEY,
    ReleaseID INT REFERENCES Game_Release(ID),
    DeveloperID INT REFERENCES Developer(ID)
)

CREATE TABLE Steam_Review
(
    ID INT IDENTITY(1, 1) PRIMARY KEY,
    Steam_Review NVARCHAR(50) NOT NULL unique
)

CREATE TABLE Region
(

```

```
        ID INT IDENTITY(1, 1) PRIMARY KEY,
        Region NVARCHAR(50) NOT NULL unique
    )

CREATE TABLE Fact_Regional_Sales
(
    ID INT IDENTITY(1, 1) PRIMARY KEY,
    ReleaseID INT REFERENCES Game_Release(ID),
    RegionID INT REFERENCES Region(ID),
    Sales REAL
)

CREATE TABLE Fact_Game_Score
(
    ID INT IDENTITY(1, 1) PRIMARY KEY,
    ReleaseID INT REFERENCES Game_Release(ID),
    Steam_ReviewID INT REFERENCES Steam_review(ID),
    Global_sales REAL,
    Metacritic_score INT,
    Metacritic_user_score REAL,
    Steam_positive_ratio INT
)
```

Додаток В. Створення збережених процедур

```
CREATE TYPE Text_table AS TABLE
(
    Text_ NVARCHAR(200)
)

GO
CREATE OR ALTER PROCEDURE Add_genres
    @table Text_table READONLY
AS
BEGIN
    INSERT INTO Genre(Genre) (SELECT DISTINCT Text_ FROM @table WHERE Text_ IS NOT
NULL EXCEPT SELECT Genre FROM Genre)
END
GO

GO
CREATE OR ALTER PROCEDURE Add_developers
    @table Text_table READONLY
AS
BEGIN
    INSERT INTO Developer(Developer) (SELECT DISTINCT TRIM(VALUE) FROM @table CROSS
APPLY STRING_SPLIT(Text_, ',') WHERE Text_ IS NOT NULL EXCEPT SELECT Developer FROM
Developer)
END
GO

GO
CREATE OR ALTER PROCEDURE Add_games
AS
BEGIN
    INSERT INTO Game(Title, GenreID) ((SELECT DISTINCT Title, Genre.ID FROM Sales
JOIN Genre ON Sales.Genre = Genre.Genre WHERE Title IS NOT NULL) EXCEPT (SELECT Title,
GenreID FROM Game))
END
GO

GO
CREATE OR ALTER PROCEDURE Add_platforms
    @table Text_table READONLY
AS
BEGIN
    INSERT INTO Platform_(Platform_) (SELECT DISTINCT Text_ FROM @table WHERE Text_
IS NOT NULL EXCEPT SELECT Platform_ FROM Platform_)
END
GO

ALTER TABLE Platform_ ADD Alt_name NVARCHAR(50)
UPDATE Platform_ SET Alt_name = 'PlayStation Vita' WHERE Platform_ = 'PSV'
UPDATE Platform_ SET Alt_name = 'PC' WHERE Platform_ = 'PC'
UPDATE Platform_ SET Alt_name = 'Wii U' WHERE Platform_ = 'WiiU'
UPDATE Platform_ SET Alt_name = 'Nintendo 64' WHERE Platform_ = 'N64'
UPDATE Platform_ SET Alt_name = 'Game Boy Advance' WHERE Platform_ = 'GBA'
UPDATE Platform_ SET Alt_name = 'PlayStation' WHERE Platform_ = 'PS'
UPDATE Platform_ SET Alt_name = 'PlayStation 3' WHERE Platform_ = 'PS3'
UPDATE Platform_ SET Alt_name = 'Wii' WHERE Platform_ = 'Wii'
UPDATE Platform_ SET Alt_name = 'PlayStation 2' WHERE Platform_ = 'PS2'
UPDATE Platform_ SET Alt_name = 'PSP' WHERE Platform_ = 'PSP'
UPDATE Platform_ SET Alt_name = '3DS' WHERE Platform_ = '3DS'
UPDATE Platform_ SET Alt_name = 'Xbox 360' WHERE Platform_ = 'X360'
UPDATE Platform_ SET Alt_name = 'Dreamcast' WHERE Platform_ = 'DC'
```

```

UPDATE Platform_ SET Alt_name = 'PlayStation 4' WHERE Platform_ = 'PS4'
UPDATE Platform_ SET Alt_name = 'Xbox' WHERE Platform_ = 'XB'
UPDATE Platform_ SET Alt_name = 'Xbox One' WHERE Platform_ = 'XOne'
UPDATE Platform_ SET Alt_name = 'DS' WHERE Platform_ = 'DS'

GO
CREATE OR ALTER PROCEDURE Add_publishers
@table Text_table READONLY
AS
BEGIN
    INSERT INTO Publisher(Publisher) (SELECT DISTINCT Text_ FROM @table WHERE Text_
IS NOT NULL EXCEPT SELECT Publisher FROM Publisher)
END
GO

GO
CREATE OR ALTER PROCEDURE Add_dates
AS
BEGIN
    INSERT INTO Date_(Year_)
    (SELECT DISTINCT Year_of_release FROM Sales
    WHERE Year_of_release IS NOT NULL
    EXCEPT SELECT Year_ FROM Date_ WHERE Month_ IS NULL AND Day_ IS NULL)
    INSERT INTO Date_(Year_, Month_, Day_)
    (SELECT DISTINCT DATEPART(YEAR, Release_date),
    DATEPART(MONTH, Release_date),
    DATEPART(DAY, Release_date)
    FROM Metacritic WHERE Release_date IS NOT NULL
    EXCEPT SELECT Year_, Month_, Day_ FROM Date_)
    INSERT INTO Date_(Year_, Month_, Day_)
    (SELECT DISTINCT DATEPART(YEAR, Release_date),
    DATEPART(MONTH, Release_date),
    DATEPART(DAY, Release_date)
    FROM Steam WHERE Release_date IS NOT NULL
    EXCEPT SELECT Year_, Month_, Day_ FROM Date_)

END
GO

GO
CREATE OR ALTER PROCEDURE Add_releases
AS
BEGIN
    INSERT INTO Game_Release(GameID, PlatformID, PublisherID)
    SELECT DISTINCT Game.ID, Platform_.ID, Publisher.ID FROM Game
    JOIN Sales ON Game.Title = Sales.Title
    LEFT JOIN Platform_ ON Sales.Platform_ = Platform_.Platform_
    LEFT JOIN Publisher ON Sales.Publisher = Publisher.Publisher
    EXCEPT SELECT GameID, PlatformID, PublisherID FROM Game_Release
    UPDATE Game_Release SET Release_date = Date_.ID
    FROM Game_Release
    JOIN Game ON Game_Release.GameID = Game.ID
    JOIN Platform_ ON Game_Release.PlatformID = Platform_.ID
    JOIN Metacritic ON lower(Metacritic.Title) = lower(Game.Title)
    JOIN Date_ ON Date_.Year_ = DATEPART(YEAR, Metacritic.Release_date)
    AND Date_.MONTH_ = DATEPART(MONTH, Metacritic.Release_date)
    AND Date_.day_ = DATEPART(DAY, Metacritic.Release_date)
    UPDATE Game_Release SET Release_date = Date_.ID
    FROM Game_Release
    JOIN Game ON Game_Release.GameID = Game.ID
    JOIN Platform_ ON Game_Release.PlatformID = Platform_.ID

```

```

        JOIN Steam ON lower(replace(replace(Steam.Title, 'TM', ''), '@', '')) =
lower(Game.Title)
        JOIN Date_ ON Date_.Year_ = DATEPART(YEAR, Steam.Release_date)
        AND Date_.MONTH_ = DATEPART(MONTH, Steam.Release_date)
        AND Date_.day_ = DATEPART(DAY, Steam.Release_date)
        WHERE Game_Release.Release_date IS NULL
    UPDATE Game_Release SET Release_date = Date_.ID
    FROM Game_Release
    JOIN Game ON Game_Release.GameID = Game.ID
    JOIN Platform_ ON Game_Release.PlatformID = Platform_.ID
    JOIN Sales ON Sales.Title = Game.Title
    JOIN Date_ ON Date_.Year_ = Sales.Year_of_release
    AND Date_.MONTH_ IS NULL
    AND Date_.day_ IS NULL
    WHERE Game_Release.Release_date IS NULL

END

GO
CREATE OR ALTER PROCEDURE Add_Game_to_Dev
AS
BEGIN
    INSERT INTO Game_to_Dev(ReleaseID, DeveloperID)
    (SELECT DISTINCT Game_Release.ID, Developer.ID FROM Game
    JOIN Game_Release ON Game.ID = Game_Release.GameID
    JOIN (SELECT DISTINCT Title, Genre, Platform_, TRIM(VALUE) AS Dev FROM
Sales CROSS APPLY STRING_SPLIT(Developer, ','))tabl2
    ON Game.Title = tabl2.Title
    and tabl2.Genre = (SELECT Genre FROM Genre WHERE ID = Game.GenreID)
    AND tabl2.Platform_ = (SELECT Platform_ FROM Platform_ WHERE ID =
Game_Release.PlatformID)
    JOIN Developer ON Developer.Developer = tabl2.Dev
    )
    EXCEPT SELECT ReleaseID, DeveloperID FROM Game_to_Dev

END
GO

GO
CREATE OR ALTER PROCEDURE Add_steam_review
@table Text_table READONLY
AS
BEGIN
    INSERT INTO Steam_Review(Steam_Review) (SELECT DISTINCT Text_ FROM @table WHERE
Text_ IS NOT NULL EXCEPT SELECT Steam_Review FROM Steam_Review)
END
GO

INSERT INTO Region(Region) VALUES('NA')
INSERT INTO Region(Region) VALUES('EU')
INSERT INTO Region(Region) VALUES('JP')
INSERT INTO Region(Region) VALUES('Other')
INSERT INTO Region(Region) VALUES('Global')

GO
CREATE OR ALTER PROCEDURE Add_regional_sales
AS
BEGIN
    INSERT INTO Fact_Regional_Sales(ReleaseID, RegionID, Sales)
    SELECT DISTINCT Game_Release.ID, (SELECT ID FROM Region WHERE Region =
'NA'), Sales.NA_Sales FROM Sales
    JOIN Game_Release ON Sales.Title = (SELECT Game.Title FROM Game
WHERE Game_Release.GameID = Game.ID)

```

```

        AND Sales.Platform_ = (SELECT Platform_ FROM Platform_ WHERE
Game_Release.PlatformID = Platform_.ID)
        UNION SELECT DISTINCT Game_Release.ID, (SELECT ID FROM Region WHERE
Region = 'EU'), Sales.EU_Sales FROM Sales
        JOIN Game_Release ON Sales.Title = (SELECT Game.Title FROM Game
WHERE Game_Release.GameID = Game.ID)
        AND Sales.Platform_ = (SELECT Platform_ FROM Platform_ WHERE
Game_Release.PlatformID = Platform_.ID)
        UNION SELECT DISTINCT Game_Release.ID, (SELECT ID FROM Region WHERE
Region = 'JP'), Sales.JP_Sales FROM Sales
        JOIN Game_Release ON Sales.Title = (SELECT Game.Title FROM Game
WHERE Game_Release.GameID = Game.ID)
        AND Sales.Platform_ = (SELECT Platform_ FROM Platform_ WHERE
Game_Release.PlatformID = Platform_.ID)
        UNION SELECT DISTINCT Game_Release.ID, (SELECT ID FROM Region WHERE
Region = 'Other'), Sales.Other_sales FROM Sales
        JOIN Game_Release ON Sales.Title = (SELECT Game.Title FROM Game
WHERE Game_Release.GameID = Game.ID)
        AND Sales.Platform_ = (SELECT Platform_ FROM Platform_ WHERE
Game_Release.PlatformID = Platform_.ID)
        UNION SELECT DISTINCT Game_Release.ID, (SELECT ID FROM Region WHERE
Region = 'Global'), Sales.Global_sales FROM Sales
        JOIN Game_Release ON Sales.Title = (SELECT Game.Title FROM Game
WHERE Game_Release.GameID = Game.ID)
        AND Sales.Platform_ = (SELECT Platform_ FROM Platform_ WHERE
Game_Release.PlatformID = Platform_.ID)
        EXCEPT SELECT ReleaseID, RegionID, Sales FROM Fact_Regional_Sales
END
GO

GO
CREATE OR ALTER PROCEDURE Add_game_scores
AS
BEGIN
    INSERT INTO Fact_Game_Score(ReleaseID, Steam_ReviewID, Global_sales,
Metacritic_score, Metacritic_user_score, Steam_positive_ratio)
        SELECT Game_Release.ID, Steam_Review.ID, Sales.Global_sales,
Metacritic.Meta_score, Metacritic.User_score, Steam.Positive_ratio
        FROM Game_Release
        JOIN Game ON Game_Release.GameID = Game.ID
        JOIN Platform_ ON Game_Release.PlatformID = Platform_.ID
        JOIN Sales ON Sales.Title = Game.Title
        AND Sales.Platform_ = Platform_.Platform_
        JOIN Metacritic ON lower(Metacritic.Title) = lower(Game.Title)
        AND Metacritic.Platform_ = Platform_.Alt_name
        LEFT JOIN Steam ON lower(replace(replace(Steam.Title, 'TM', ''), '®', ''))
= lower(Game.Title)
        LEFT JOIN Steam_Review ON Steam.Rating = Steam_Review.Steam_Review
    EXCEPT SELECT ReleaseID, Steam_ReviewID, Global_sales, Metacritic_score,
Metacritic_user_score, Steam_positive_ratio FROM Fact_Game_Score

END
GO

```