# Video Game Sales and Genre Popularity (1980–2016)

## Introduction

This project explores global video game trends using a dataset of over 15,000 games released between 1980 and 2016. The goal is to uncover patterns in platform and genre popularity, regional sales dominance, and publisher performance. The project is organized into five sections, each using a specific SQL competency: Joins, Subqueries, Table Expressions, Set Operators, and Data Modifications.

## Dataset

### **Name:** Video Game Sales Dataset

**Source:** Kaggle (original), manually cleaned in Excel for SQL import
**Link:** Video Game Sales 2019 – Kaggle
**Columns:** 14 — Name, Platform, Genre, Developer, Publisher, Year_of_Release, Sales(5), Scores(2), Rating
**Record Count:** 15,751 rows (after cleaning)
**Notes:** Cleaned with Python queries from VS Code and manually in Excel to resolve encoding and data type issues that were preventing SQL import.

## Data Exploration

This section explores the structure of the tables `Video_Games_1980_2016`, `VD_Genres`, `VD_Platforms`, and `VD_Publishers`. The goal is to understand the dataset's schema, column names and meanings, data types, record counts, and uncover other interesting data

```
USE CAP2761C_Project
SELECT TOP 5 *
FROM Video_Games_1980_2016;
```

# Video_Games_1980_2016 Table Information:

| Column Name | Data Type | Description |
|---|---|---|
| Name | NVARCHAR(255) | Title of the video game |
| Platform | NVARCHAR(50) | Platform the game was released on (e.g., PS4, PC) |
| Genre | NVARCHAR(50) | Type/category of the game (e.g., Action, Sports) |
| Developer | NVARCHAR(255) | Studio or developer name |
| Publisher | NVARCHAR(255) | Company that published the game |
| Year_of_Release | INT | Year the game was released |
| NA_Sales | FLOAT | Sales in North America (in millions) |
| EU_Sales | FLOAT | Sales in Europe (in millions) |
| JP_Sales | FLOAT | Sales in Japan (in millions) |
| Other_Sales | FLOAT | Sales in other regions (in millions) |
| Global_Sales | FLOAT | Total global sales (in millions) |
| Critic_Score | FLOAT | Aggregate critic review score |
| User_Score | FLOAT | Aggregate user review score |
| Rating | NVARCHAR(50) | ESRB content rating (e.g., E, M, T) |

# Creating Separate Tables with Key Columns:

- Genre_ID > VD_Genres
- Platform_ID > VD_Platforms
- Publisher_ID > VD_Publishers

```
USE CAP2761C_Project
CREATE TABLE VD_Genres (
    Genre_ID INT IDENTITY(1,1) PRIMARY KEY,
    Genre_Name NVARCHAR(100) UNIQUE
);

USE CAP2761C_Project
INSERT INTO VD_Genres (Genre_Name)
SELECT DISTINCT Genre
FROM Video_Games_1980_2016
WHERE Genre IS NOT NULL;
```

## VD_Genres Table Information:

| Column Name | Data Type | Description |
|---|---|---|
| Genre_ID | INT (PK) | Unique ID for genre |
| Genre_Name | NVARCHAR(100) | Name of the game genre |

```
USE CAP2761C_Project
CREATE TABLE VD_Platforms (
    Platform_ID INT IDENTITY(1,1) PRIMARY KEY,
    Platform_Name NVARCHAR(100) UNIQUE
);

USE CAP2761C_Project
INSERT INTO VD_Platforms (Platform_Name)
SELECT DISTINCT Platform
FROM Video_Games_1980_2016
WHERE Platform IS NOT NULL;
```

## VD_Platforms table Information:

| Column Name | Data Type | Description |
|---|---|---|
| Platform_ID | INT (PK) | Unique ID for platform |
| Platform_Name | NVARCHAR(100) | Name of the game platform |

```
USE CAP2761C_Project
CREATE TABLE VD_Publishers (
    Publisher_ID INT IDENTITY(1,1) PRIMARY KEY,
    Publisher_Name NVARCHAR(255) UNIQUE
);

USE CAP2761C_Project
INSERT INTO VD_Publishers (Publisher_Name)
SELECT DISTINCT Publisher
FROM Video_Games_1980_2016
WHERE Publisher IS NOT NULL;
```

## VD_Publishers table Information:

| Column Name | Data Type | Description |
|---|---|---|
| Publisher_ID | INT (PK) | Unique ID for publisher |
| Publisher_Name | NVARCHAR(255) | Name of the game publisher |

## Altering Main Table to Add Foreign Key Columns

```
USE CAP2761C_Project
ALTER TABLE Video_Games_1980_2016 ADD Genre_ID INT;
```

```
USE CAP2761C_Project
ALTER TABLE Video_Games_1980_2016 ADD Platform_ID INT;

USE CAP2761C_Project
ALTER TABLE Video_Games_1980_2016 ADD Publisher_ID INT;
```

## Inserting the ID Columns with JOINs

```
USE CAP2761C_Project
UPDATE vg
SET Genre_ID = g.Genre_ID
FROM Video_Games_1980_2016 vg
JOIN VD_Genres g ON vg.Genre = g.Genre_Name;

USE CAP2761C_Project
UPDATE vg
SET Platform_ID = p.Platform_ID
FROM Video_Games_1980_2016 vg
JOIN VD_Platforms p ON vg.Platform = p.Platform_Name;

USE CAP2761C_Project
UPDATE vg
SET Publisher_ID = pub.Publisher_ID
FROM Video_Games_1980_2016 vg
JOIN VD_Publishers pub ON vg.Publisher = pub.Publisher_Name;
```

## Updated Video_Games_1980_2016 Table Information:

| Column Name | Data Type | Description |
|---|---|---|
| Name | NVARCHAR(255) | Title of the video game |
| Platform | NVARCHAR(50) | Platform the game was released on (e.g., PS4, PC) |
| Genre | NVARCHAR(50) | Type/category of the game (e.g., Action, Sports) |
| Developer | NVARCHAR(255) | Studio or developer name |
| Publisher | NVARCHAR(255) | Company that published the game |
| Year_of_Release | INT | Year the game was released |
| NA_Sales | FLOAT | Sales in North America (in millions) |
| EU_Sales | FLOAT | Sales in Europe (in millions) |
| JP_Sales | FLOAT | Sales in Japan (in millions) |
| Other_Sales | FLOAT | Sales in other regions (in millions) |

| Column Name | Data Type | Description |
|---|---|---|
| Global_Sales | FLOAT | Total global sales (in millions) |
| Critic_Score | FLOAT | Aggregate critic review score |
| User_Score | FLOAT | Aggregate user review score |
| Rating | NVARCHAR(50) | ESRB content rating (e.g., E, M, T) |
| Platform_ID | INT | Foreign key referencing Video_Games_Platforms |
| Genre_ID | INT | Foreign key referencing Video_Games_Genres |
| Publisher_ID | INT | Foreign key referencing Video_Games_Publishers |

# Section 1: JOINs

## What were the Total Sales by Genre and Platform?

```
USE CAP2761C_Project

SELECT
    g.Genre_Name,
    p.Platform_Name,
    ROUND(SUM(vg.Global_Sales), 1) AS Total_Global_Sales
FROM Video_Games_1980_2016 vg
JOIN VD_Genres g ON vg.Genre_ID = g.Genre_ID
JOIN VD_Platforms p ON vg.Platform_ID = p.Platform_ID
GROUP BY g.Genre_Name, p.Platform_Name
ORDER BY Total_Global_Sales DESC;
```

## Which are the Top 5 Gaming Platforms and their Global Sales?

```
USE CAP2761C_Project

SELECT TOP 5
    p.Platform_Name,
    ROUND(SUM(vg.Global_Sales), 1) AS Total_Global_Sales
FROM Video_Games_1980_2016 vg
JOIN VD_Platforms p ON vg.Platform_ID = p.Platform_ID
GROUP BY p.Platform_Name
ORDER BY Total_Global_Sales DESC;
```

## Section 2: SUBQUERIES

## How many games have been released by each of the Top 5 Gaming Platforms?

```
USE CAP2761C_Project

SELECT
    p.Platform_Name,
    COUNT(*) AS Total_Games,
    ROUND(SUM(vg.Global_Sales), 1) AS Total_Global_Sales
FROM Video_Games_1980_2016 vg
JOIN VD_Platforms p ON vg.Platform_ID = p.Platform_ID
WHERE p.Platform_Name IN (
    SELECT TOP 5 p2.Platform_Name
    FROM Video_Games_1980_2016 vg2
    JOIN VD_Platforms p2 ON vg2.Platform_ID = p2.Platform_ID
    GROUP BY p2.Platform_Name
    ORDER BY SUM(vg2.Global_Sales) DESC
)
GROUP BY p.Platform_Name
ORDER BY Total_Global_Sales DESC;
```

## Which are the Top 3 Games of each Platform?

```
USE CAP2761C_Project

;WITH Ranked_Games_By_Platform AS (
    SELECT
        vg.Name,
        p.Platform_Name,
        pub.Publisher_Name,
        vg.Year_of_Release,
        vg.Global_Sales,
        RANK() OVER (PARTITION BY p.Platform_Name ORDER BY
vg.Global_Sales DESC) AS Game_Rank
    FROM Video_Games_1980_2016 vg
    JOIN VD_Platforms p ON vg.Platform_ID = p.Platform_ID
    JOIN VD_Publishers pub ON vg.Publisher_ID = pub.Publisher_ID
)
SELECT Name, Platform_Name, Publisher_Name, Year_of_Release,
Global_Sales
FROM Ranked_Games_By_Platform
WHERE Game_Rank <= 3
ORDER BY Platform_Name, Game_Rank;
```

## Which are the Top 3 Games of each Genre?

```
USE CAP2761C_Project

;WITH Ranked_Games_By_Genre AS (
    SELECT
        vg.Name,
        g.Genre_Name,
        p.Platform_Name,
        pub.Publisher_Name,
        vg.Year_of_Release,
        vg.Global_Sales,
        RANK() OVER (PARTITION BY g.Genre_Name ORDER BY
vg.Global_Sales DESC) AS Game_Rank
    FROM Video_Games_1980_2016 vg
    JOIN VD_Genres g ON vg.Genre_ID = g.Genre_ID
    JOIN VD_Platforms p ON vg.Platform_ID = p.Platform_ID
    JOIN VD_Publishers pub ON vg.Publisher_ID = pub.Publisher_ID
)
SELECT Name, Genre_Name, Platform_Name, Publisher_Name,
Year_of_Release, Global_Sales
FROM Ranked_Games_By_Genre
WHERE Game_Rank <= 3
ORDER BY Genre_Name, Game_Rank;
```

# Section 3: TABLE EXPRESSIONS

## Which is the most Popular Genre per Platform?

```
USE CAP2761C_Project

;WITH Genre_Sales AS (
    SELECT
        p.Platform_Name,
        g.Genre_Name,
        ROUND(SUM(vg.Global_Sales), 1) AS Total_Global_Sales,
        RANK() OVER (PARTITION BY p.Platform_Name ORDER BY
SUM(vg.Global_Sales) DESC) AS Genre_Rank
    FROM Video_Games_1980_2016 vg
    JOIN VD_Platforms p ON vg.Platform_ID = p.Platform_ID
    JOIN VD_Genres g ON vg.Genre_ID = g.Genre_ID
    GROUP BY p.Platform_Name, g.Genre_Name
)
SELECT Platform_Name, Genre_Name, Total_Global_Sales
FROM Genre_Sales
WHERE Genre_Rank = 1
ORDER BY Platform_Name;
```

## Which were the Top 3 Genres per Region 2006-2016?

```
USE CAP2761C_Project

;WITH Genre_Regional_Sales AS (
    SELECT
        g.Genre_Name,
        ROUND(SUM(vg.NA_Sales), 1) AS NA_Sales,
        ROUND(SUM(vg.EU_Sales), 1) AS EU_Sales,
        ROUND(SUM(vg.JP_Sales), 1) AS JP_Sales
    FROM Video_Games_1980_2016 vg
    JOIN VD_Genres g ON vg.Genre_ID = g.Genre_ID
    WHERE vg.Year_of_Release BETWEEN 2006 AND 2016
    GROUP BY g.Genre_Name
),
Ranked AS (
    SELECT Genre_Name, 'NA' AS Region, NA_Sales AS Regional_Sales,
           RANK() OVER (PARTITION BY 'NA' ORDER BY NA_Sales DESC) AS
Genre_Rank
    FROM Genre_Regional_Sales
    UNION ALL
    SELECT Genre_Name, 'EU', EU_Sales,
           RANK() OVER (PARTITION BY 'EU' ORDER BY EU_Sales DESC)
    FROM Genre_Regional_Sales
    UNION ALL
    SELECT Genre_Name, 'JP', JP_Sales,
           RANK() OVER (PARTITION BY 'JP' ORDER BY JP_Sales DESC)
    FROM Genre_Regional_Sales
)
SELECT *
FROM Ranked
WHERE Genre_Rank <= 3
ORDER BY Region, Genre_Rank;
```

# Section 4: SET OPERATORS

## Which were the Top 3 Platforms per Region 2006-2016?

```
USE CAP2761C_Project

;WITH Platform_Regional_Sales AS (
    SELECT
        p.Platform_Name,
        ROUND(SUM(vg.NA_Sales), 1) AS NA_Sales,
        ROUND(SUM(vg.EU_Sales), 1) AS EU_Sales,
        ROUND(SUM(vg.JP_Sales), 1) AS JP_Sales
    FROM Video_Games_1980_2016 vg
    JOIN VD_Platforms p ON vg.Platform_ID = p.Platform_ID
    WHERE vg.Year_of_Release BETWEEN 2006 AND 2016
```

```
    GROUP BY p.Platform_Name
),
Ranked AS (
    SELECT Platform_Name, 'NA' AS Region, NA_Sales AS Regional_Sales,
           RANK() OVER (PARTITION BY 'NA' ORDER BY NA_Sales DESC) AS
Platform_Rank
    FROM Platform_Regional_Sales
    UNION ALL
    SELECT Platform_Name, 'EU', EU_Sales,
           RANK() OVER (PARTITION BY 'EU' ORDER BY EU_Sales DESC)
    FROM Platform_Regional_Sales
    UNION ALL
    SELECT Platform_Name, 'JP', JP_Sales,
           RANK() OVER (PARTITION BY 'JP' ORDER BY JP_Sales DESC)
    FROM Platform_Regional_Sales
)
SELECT *
FROM Ranked
WHERE Platform_Rank <= 3
ORDER BY Region, Platform_Rank;
```

## Which were the Top 3 Publishers per Region and how many games they released 2006-2016?

```
USE CAP2761C_Project

;WITH Publisher_Regional AS (
    SELECT
        pub.Publisher_Name,
        COUNT(*) AS Total_Games,
        ROUND(SUM(vg.NA_Sales), 1) AS NA_Sales,
        ROUND(SUM(vg.EU_Sales), 1) AS EU_Sales,
        ROUND(SUM(vg.JP_Sales), 1) AS JP_Sales
    FROM Video_Games_1980_2016 vg
    JOIN VD_Publishers pub ON vg.Publisher_ID = pub.Publisher_ID
    WHERE vg.Year_of_Release BETWEEN 2006 AND 2016
    GROUP BY pub.Publisher_Name
),
Ranked AS (
    SELECT Publisher_Name, Total_Games, 'NA' AS Region, NA_Sales AS
Regional_Sales,
           RANK() OVER (PARTITION BY 'NA' ORDER BY NA_Sales DESC) AS
Publisher_Rank
    FROM Publisher_Regional
    UNION ALL
    SELECT Publisher_Name, Total_Games, 'EU', EU_Sales,
           RANK() OVER (PARTITION BY 'EU' ORDER BY EU_Sales DESC)
    FROM Publisher_Regional
```

```
    UNION ALL
    SELECT Publisher_Name, Total_Games, 'JP', JP_Sales,
           RANK() OVER (PARTITION BY 'JP' ORDER BY JP_Sales DESC)
    FROM Publisher_Regional
)
SELECT *
FROM Ranked
WHERE Publisher_Rank <= 3
ORDER BY Region, Publisher_Rank;
```

## Section 5: DATA MODIFICATIONS

### How can we rename a specific name from the Genre_Name column?

```
USE CAP2761C_Project

UPDATE VD_Genres
SET Genre_Name = 'Miscellaneous'
WHERE Genre_Name = 'Misc';
```

### How can we bring the Main table back to its original column count?

```
USE CAP2761C_Project

ALTER TABLE Video_Games_1980_2016
DROP COLUMN Genre_ID, Publisher_ID, Platform_ID;
```

## Conclusions

Looking at video game sales from 1980 to 2016, I found some clear patterns in which genres, platforms, and publishers performed the best — and how preferences changed across time and regions.

## Genre Highlights

- Genres like Action, Shooter, and Sports consistently topped global sales.

- Japan stood out with a strong preference for Role-Playing games, while Shooter and Sports dominated in North America and Europe.

- Nintendo outperformed all other Publishers in Top games by Genre.

## Platform Trends

- The top-selling platforms globally included PlayStation 2, Nintendo DS, Wii, PlayStation 3, and Xbox 360.

- Each platform leaned into certain genres — for example, Shooters were huge on Xbox, while Family and Party games thrived on Nintendo systems.

- The PS3 was the most popular gaming Platform across all Regions between 2006 and 2016.

## Regional Differences

- North America and Europe had similar tastes — Sports and Shooter games led the charts.

- In Japan, Role-Playing and Fighting games were much more popular.

- Nintendo showed strong performance across all regions, especially in Japan.

- Only a few publishers — like Nintendo, EA, and Activision — made it into the top 10 across all three major regions.

## Publisher Insights

- Nintendo led in both sales and number of releases — no surprise given their global reach and franchise power.

- Nintendo is the only Publisher with presence and dominance in all Regions.

## Data Cleanup & Modifications

- I made a few adjustments to clean and improve the dataset: for example, I renamed the genre "Misc" to "Miscellaneous".

- I also dropped the ID columns (Genre_ID, Platform_ID, Publisher_ID) after using them for joins — that way, the main table stayed clean and ready for reporting.

## Final Thoughts

Overall, this project showed how structured SQL analysis — with joins, subqueries, CTEs, set operators, and even data modifications — can reveal real business insights. By combining those techniques with visualizations, I was able to find out which were the most popular gaming Platforms, Publishers and Genres across North America, Europe and Japan.