

**Astana IT University**

# **FINAL EXAM**

**Project:** Steam Video Games Database

**Student:** Iskatova Balzhami

**Group:** CS-2312

**Instructor:** Bakayeva Bakyt

**Astana, 2025**

# Database Project Report

*Topic: Steam Video Games*

## 1. Database Design & Conceptualization (Based on Assignment 1)

### ● ER Diagram & Business Case

#### 1. Business Case

The project is based on the creation of a game database that includes data about users, games, purchases and other additional information. The main purpose of this database is to allow users to browse games and purchase them based on the characteristics and reviews of buyers. The goal of the project is to create a reliable database system that efficiently processes all operations.

#### *Key Features:*

- Storing information about games, users, reviews and purchases.
- Extracting and processing data using SQL queries.
- Ensuring data integrity and applying constraints using primary and foreign keys.

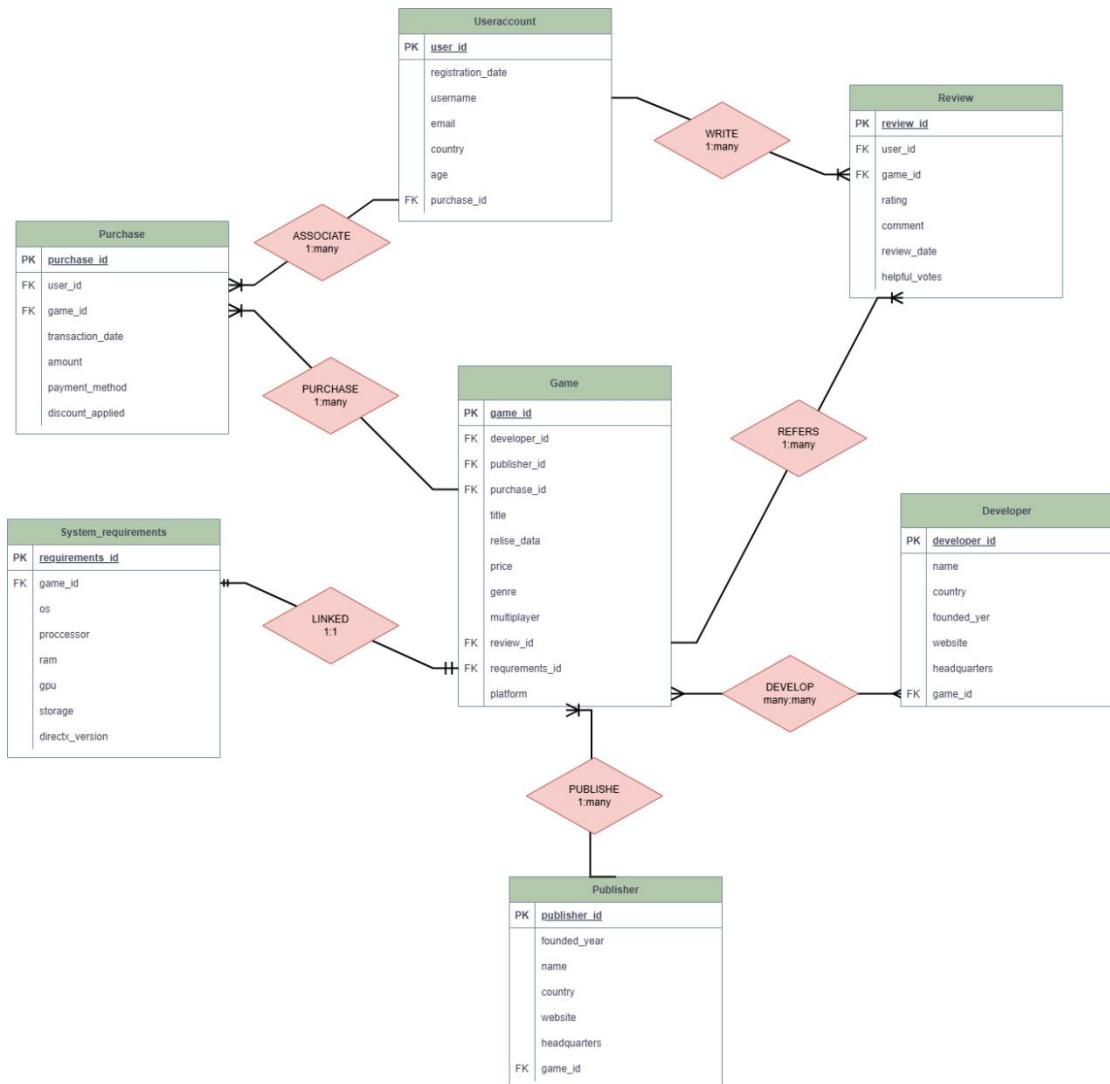
#### *Key Users:*

- End Users: browsing and purchasing games.
- Administrators: managing the platform, analyzing user activity and creating reports.

#### **Problems to solve:**

- Efficient storage and retrieval of data.
- Generating useful information using advanced SQL queries.

## 2.ER Diagram



### **3.Database Implementation & Data Population (Based on Assignment 2)**

#### **● DDL Implementation (SQL Schema Design)**

Create tables with using DDL implementation:

#### **Developer**

Tracks information about game developers and their relationship with published games.

Key fields: developer\_id, name, country, foundation\_year, website

```
Copy Copy to Query Editor

CREATE TABLE Developer (
    developer_id SERIAL PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    country VARCHAR(100) NOT NULL,
    founded_year INT,
    website VARCHAR(255),
    headquarters VARCHAR(255)
);
```

Messages

Query returned successfully in 34 msec.

#### **Publisher**

Stores publisher information.

Key fields: publisher\_id, name, country, foundation\_year, website

```
Copy Copy to Query Editor

CREATE TABLE Publisher (
    publisher_id SERIAL PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    country VARCHAR(100) NOT NULL,
    founded_year INT,
    website VARCHAR(255),
    headquarters VARCHAR(255)
);
```

Messages

Query returned successfully in 35 msec.

#### **Game**

Contains details of the games in the platform's library.

Key fields: game\_id, title, genre, release\_date, price, publisher\_id

[Copy](#) [Copy to Query Editor](#)

```
CREATE TABLE Game (
    game_id SERIAL PRIMARY KEY,
    title VARCHAR(255) NOT NULL,
    genre VARCHAR(100) NOT NULL,
    release_date DATE NOT NULL,
    price DECIMAL(10,2) NOT NULL,
    developer_id INT REFERENCES Developer(developer_id),
    publisher_id INT REFERENCES Publisher(publisher_id),
    platform VARCHAR(100),
    multiplayer BOOLEAN
);
```

#### Messages

Query returned successfully in 33 msec.

## UserAccount

Contains user information for an online game review and purchase platform.

Key fields: user\_id, username, email, password, account\_status, registration\_date

[Copy](#) [Copy to Query Editor](#)

```
CREATE TABLE UserAccount (
    user_id SERIAL PRIMARY KEY,
    username VARCHAR(100) NOT NULL UNIQUE,
    email VARCHAR(255) NOT NULL UNIQUE,
    registration_date DATE NOT NULL,
    country VARCHAR(100),
    age INT
);
```

#### Messages

Query returned successfully in 30 msec.

## Review

Contains user reviews for games.

Key fields: review\_id, user\_id, game\_id, rating, comment, review\_date

[Copy](#) [Copy to Query Editor](#)

```
CREATE TABLE Review (
    review_id SERIAL PRIMARY KEY,
    user_id INT REFERENCES UserAccount(user_id),
    game_id INT REFERENCES Game(game_id),
    rating INT CHECK (rating BETWEEN 1 AND 10),
    comment TEXT,
    review_date DATE NOT NULL,
    helpful_votes INT DEFAULT 0
);
```

#### Messages

Query returned successfully in 38 msec.

## Purchase

Tracks game purchases by users.

Key fields: purchase\_id, user\_id, game\_id, purchase\_date, amount, discount\_applied

[Copy](#) [Copy to Query Editor](#)

```
CREATE TABLE Purchase (
    purchase_id SERIAL PRIMARY KEY,
    user_id INT REFERENCES UserAccount(user_id),
    game_id INT REFERENCES Game(game_id),
    transaction_date DATE NOT NULL,
    amount DECIMAL(10,2) NOT NULL,
    payment_method VARCHAR(50),
    discount_applied BOOLEAN
);
```

#### Messages

Query returned successfully in 31 msec.

## System Requirements

Tracks the minimum and recommended technical specifications required to run each game. Key fields: system\_id, game\_id, os, cpu, gpu, ram,

[Copy](#) [Copy to Query Editor](#)

```
CREATE TABLE System_Requirements (
    requirement_id SERIAL PRIMARY KEY,
    game_id INT REFERENCES Game(game_id),
    os VARCHAR(255) NOT NULL,
    processor VARCHAR(255) NOT NULL,
    ram VARCHAR(50) NOT NULL,
    gpu VARCHAR(255) NOT NULL,
    storage VARCHAR(50),
    directx_version VARCHAR(20)
);
```

### Messages

Query returned successfully in 31 msec.

## **Utilize the ALTER TABLE command to modify structures :**

Data Output Messages Notifications [Query](#) [Query History](#)

```
1 ALTER TABLE Game ADD COLUMN platform VARCHAR(50);
2
3 ALTER TABLE Game ALTER COLUMN price TYPE NUMERIC(10, 2);
4
```

**price**

General [Definition](#) Constraints Variables Security SQL

Data type: numeric

Length/Precision: 10

Scale: 2

Compression: Select compression

Collation:

Statistics:

Storage: MAIN

[Close](#) [Reset](#) [Save](#)

## ● **Data Insertion (At Least 50 Rows Per Table)**

Use INSERT INTO statements with meaningful, realistic data.

My database was based on kaggle.com “Steam Games Dataset” - [Steam Games Dataset](#). I choose only 50 rows, that my database was mostly orginaized and understandable.

## Explanation & Data Types:

developer\_id: Integer, Primary Key – Unique identifier for each developer.

**name:** VARCHAR(100) – Developer's name.  
**country:** VARCHAR(50) – Country of origin.  
**foundation\_year:** YEAR – Year of establishment.  
**website:** VARCHAR(200) – Developer's official website URL.



```

Data Output Messages Notifications Query Query History
1 ✓ INSERT INTO developer (developer_id, name, country, founded_year, website, headquarters) VALUES
2 (1, 'Valve Corporation', 'USA', 1996, 'https://www.valvesoftware.com', 'Bellevue, WA'),
3 (2, 'CD Projekt Red', 'Poland', 2002, 'https://www.cdprojektre.com', 'Warsaw, Poland'),
4 (3, 'Rockstar Games', 'USA', 1998, 'https://www.rockstargames.com', 'New York City, NY'),
5 (4, 'Ubisoft', 'France', 1986, 'https://www.ubisoft.com', 'Montreuil, France'),
6 (5, 'Bethesda Game Studios', 'USA', 2001, 'https://bethesda.net', 'Rockville, MD'),
7 (6, 'Electronic Arts', 'USA', 1982, 'https://www.ea.com', 'Redwood City, CA'),
8 (7, 'Square Enix', 'Japan', 2003, 'https://www.square-enix.com', 'Tokyo, Japan'),
9 (8, 'Bandai Namco', 'Japan', 2006, 'https://www.bandainamcoent.com', 'Tokyo, Japan'),
10 (9, 'Blizzard Entertainment', 'USA', 1991, 'https://www.blizzard.com', 'Irvine, CA'),
11 (10, 'Capcom', 'Japan', 1979, 'https://www.capcom.com', 'Osaka, Japan'),
12 (11, 'Konami', 'Japan', 1969, 'https://www.konami.com', 'Tokyo, Japan'),
13 (12, 'Naughty Dog', 'USA', 1984, 'https://www.naughtydog.com', 'Santa Monica, CA'),
14 (13, 'Insomniac Games', 'USA', 1994, 'https://www.insomniacgames.com', 'Burbank, CA'),
15 (14, 'Guerrilla Games', 'Netherlands', 2000, 'https://www.guerrillagames.com', 'Amsterdam, Netherlands'),
16 (15, 'FromSoftware', 'Japan', 1986, 'https://www.fromsoftware.jp', 'Tokyo, Japan'),
17 (16, 'Sucker Punch Productions', 'USA', 1997, 'https://www.suckerpunch.com', 'Bellevue, WA'),
18 (17, 'Monolith Soft', 'Japan', 1999, 'https://www.monolithsoft.co.jp', 'Tokyo, Japan'),
19 (18, 'Treyarch', 'USA', 1996, 'https://www.treyarch.com', 'Santa Monica, CA'),
20 (19, 'Infinity Ward', 'USA', 2002, 'https://www.infinityward.com', 'Woodland Hills, CA'),
21 (20, 'Respawn Entertainment', 'USA', 2010, 'https://www.respawn.com', 'Los Angeles, CA'),
22 (21, 'Obsidian Entertainment', 'USA', 2003, 'https://www.obsidian.net', 'Irvine, CA'),
23 (22, 'BioWare', 'Canada', 1995, 'https://www.bioware.com', 'Edmonton, Canada'),
24 (23, 'Telltale Games', 'USA', 2004, 'https://www.telltale.com', 'San Rafael, CA'),
25 (24, 'Remedy Entertainment', 'Finland', 1995, 'https://www.remedygames.com', 'Espoo, Finland'),
26 (25, 'PlatinumGames', 'Japan', 2006, 'https://www.platinumgames.com', 'Osaka, Japan'),
27 (26, 'Clover Studio', 'Japan', 2004, 'https://www.cloverstudio.com', 'Osaka, Japan'),
28 (27, 'id Software', 'USA', 1991, 'https://www.idsoftware.com', 'Richardson, TX'),
29 (28, 'Arkane Studios', 'France', 1999, 'https://www.arkane-studios.com', 'Lyon, France'),
30 (29, 'Larian Studios', 'Belgium', 1996, 'https://www.larian.com', 'Ghent, Belgium'),
31 (30, 'Techland', 'Poland', 1991, 'https://www.techland.net', 'Wroclaw, Poland'),
32 Total rows: 50 | Query completed: 00:00:00.158
  
```

CRLF | Ln 33, Col 69

developer_id [PK] integer	name character varying (255)	country character varying (100)	founded_year integer	website character varying (255)	headquarters character varying (255)
1	Valve Corporation	USA	1996	https://www.valvesoftware.com	Bellevue, WA
2	CD Projekt Red	Poland	2002	https://www.cdprojektre.com	Warsaw, Poland
3	Rockstar Games	USA	1998	https://www.rockstargames.com	New York City, NY
4	Ubisoft	France	1986	https://www.ubisoft.com	Montreuil, France
5	Bethesda Game Studios	USA	2001	https://bethesda.net	Rockville, MD
6	Electronic Arts	USA	1982	https://www.ea.com	Redwood City, CA
7	Square Enix	Japan	2003	https://www.square-enix.com	Tokyo, Japan
8	Bandai Namco	Japan	2006	https://www.bandainamcoent.com	Tokyo, Japan
9	Blizzard Entertainment	USA	1991	https://www.blizzard.com	Irvine, CA
10	Capcom	Japan	1979	https://www.capcom.com	Osaka, Japan
11	Konami	Japan	1969	https://www.konami.com	Tokyo, Japan
12	Naughty Dog	USA	1984	https://www.naughtydog.com	Santa Monica, CA
13	Insomniac Games	USA	1994	https://www.insomniacgames.com	Burbank, CA
14	Guerrilla Games	Netherlands	2000	https://www.guerrillagames.com	Amsterdam, Netherlands
15	FromSoftware	Japan	1986	https://www.fromsoftware.jp	Tokyo, Japan
16	Sucker Punch Productions	USA	1997	https://www.suckerpunch.com	Bellevue, WA
17	Monolith Soft	Japan	1999	https://www.monolithsoft.co.jp	Tokyo, Japan
18	Treyarch	USA	1996	https://www.treyarch.com	Santa Monica, CA
19	Infinity Ward	USA	2002	https://www.infinityward.com	Woodland Hills, CA
20	Respawn Entertainment	USA	2010	https://www.respawn.com	Los Angeles, CA
21	Obsidian Entertainment	USA	2003	https://www.obsidian.net	Irvine, CA
22	Ot-Mars	Canada	2005	https://www.ot-mars.com	Edmonton, Canada

## Explanation & Data Types:

**game\_id:** Integer, Primary Key – Unique identifier for each game.  
**title:** VARCHAR(100) – Game title.  
**genre:** VARCHAR(30) – Game genre (e.g., FPS, RPG, Adventure).  
**release\_date:** DATE – Represents the game's release date.  
**price:** DECIMAL(5, 2) – Price of the game with two decimal places for cents.  
**publisher\_id:** Integer, Foreign Key – Refers to the publisher table.

Data Output Messages Notifications Query Query History

```

1 ✓ INSERT INTO game (game_id, title, genre, release_date, developer_id, platform, price) VALUES
2 (1, 'Half-Life', 'FPS', '1998-11-19', 1, 'PC', 19.99),
3 (2, 'The Witcher 3: Wild Hunt', 'RPG', '2015-05-19', 2, 'PC, PS4, Xbox One', 39.99),
4 (3, 'Grand Theft Auto V', 'Action', '2013-09-17', 3, 'PC, PS4, PS5, Xbox', 59.99),
5 (4, 'Assassin's Creed Valhalla', 'Action', '2020-11-10', 4, 'PC, PS5, Xbox', 69.99),
6 (5, 'The Elder Scrolls V: Skyrim', 'RPG', '2011-11-11', 5, 'PC, PS4, Xbox', 29.99),
7 (6, 'FIFA 23', 'Sports', '2022-09-27', 6, 'PC, PS5, Xbox', 59.99),
8 (7, 'Final Fantasy XV', 'RPG', '2016-11-29', 7, 'PC, PS4, Xbox', 49.99),
9 (8, 'Dark Souls III', 'RPG', '2016-03-24', 15, 'PC, PS4, Xbox', 39.99),
10 (9, 'Overwatch', 'FPS', '2016-05-24', 9, 'PC, PS4, Xbox', 39.99),
11 (10, 'Resident Evil 2', 'Horror', '2019-01-25', 10, 'PC, PS4, Xbox', 59.99),
12 (11, 'Metal Gear Solid V', 'Action', '2015-09-01', 36, 'PC, PS4, Xbox', 29.99),
13 (12, 'Uncharted 4: A Thief's End', 'Adventure', '2016-05-10', 12, 'PS4', 29.99),
14 (13, 'Spider-Man', 'Action', '2018-09-07', 13, 'PS4, PS5', 49.99),
15 (14, 'Horizon Zero Dawn', 'RPG', '2017-02-28', 14, 'PC, PS4', 39.99),
16 (15, 'Sekiro: Shadows Die Twice', 'Action', '2019-03-22', 15, 'PC, PS4, Xbox', 59.99),
17 (16, 'Ghost of Tsushima', 'Action', '2020-07-17', 16, 'PS4, PS5', 69.99),
18 (17, 'Xenoblade Chronicles 2', 'RPG', '2017-12-01', 17, 'Nintendo Switch', 49.99),
19 (18, 'Call of Duty: Black Ops 4', 'FPS', '2018-10-12', 18, 'PC, PS4, Xbox', 59.99),
20 (19, 'Titanfall 2', 'FPS', '2016-10-28', 20, 'PC, PS4, Xbox', 19.99),
21 (20, 'Pillars of Eternity II', 'RPG', '2018-05-08', 21, 'PC', 29.99),
22 (21, 'Dragon Age: Inquisition', 'RPG', '2014-11-18', 22, 'PC, PS4, Xbox', 39.99),
23 (22, 'The Walking Dead: Season One', 'Adventure', '2012-04-24', 23, 'PC, PS4, Xbox', 14.99),
24 (23, 'Control', 'Action', '2019-08-27', 24, 'PC, PS4, Xbox', 49.99),
25 (24, 'Bayonetta 2', 'Action', '2014-09-20', 25, 'Nintendo Switch', 49.99),
26 (25, 'DOOM Eternal', 'FPS', '2020-03-20', 27, 'PC, PS4, Xbox', 59.99),
27 (26, 'Dishonored 2', 'Action', '2016-11-11', 28, 'PC, PS4, Xbox', 29.99),
28 (27, 'Divinity: Original Sin II', 'RPG', '2017-09-14', 29, 'PC', 44.99),
29 (28, 'Dying Light', 'Horror', '2015-01-27', 30, 'PC, PS4, Xbox', 29.99),
30 (29, 'Crysis 3', 'FPS', '2013-02-19', 31, 'PC, PS4, Xbox', 19.99),
31 (30, 'Hades', 'Rogue-like', '2020-09-17', 32, 'PC, Nintendo Switch', 24.99),
Total rows: 31 Query complete 00:00:00.052 CRLF Ln 52, Col 1
```

game_id [PK] integer	title character varying (255)	genre character varying (100)	release_date date	price numeric (10,2)	developer_id integer	publisher_id integer	platform character varying (100)	multipla boolean
1	1 Half-Life	FPS	1998-11-19	19.99	1	[null]	PC	[null]
2	2 The Witcher 3: Wild Hunt	RPG	2015-05-19	39.99	2	[null]	PC, PS4, Xbox One	[null]
3	3 Grand Theft Auto V	Action	2013-09-17	59.99	3	[null]	PC, PS4, PS5, Xbox	[null]
4	4 Assassin's Creed Valhalla	Action	2020-11-10	69.99	4	[null]	PC, PS5, Xbox	[null]
5	5 The Elder Scrolls V: Skyrim	RPG	2011-11-11	29.99	5	[null]	PC, PS4, Xbox	[null]
6	6 FIFA 23	Sports	2022-09-27	59.99	6	[null]	PC, PS5, Xbox	[null]
7	7 Final Fantasy XV	RPG	2016-11-29	49.99	7	[null]	PC, PS4, Xbox	[null]
8	8 Dark Souls III	RPG	2016-03-24	39.99	15	[null]	PC, PS4, Xbox	[null]
9	9 Overwatch	FPS	2016-05-24	39.99	9	[null]	PC, PS4, Xbox	[null]
10	10 Resident Evil 2	Horror	2019-01-25	59.99	10	[null]	PC, PS4, Xbox	[null]
11	11 Metal Gear Solid V	Action	2015-09-01	29.99	36	[null]	PC, PS4, Xbox	[null]
12	12 Uncharted 4: A Thief's End	Adventure	2016-05-10	29.99	12	[null]	PS4	[null]
13	13 Spider-Man	Action	2018-09-07	49.99	13	[null]	PS4, PS5	[null]
14	14 Horizon Zero Dawn	RPG	2017-02-28	39.99	14	[null]	PC, PS4	[null]
15	15 Sekiro: Shadows Die Twice	Action	2019-03-22	59.99	15	[null]	PC, PS4, Xbox	[null]
16	16 Ghost of Tsushima	Action	2020-07-17	69.99	16	[null]	PS4, PS5	[null]
17	17 Xenoblade Chronicles 2	RPG	2017-12-01	49.99	17	[null]	Nintendo Switch	[null]
18	18 Call of Duty: Black Ops 4	FPS	2018-10-12	59.99	18	[null]	PC, PS4, Xbox	[null]
19	19 Titanfall 2	FPS	2016-10-28	19.99	20	[null]	PC, PS4, Xbox	[null]
20	20 Pillars of Eternity II	RPG	2018-05-08	29.99	21	[null]	PC	[null]
21	21 Dragon Age: Inquisition	RPG	2014-11-18					

Total rows: 50 Query complete 00:00:00.044 CRLF Ln 1, Col 20

✓ Successfully run. Total query runtime: 44 msec. 50 rows affected.

## Explanation & Data Types:

user\_id: Integer, Primary Key – Ensures unique identification for each user.

username: VARCHAR(50) – Holds the username; constrained to 50 characters.

email: VARCHAR(100) – Contains the user's email address.

account\_status: VARCHAR(10) – Can be 'Active', 'Inactive', or 'Premium'.

registration\_date: DATE – Represents the date the user registered.

Data Output Messages Notifications Query Query History

```

1 ✓ INSERT INTO useraccount (user_id, username, email, registration_date, country) VALUES
2 (101, 'john_doe', 'john.doe@example.com', '2022-01-01', 'USA'),
3 (102, 'jane_smith', 'jane.smith@example.com', '2022-01-02', 'Canada'),
4 (103, 'michael_jones', 'michael.jones@example.com', '2022-01-03', 'UK'),
5 (104, 'sarah_brown', 'sarah.brown@example.com', '2022-01-04', 'Germany'),
6 (105, 'david_miller', 'david.miller@example.com', '2022-01-05', 'France'),
7 (106, 'emily_davis', 'emily.davis@example.com', '2022-01-06', 'Japan'),
8 (107, 'chris_wilson', 'chris.wilson@example.com', '2022-01-07', 'Australia'),
9 (108, 'anna_taylor', 'anna.taylor@example.com', '2022-01-08', 'Russia'),
10 (109, 'james_white', 'james.white@example.com', '2022-01-09', 'Brazil'),
11 (110, 'lisa_harris', 'lisa.harris@example.com', '2022-01-10', 'India'),
12 (111, 'kevin_clark', 'kevin.clark@example.com', '2022-01-11', 'China'),
13 (112, 'laura_lee', 'laura.lee@example.com', '2022-01-12', 'Italy'),
14 (113, 'robert_hall', 'robert.hall@example.com', '2022-01-13', 'Mexico'),
15 (114, 'sophia_wright', 'sophia.wright@example.com', '2022-01-14', 'Spain'),
16 (115, 'daniel_thompson', 'daniel.thompson@example.com', '2022-01-15', 'Netherlands'),
17 (116, 'amelia_walker', 'amelia.walker@example.com', '2022-01-16', 'Sweden'),
18 (117, 'jack_evans', 'jack.evans@example.com', '2022-01-17', 'Norway'),
19 (118, 'olivia_moore', 'olivia.moore@example.com', '2022-01-18', 'Denmark'),
20 (119, 'ethan_hill', 'ethan.hill@example.com', '2022-01-19', 'Finland'),
21 (120, 'isabella_adams', 'isabella.adams@example.com', '2022-01-20', 'Poland'),
22 (121, 'alex_martin', 'alex.martin@example.com', '2022-01-21', 'USA'),
23 (122, 'mia_scott', 'mia.scott@example.com', '2022-01-22', 'Canada'),
24 (123, 'lucas_kelly', 'lucas.kelly@example.com', '2022-01-23', 'UK'),
25 (124, 'grace_carter', 'grace.carter@example.com', '2022-01-24', 'Germany'),
26 (125, 'logan_brown', 'logan.brown@example.com', '2022-01-25', 'France'),
27 (126, 'zoe_anderson', 'zoe.anderson@example.com', '2022-01-26', 'Japan'),
28 (127, 'henry_taylor', 'henry.taylor@example.com', '2022-01-27', 'Australia'),
29 (128, 'ellie_wilson', 'ellie.wilson@example.com', '2022-01-28', 'Russia'),
30 (129, 'ryan_lee', 'ryan.lee@example.com', '2022-01-29', 'Brazil'),
31 (130, 'chloe_jackson', 'chloe.jackson@example.com', '2022-01-30', 'India'),
Total rows: 31 Query complete 00:00:00.043
```

CRLF Ln 10, Col 73

	user_id [PK] integer	username character varying (100)	email character varying (255)	registration_date date	country character varying (100)	age integer
1	101	john_doe	john.doe@example.com	2022-01-01	USA	[null]
2	102	jane_smith	jane.smith@example.com	2022-01-02	Canada	[null]
3	103	michael_jones	michael.jones@example.com	2022-01-03	UK	[null]
4	104	sarah_brown	sarah.brown@example.com	2022-01-04	Germany	[null]
5	105	david_miller	david.miller@example.com	2022-01-05	France	[null]
6	106	emily_davis	emily.davis@example.com	2022-01-06	Japan	[null]
7	107	chris_wilson	chris.wilson@example.com	2022-01-07	Australia	[null]
8	108	anna_taylor	anna.taylor@example.com	2022-01-08	Russia	[null]
9	109	james_white	james.white@example.com	2022-01-09	Brazil	[null]
10	110	lisa_harris	lisa.harris@example.com	2022-01-10	India	[null]
11	111	kevin_clark	kevin.clark@example.com	2022-01-11	China	[null]
12	112	laura_lee	laura.lee@example.com	2022-01-12	Italy	[null]
13	113	robert_hall	robert.hall@example.com	2022-01-13	Mexico	[null]
14	114	sophia_wright	sophia.wright@example.com	2022-01-14	Spain	[null]
15	115	daniel_thompson	daniel.thompson@example.com	2022-01-15	Netherlands	[null]
16	116	amelia_walker	amelia.walker@example.com	2022-01-16	Sweden	[null]
17	117	jack_evans	jack.evans@example.com	2022-01-17	Norway	[null]
18	118	olivia_moore	olivia.moore@example.com	2022-01-18	Denmark	[null]
19	119	ethan_hill	ethan.hill@example.com	2022-01-19	Finland	[null]
20	120	isabella_adams	isabella.adams@example.com	2022-01-20	Poland	[null]
21	121	alex_martin	alex.martin@example.com	2022-01-21	USA	[null]
...	...	...	...	...	...	...
Total rows: 50	Query complete 00:00:00.045					

CRLF Ln 1, Col 27

## Explanation & Data Types:

review\_id: Integer, Primary Key – Unique identifier for each review.  
 user\_id: Integer, Foreign Key – Refers to the UserAccount table.  
 game\_id: Integer, Foreign Key – Refers to the Game table.  
 rating: Integer – Rating from 1 to 5.  
 comment: TEXT – User's detailed comment about the game.  
 review\_date: DATE – Date when the review was posted.

Data Output Messages Notifications Query Query History X

```

1 ✓ INSERT INTO review (review_id, user_id, game_id, rating, comment, review_date, helpful_votes)
2 VALUES
3 (1, 3, 2, 5, 'Amazing game, loved every bit of it!', '2024-10-12', 85),
4 (2, 5, 4, 4, 'Great game, but could be better with more updates.', '2023-07-15', 62),
5 (3, 6, 1, 3, 'Average experience, not what I expected.', '2022-03-18', 45),
6 (4, 2, 3, 2, 'Disappointing. Bugs and crashes all the time.', '2021-09-25', 30),
7 (5, 4, 5, 5, 'Masterpiece! Highly recommended.', '2020-12-01', 100),
8 (6, 1, 2, 4, 'Pretty fun, but lacks content in the endgame.', '2024-08-22', 75),
9 (7, 7, 6, 1, 'Terrible. Waste of money.', '2023-11-14', 10),
10 (8, 3, 7, 5, 'Fantastic graphics and gameplay!', '2022-05-04', 95),
11 (9, 8, 8, 3, 'Not bad, but needs improvement.', '2021-04-30', 55),
12 (10, 10, 9, 4, 'Overall a good experience.', '2020-07-20', 70),
13 (11, 9, 10, 5, 'One of the best games I have ever played!', '2024-01-10', 90),
14 (12, 2, 11, 2, 'Too short and repetitive.', '2023-03-11', 15),
15 (13, 3, 12, 3, 'Nice concept but poorly executed.', '2022-11-05', 25),
16 (14, 4, 13, 4, 'Fun multiplayer experience.', '2021-06-10', 65),
17 (15, 5, 14, 1, 'Crashes every time I play. Avoid.', '2020-03-02', 5),
18 (16, 6, 15, 4, 'Good story, enjoyable.', '2024-09-10', 72),
19 (17, 7, 16, 5, 'Loved it! Played for hours.', '2023-08-15', 88),
20 (18, 8, 17, 3, 'Needs better controls.', '2022-01-22', 48),
21 (19, 9, 18, 2, 'Too many microtransactions.', '2021-12-09', 20),
22 (20, 10, 19, 4, 'Solid game overall.', '2020-11-11', 60),
23 (21, 11, 20, 4, 'Good game with minor bugs.', '2024-06-22', 50),
24 (22, 12, 21, 5, 'Absolutely loved this game!', '2023-02-28', 90),
25 (23, 13, 22, 2, 'Not worth the price.', '2022-07-30', 20),
26 (24, 14, 23, 3, 'Mediocre at best.', '2021-01-15', 35),
27 (25, 15, 24, 4, 'Fun with friends!', '2020-04-20', 75),
28 (26, 16, 25, 1, 'Terrible graphics and boring gameplay.', '2024-03-10', 5),
29 (27, 17, 26, 5, 'One of my favorite games!', '2023-08-05', 100),
30 (28, 18, 27, 3, 'Mixed feelings about this one.', '2022-05-25', 40),
31 (29, 19, 28, 2, 'The concept is good but poorly executed.', '2021-09-18', 30),

```

Total rows: 50 Query complete 00:00:00.045 CRLF Ln 1, Col 1

review_id [PK] integer	user_id integer	game_id integer	rating integer	comment text	review_date date	helpful_votes integer
1	2	101	1	3 Amazing gameplay and graphics!	2024-06-08	37
2	3	101	11	3 Not recommended. Needs improvements.	2024-06-01	29
3	4	101	21	3 Amazing gameplay and graphics!	2024-05-04	42
4	5	101	31	1 Not recommended. Needs improvements.	2024-05-21	1
5	6	101	41	6 Amazing gameplay and graphics!	2024-04-02	30
6	7	102	2	5 Not recommended. Needs improvements.	2024-08-17	9
7	8	102	12	2 Good, but with some bugs.	2024-07-27	26
8	9	102	22	2 Good, but with some bugs.	2024-05-03	31
9	10	102	32	5 Amazing gameplay and graphics!	2024-11-27	12
10	11	102	42	5 Not recommended. Needs improvements.	2024-03-06	39
11	12	103	3	6 Amazing gameplay and graphics!	2024-07-06	30
12	13	103	13	4 Good, but with some bugs.	2024-07-18	1
13	14	103	23	5 Good, but with some bugs.	2024-03-04	34
14	15	103	33	4 Amazing gameplay and graphics!	2024-08-24	18
15	16	103	43	1 Not recommended. Needs improvements.	2024-04-08	28
16	17	104	4	4 Good, but with some bugs.	2024-07-02	15
17	18	104	14	3 Good, but with some bugs.	2024-05-25	9
18	19	104	24	6 Good, but with some bugs.	2024-07-05	25
19	20	104	34	1 Amazing gameplay and graphics!	2024-12-06	24
20	21	104	44	2 Good, but with some bugs.	2024-05-18	45
21	22	105	5	4 Not recommended. Needs improvements.	2024-11-10	13
22	23	105	15	1	2024-11-09	26

Total rows: 50 Query complete 00:00:00.045 CRLF Ln 1, Col 1

## Explanation & Data Types:

requirement\_id: Integer, Primary Key – Unique identifier for each system requirement entry.

game\_id: Integer, Foreign Key – Refers to the Game table.

os: VARCHAR(50) – Operating system (e.g., Windows 10, MacOS).

cpu: VARCHAR(100) – Processor model required to run the game.

gpu: VARCHAR(100) – Graphics card model required.

ram: VARCHAR(20) – Amount of RAM required (formatted as '8 GB', '16 GB').

storage: VARCHAR(20) – Disk space required (formatted as '50 GB', '100 GB').

Data Output Messages Notifications Query Query History

```

1 ✓ INSERT INTO system_requirements (requirement_id, game_id, os, processor, ram, gpu, storage, directx_version) VALUES
2   (1, 1, 'Windows 10', 'Intel Core i5', '8 GB', 'NVIDIA GTX 1050', '50 GB', 'DirectX 11'),
3   (2, 2, 'Windows 10', 'Intel Core i7', '16 GB', 'NVIDIA GTX 1080', '100 GB', 'DirectX 12'),
4   (3, 3, 'Windows 7', 'AMD Ryzen 5', '8 GB', 'AMD RX 580', '75 GB', 'DirectX 11'),
5   (4, 4, 'Windows 10', 'Intel Core i9', '32 GB', 'NVIDIA RTX 3070', '150 GB', 'DirectX 12'),
6   (5, 5, 'Windows 8', 'Intel Core i3', '4 GB', 'NVIDIA GTX 950', '30 GB', 'DirectX 11'),
7   (6, 6, 'Windows 10', 'AMD Ryzen 7', '16 GB', 'AMD RX 5700 XT', '120 GB', 'DirectX 12'),
8   (7, 7, 'Windows 7', 'Intel Core i5', '8 GB', 'NVIDIA GTX 970', '60 GB', 'DirectX 11'),
9   (8, 8, 'Windows 10', 'Intel Core i7', '16 GB', 'NVIDIA GTX 1660', '90 GB', 'DirectX 12'),
10  (9, 9, 'Windows 8', 'AMD Ryzen 5', '8 GB', 'AMD RX 560', '50 GB', 'DirectX 11'),
11  (10, 10, 'Windows 8', 'Intel Core i5', '8 GB', 'NVIDIA GTX 1050 Ti', '40 GB', 'DirectX 11'),
12  (11, 11, 'Windows 10', 'Intel Core i7', '16 GB', 'NVIDIA GTX 1070', '80 GB', 'DirectX 12'),
13  (12, 12, 'Windows 10', 'AMD Ryzen 3', '4 GB', 'AMD RX 550', '20 GB', 'DirectX 11'),
14  (13, 13, 'Windows 10', 'Intel Core i9', '32 GB', 'NVIDIA RTX 3080', '200 GB', 'DirectX 12'),
15  (14, 14, 'Windows 7', 'Intel Core i5', '8 GB', 'NVIDIA GTX 760', '45 GB', 'DirectX 11'),
16  (15, 15, 'Windows 10', 'AMD Ryzen 7', '16 GB', 'AMD RX 6800 XT', '150 GB', 'DirectX 12'),
17  (16, 16, 'Windows 10', 'Intel Core i3', '4 GB', 'NVIDIA GTX 660', '35 GB', 'DirectX 11'),
18  (17, 17, 'Windows 8', 'Intel Core i5', '8 GB', 'NVIDIA GTX 770', '50 GB', 'DirectX 11'),
19  (18, 18, 'Windows 10', 'AMD Ryzen 5', '8 GB', 'AMD RX 570', '70 GB', 'DirectX 12'),
20  (19, 19, 'Windows 7', 'Intel Core i7', '16 GB', 'NVIDIA GTX 980', '100 GB', 'DirectX 11'),
21  (20, 20, 'Windows 10', 'Intel Core i9', '32 GB', 'NVIDIA RTX 2080', '180 GB', 'DirectX 12'),
22  (21, 21, 'Windows 10', 'Intel Core i5', '8 GB', 'NVIDIA GTX 1650', '50 GB', 'DirectX 11'),
23  (22, 22, 'Windows 10', 'AMD Ryzen 7', '16 GB', 'AMD RX 5600 XT', '110 GB', 'DirectX 12'),
24  (23, 23, 'Windows 10', 'Intel Core i7', '16 GB', 'NVIDIA GTX 1660 Ti', '90 GB', 'DirectX 12'),
25  (24, 24, 'Windows 10', 'AMD Ryzen 5', '8 GB', 'AMD RX 470', '60 GB', 'DirectX 11'),
26  (25, 25, 'Windows 7', 'Intel Core i3', '4 GB', 'NVIDIA GTX 750 Ti', '25 GB', 'DirectX 11'),
27  (26, 26, 'Windows 10', 'Intel Core i7', '16 GB', 'NVIDIA GTX 1070 Ti', '85 GB', 'DirectX 12'),
28  (27, 27, 'Windows 10', 'AMD Ryzen 9', '32 GB', 'AMD RX 6900 XT', '200 GB', 'DirectX 12'),
29  (28, 28, 'Windows 8', 'Intel Core i5', '8 GB', 'NVIDIA GTX 960', '60 GB', 'DirectX 11'),
30  (29, 29, 'Windows 10', 'Intel Core i9', '32 GB', 'NVIDIA RTX 3090', '250 GB', 'DirectX 12'),
31  (30, 30, 'Windows 7', 'Intel Core i5', '8 GB', 'NVIDIA GTX 650', '40 GB', 'DirectX 11'),

```

Total rows: 31 Query complete 00:00:00.057

	requirement_id [PK] integer	game_id integer	os character varying (255)	processor character varying (255)	ram character varying (50)	gpu character varying (255)	storage character varying (50)	directx_version character varying (20)	CRLF	Ln 52, Col 1
1	1	1	Windows 10	Intel Core i5	8 GB	NVIDIA GTX 1050	50 GB	DirectX 11		
2	2	2	Windows 10	Intel Core i7	16 GB	NVIDIA GTX 1080	100 GB	DirectX 12		
3	3	3	Windows 7	AMD Ryzen 5	8 GB	AMD RX 580	75 GB	DirectX 11		
4	4	4	Windows 10	Intel Core i9	32 GB	NVIDIA RTX 3070	150 GB	DirectX 12		
5	5	5	Windows 8	Intel Core i3	4 GB	NVIDIA GTX 950	30 GB	DirectX 11		
6	6	6	Windows 10	AMD Ryzen 7	16 GB	AMD RX 5700 XT	120 GB	DirectX 12		
7	7	7	Windows 7	Intel Core i5	8 GB	NVIDIA GTX 970	60 GB	DirectX 11		
8	8	8	Windows 10	Intel Core i7	16 GB	NVIDIA GTX 1660	90 GB	DirectX 12		
9	9	9	Windows 10	AMD Ryzen 5	8 GB	AMD RX 560	50 GB	DirectX 11		
10	10	10	Windows 8	Intel Core i5	8 GB	NVIDIA GTX 1050 Ti	40 GB	DirectX 11		
11	11	11	Windows 10	Intel Core i7	16 GB	NVIDIA GTX 1070	80 GB	DirectX 12		
12	12	12	Windows 10	AMD Ryzen 3	4 GB	AMD RX 550	20 GB	DirectX 11		
13	13	13	Windows 10	Intel Core i9	32 GB	NVIDIA RTX 3080	200 GB	DirectX 12		
14	14	14	Windows 7	Intel Core i5	8 GB	NVIDIA GTX 760	45 GB	DirectX 11		
15	15	15	Windows 10	AMD Ryzen 7	16 GB	AMD RX 6800 XT	150 GB	DirectX 12		
16	16	16	Windows 10	Intel Core i3	4 GB	NVIDIA GTX 660	35 GB	DirectX 11		
17	17	17	Windows 8	Intel Core i5	8 GB	NVIDIA GTX 770	50 GB	DirectX 11		
18	18	18	Windows 10	AMD Ryzen 5	8 GB	AMD RX 570	70 GB	DirectX 12		
19	19	19	Windows 7	Intel Core i7	16 GB	NVIDIA GTX 980	100 GB	DirectX 11		
20	20	20	Windows 10	Intel Core i9	32 GB	NVIDIA RTX 2080	180 GB	DirectX 12		
21	21	21	Windows 10	Intel Core i5	8 GB	NVIDIA GTX 1650	50 GB	DirectX 11		

Total rows: 50 Query complete 00:00:00.052

CRLF Ln 1, Col 35

- Relationships and Coordinality:

**Relationship:** UserAccount - Review

Details: One user can write multiple reviews, but each review is associated with one user.

**Coordinality:** One-to-Many

Foreign Key: review.user\_id - useraccount.user\_id

**Relationship:** Game - Review

Details: One game can have multiple reviews, but each review refers to one game.

**Coordinality:** One-to-Many

Foreign Key: review.game\_id - game.game\_id

**Relationship:** Game - SystemRequirements

Details: Each game has specific system requirements, and each system requirement entry is linked to one game.

**Coordinality:** One-to-One

Foreign Key: system\_requirements.game\_id - game.game\_id

**Relationship:** Game - Purchase

Details: One game can be purchased multiple times by different users, but each purchase refers to one game.

**Coordinality:** One-to-Many

Foreign Key: purchase.game\_id - game.game\_id

**Relationship:** UserAccount - Purchase

Details: One user can make multiple purchases, but each purchase is associated with one user.

**Coordinality:** One-to-Many

Foreign Key: purchase.user\_id - useraccount.user\_id

**Relationship:** Game - Developer

Details: A game can be developed by multiple developers, and a developer can work on multiple games

**Coordinality:** Many-to-Many

.

**Relationship:** Game - Publisher

Details: One publisher can publish multiple games, but each game is published by only one publisher.

**Coordinality:** Many-to-One

Foreign Key: game.publisher\_id - publisher.publisher\_id

## **Here I use correct 3NF form.**

Normalization Explanation (Up to 3NF)

First Normal Form (1NF):

Each column contains atomic values.

No repeating groups or arrays are allowed in any table.

Second Normal Form (2NF):

All non-key attributes are fully dependent on the primary key.

Tables are organized to remove partial dependencies.

Third Normal Form (3NF):

No transitive dependencies exist

All attributes are only dependent on the primary key.

## **2. Data Querying & Manipulation (Based on Assignment 3)**

Here I used queries (WHERE filtering, String functions, Date functions, Modifying data, Deleting data), for convenient data processing:

### **1. WHERE Filtration (5 queries)**

Specifies a condition for repeated execution of a statement or block of SQL statements. The statement is called in a loop until another condition is specified.

In these tasks, the loop filters data based on the conditions specified to it.

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT user_id, username, country
2   FROM UserAccount
3   WHERE country = 'USA';
```

[Data Output](#) [Messages](#) [Notifications](#) [Query](#) [Query History](#)

[SQL](#)

	user_id [PK] integer	username character varying (100)	country character varying (100)
1	101	john_doe	USA
2	121	alex_martin	USA
3	141	isaac_bailey	USA

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT game_id, title, release_date
2   FROM Game
3   WHERE release_date > '2020-01-01';
4
```

	game_id [PK] integer	title character varying (255)	release_date date
1	4	Assassin's Creed Valhalla	2020-11-10
2	6	FIFA 23	2022-09-27
3	16	Ghost of Tsushima	2020-07-17
4	25	DOOM Eternal	2020-03-20
5	30	Hades	2020-09-17
6	34	Hitman 3	2021-01-20
7	40	Returnal	2021-04-30
8	44	Cyberpunk 2077	2020-12-10
9	47	Horizon Forbidden West	2022-02-18

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT purchase_id, user_id, amount
2   FROM Purchase
3   WHERE amount > 100;
```

2.

	<b>purchase_id</b> [PK] integer	<b>user_id</b> integer	<b>amount</b> numeric (10,2)
1	21	141	108.36
2	27	141	109.71
3	29	141	106.34
4	30	141	109.61
5	44	141	108.67

Data Output Messages Notifications Query Query History

```

1 ▾ SELECT review_id, rating, comment
2 FROM Review
3 WHERE rating >= 4.5;
4

```

3.

	<b>review_id</b> [PK] integer	<b>rating</b> integer	<b>comment</b> text
1	6	6	Amazing gameplay and graphics!
2	7	5	Not recommended. Needs improvements.
3	10	5	Amazing gameplay and graphics!
4	11	5	Not recommended. Needs improvements.
5	12	6	Amazing gameplay and graphics!
6	14	5	Good, but with some bugs.
7	19	6	Good, but with some bugs.
8	28	5	Not recommended. Needs improvements.
9	29	5	Good, but with some bugs.
10	33	5	Amazing gameplay and graphics!
11	36	6	Not recommended. Needs improvements.
12	38	5	Amazing gameplay and graphics!
13	39	5	Not recommended. Needs improvements.
14	44	6	Not recommended. Needs improvements.
15	45	5	Not recommended. Needs improvements.
16	49	5	Good, but with some bugs.

Data Output Messages Notifications Query Query History

```
1 ▾ SELECT requirement_id, game_id, ram
2   FROM system_requirements
3   WHERE REGEXP_REPLACE(ram, '[^0-9]', '', 'g')::INTEGER < 8;
4.
```

4.

	requirement_id [PK] integer	game_id integer	ram character varying (50)
1	5	5	4 GB
2	12	12	4 GB
3	16	16	4 GB
4	25	25	4 GB
5	33	33	4 GB
6	34	34	4 GB
7	42	42	4 GB

## 2. String Functions (5 queries)

String functions in SQL are built-in functions that allow users to manipulate character data in a variety of ways. These functions can perform tasks such as formatting text, extracting substrings, and searching for specific patterns in a string.

In this task, we are joining different columns together

	Data Output	Messages	Notifications	Query	Query History
1	▼	<pre>SELECT game_id, UPPER(title) AS uppercase_title</pre>			
2		<pre>FROM Game;</pre>			
3					
1	game_id [PK] integer	uppercase_title text			
2	1	HALF-LIFE			
3	2	THE WITCHER 3: WILD HUNT			
4	3	GRAND THEFT AUTO V			
5	4	ASSASSIN'S CREED VALHALLA			
6	5	THE ELDER SCROLLS V: SKYRIM			
7	6	FIFA 23			
8	7	FINAL FANTASY XV			
9	8	DARK SOULS III			
10	9	OVERWATCH			
11	10	RESIDENT EVIL 2			
12	11	METAL GEAR SOLID V			
13	12	UNCHARTED 4: A THIEF'S END			
14	13	SPIDER-MAN			
15	14	HORIZON ZERO DAWN			
16	15	SEKIRO: SHADOWS DIE TWICE			
17	16	GHOST OF TSUSHIMA			
18	17	XENOBLADE CHRONICLES 2			
19	18	CALL OF DUTY: BLACK OPS 4			
20	19	TITANFALL 2			
21	20	PILLARS OF ETERNITY II			
22	21	DRAGON AGE: INQUISITION			
23	22	THE WALKING DEAD: SEASON ONE			
Total rows: 50		Query complete 00:00:00.080			

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT user_id, LEFT(username, 3) AS username_prefix
2   FROM UserAccount;
3   |
```

	user_id [PK] integer	username_prefix text
1	101	joh
2	102	jan
3	103	mic
4	104	sar
5	105	dav
6	106	emi
7	107	chr
8	108	ann
9	109	jam
10	110	lis
11	111	kev
12	112	lau
13	113	rob
14	114	sop
15	115	dan
16	116	ame
17	117	jac
18	118	oli
19	119	eth
20	120	isa
21	121	ale
22	122	...

Total rows: 50

Query complete 00:00:00.054

Data Output	Messages	Notifications	<u>Query</u>	Query History
<pre> 1 ✓ SELECT user_id, CONCAT(username, ' (', email, ')') AS user_details 2   FROM UserAccount; 3 </pre>				
	<b>user_id</b> [PK] integer	<b>user_details</b> text		🔒
1	101	john_doe (john.doe@example.com)		
2	102	jane_smith (jane.smith@example.com)		
3	103	michael_jones (michael.jones@example.com)		
4	104	sarah_brown (sarah.brown@example.com)		
5	105	david_miller (david.miller@example.com)		
6	106	emily_davis (emily.davis@example.com)		
7	107	chris_wilson (chris.wilson@example.com)		
8	108	anna_taylor (anna.taylor@example.com)		
9	109	james_white (james.white@example.com)		
10	110	lisa_harris (lisa.harris@example.com)		
11	111	kevin_clark (kevin.clark@example.com)		
12	112	laura_lee (laura.lee@example.com)		
13	113	robert_hall (robert.hall@example.com)		
14	114	sophia_wright (sophia.wright@example.com)		
15	115	daniel_thompson (daniel.thompson@example.com)		
16	116	amelia_walker (amelia.walker@example.com)		
17	117	jack_evans (jack.evans@example.com)		
18	118	olivia_moore (olivia.moore@example.com)		
19	119	ethan_hill (ethan.hill@example.com)		
20	120	isabella_adams (isabella.adams@example.com)		
21	121	alex_martin (alex.martin@example.com)		
22	122	-----		
Total rows: 50		Query complete 00:00:00.052		

Data Output Messages Notifications Query Query History

```
1 ▾ SELECT game_id, title, LENGTH(title) AS title_length  
2 FROM Game;  
3
```

	game_id [PK] integer	title character varying (255)	title_length integer
1	1	Half-Life	9
2	2	The Witcher 3: Wild Hunt	24
3	3	Grand Theft Auto V	18
4	4	Assassin's Creed Valhalla	25
5	5	The Elder Scrolls V: Skyrim	27
6	6	FIFA 23	7
7	7	Final Fantasy XV	16
8	8	Dark Souls III	14
9	9	Overwatch	9
10	10	Resident Evil 2	15
11	11	Metal Gear Solid V	18
12	12	Uncharted 4: A Thief's End	26
13	13	Spider-Man	10
14	14	Horizon Zero Dawn	17
15	15	Sekiro: Shadows Die Twice	25
16	16	Ghost of Tsushima	17
17	17	Xenoblade Chronicles 2	22
18	18	Call of Duty: Black Ops 4	25
19	19	Titanfall 2	11
20	20	Pillars of Eternity II	22
21	21	Dragon Age: Inquisition	23
22	22	THE WITCHER 3: WILD HUNT	22

Total rows: 50

Query complete 00:00:00.053

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT review_id, REPLACE(comment, ' ', '_') AS modified_comment
2   FROM Review;
3
```

	review_id [PK] integer	modified_comment text	🔒
1	2	Amazing_gameplay_and_graphics!	
2	3	Not_recommended._Needs_improvements.	
3	4	Amazing_gameplay_and_graphics!	
4	5	Not_recommended._Needs_improvements.	
5	6	Amazing_gameplay_and_graphics!	
6	7	Not_recommended._Needs_improvements.	
7	8	Good,_but_with_some_bugs.	
8	9	Good,_but_with_some_bugs.	
9	10	Amazing_gameplay_and_graphics!	
10	11	Not_recommended._Needs_improvements.	
11	12	Amazing_gameplay_and_graphics!	
12	13	Good,_but_with_some_bugs.	
13	14	Good,_but_with_some_bugs.	
14	15	Amazing_gameplay_and_graphics!	
15	16	Not_recommended._Needs_improvements.	
16	17	Good,_but_with_some_bugs.	
17	18	Good,_but_with_some_bugs.	
18	19	Good,_but_with_some_bugs.	
19	20	Amazing_gameplay_and_graphics!	
20	21	Good,_but_with_some_bugs.	
21	22	Not_recommended._Needs_improvements.	
...	...	...	...

Total rows: 50

Query complete 00:00:00.055

### 3. Date Functions (5 queries)

The ADDDATE() function does what its name suggests - adds an interval to a date or date time.

In this task, we counted the time of registered users, their interval activity over 30 days. Years of game creation, and release dates.

Data Output Messages Notifications Query Query History

```
1 ✓ SELECT user_id, username, AGE(NOW(), registration_date) AS account_age
2   FROM UserAccount;
3   |
```

	user_id [PK] integer	username character varying (100)	account_age interval
1	101	john_doe	3 years 1 mon 9 days 04:27:33.310367
2	102	jane_smith	3 years 1 mon 8 days 04:27:33.310367
3	103	michael_jones	3 years 1 mon 7 days 04:27:33.310367
4	104	sarah_brown	3 years 1 mon 6 days 04:27:33.310367
5	105	david_miller	3 years 1 mon 5 days 04:27:33.310367
6	106	emily_davis	3 years 1 mon 4 days 04:27:33.310367
7	107	chris_wilson	3 years 1 mon 3 days 04:27:33.310367
8	108	anna_taylor	3 years 1 mon 2 days 04:27:33.310367
9	109	james_white	3 years 1 mon 1 day 04:27:33.310367
10	110	lisa_harris	3 years 1 mon 0 days 04:27:33.310367
11	111	kevin_clark	3 years 30 days 04:27:33.310367
12	112	laura_lee	3 years 29 days 04:27:33.310367
13	113	robert_hall	3 years 28 days 04:27:33.310367
14	114	sophia_wright	3 years 27 days 04:27:33.310367
15	115	daniel_thompson	3 years 26 days 04:27:33.310367
16	116	amelia_walker	3 years 25 days 04:27:33.310367
17	117	jack_evans	3 years 24 days 04:27:33.310367
18	118	olivia_moore	3 years 23 days 04:27:33.310367
19	119	ethan_hill	3 years 22 days 04:27:33.310367
20	120	isabella_adams	3 years 21 days 04:27:33.310367
21	121	alex_martin	3 years 20 days 04:27:33.310367
22	122	william_smith	3 years 19 days 04:27:33.310367

Total rows: 50    Query complete 00:00:00.062

Data Output Messages Notifications Query Query History

```
1 ✓ SELECT purchase_id, transaction_date, amount
2 FROM Purchase
3 WHERE transaction_date >= NOW() - INTERVAL '30 days';
4
```

	purchase_id [PK] integer	transaction_date date	amount numeric (10,2)
1	7	2025-02-03	94.64
2	13	2025-02-09	75.28
3	26	2025-01-22	39.22

Data Output Messages Notifications Query Query History

```
1 ✓ SELECT game_id, title, EXTRACT(YEAR FROM release_date) AS release_year
2   FROM Game;
3
```

	game_id [PK] integer	title character varying (255)	release_year numeric
1	1	Half-Life	1998
2	2	The Witcher 3: Wild Hunt	2015
3	3	Grand Theft Auto V	2013
4	4	Assassin's Creed Valhalla	2020
5	5	The Elder Scrolls V: Skyrim	2011
6	6	FIFA 23	2022
7	7	Final Fantasy XV	2016
8	8	Dark Souls III	2016
9	9	Overwatch	2016
10	10	Resident Evil 2	2019
11	11	Metal Gear Solid V	2015
12	12	Uncharted 4: A Thief's End	2016
13	13	Spider-Man	2018
14	14	Horizon Zero Dawn	2017
15	15	Sekiro: Shadows Die Twice	2019
16	16	Ghost of Tsushima	2020
17	17	Xenoblade Chronicles 2	2017
18	18	Call of Duty: Black Ops 4	2018
19	19	Titanfall 2	2016
20	20	Pillars of Eternity II	2018
21	21	Dragon Age: Inquisition	2014
22	22	The Walking Dead: Season One	2012

Total rows: 50    Query complete 00:00:00.103

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT review_id, review_date, TO_CHAR(review_date, 'Day') AS day_of_week
2   FROM Review;
3 |
```

	review_id [PK] integer	review_date date	day_of_week
1	2	2024-06-08	Saturday
2	3	2024-06-01	Saturday
3	4	2024-05-04	Saturday
4	5	2024-05-21	Tuesday
5	6	2024-04-02	Tuesday
6	7	2024-08-17	Saturday
7	8	2024-07-27	Saturday
8	9	2024-05-03	Friday
9	10	2024-11-27	Wednesday
10	11	2024-03-06	Wednesday
11	12	2024-07-06	Saturday
12	13	2024-07-18	Thursday
13	14	2024-03-04	Monday
14	15	2024-08-24	Saturday
15	16	2024-04-08	Monday
16	17	2024-07-02	Tuesday
17	18	2024-05-25	Saturday
18	19	2024-07-05	Friday
19	20	2024-12-06	Friday
20	21	2024-05-18	Saturday
21	22	2024-11-10	Sunday
22	23	2024-11-20	Wednesday

Total rows: 50

Query complete 00:00:00.062

Data Output	Messages	Notifications	<u>Query</u>	Query History
1	<code>SELECT game_id, title, release_date FROM Game WHERE EXTRACT(MONTH FROM release_date) = 11;</code>			
2				
3				
4				
	game_id [PK] integer	title character varying (255)	release_date date	
1	1	Half-Life	1998-11-19	
2	4	Assassin's Creed Valhalla	2020-11-10	
3	5	The Elder Scrolls V: Skyrim	2011-11-11	
4	7	Final Fantasy XV	2016-11-29	
5	21	Dragon Age: Inquisition	2014-11-18	
6	26	Dishonored 2	2016-11-11	
7	33	Death Stranding	2019-11-08	
8	46	Star Wars Jedi: Fallen Order	2019-11-15	
9	50	World of Warcraft	2004-11-23	

#### 4. Data Modification (5 UPDATE queries)

Changed Data Tracking System logs data about changes to user tables, tracking both the fact of DML changes and the actual data changed.  
**With this command we make changes to our database, track new data entered and collect statistics.**

Data Output	Messages	Notifications	<u>Query</u>	Query History
1	<code>UPDATE Game SET price = price * 0.9 WHERE release_date &lt; '2020-01-01';</code>			
2				
3				
4				

	game_id [PK] integer	title character varying (255)	genre character varying (100)	release_date date	price numeric (10,2)
1	4	Assassin's Creed Valhalla	Action	2020-11-10	69.99
2	6	FIFA 23	Sports	2022-09-27	59.99
3	16	Ghost of Tsushima	Action	2020-07-17	69.99
4	25	DOOM Eternal	FPS	2020-03-20	59.99
5	30	Hades	Rogue-like	2020-09-17	24.99
6	34	Hitman 3	Stealth	2021-01-20	69.99
7	40	Returnal	Rogue-like	2021-04-30	69.99
8	44	Cyberpunk 2077	RPG	2020-12-10	59.99
9	47	Horizon Forbidden West	RPG	2022-02-18	69.99
10	1	Half-Life	FPS	1998-11-19	17.99
11	2	The Witcher 3: Wild Hunt	RPG	2015-05-19	35.99
12	3	Grand Theft Auto V	Action	2013-09-17	53.99
13	5	The Elder Scrolls V: Skyrim	RPG	2011-11-11	26.99
14	7	Final Fantasy XV	RPG	2016-11-29	44.99
15	8	Dark Souls III	RPG	2016-03-24	35.99
16	9	Overwatch	FPS	2016-05-24	35.99
17	10	Resident Evil 2	Horror	2019-01-25	53.99
18	11	Metal Gear Solid V	Action	2015-09-01	26.99
19	12	Uncharted 4: A Thief's End	Adventure	2016-05-10	26.99
20	13	Spider-Man	Action	2018-09-07	44.99
21	14	Horizon Zero Dawn	RPG	2017-02-28	35.99

Data Output   Messages   Notifications   Query   Query History

```

1 ✓ UPDATE useraccount
2   SET status = 'Premium'
3 WHERE user_id BETWEEN 101 AND 120;
4

```

	user_id [PK] integer	username character varying (100)	email character varying (255)	registration_date date	country character varying (100)	age integer	status character varying (20)
30	150	bella_reed	bella.reed@example.com	2022-02-19	India	[null]	[null]
31	101	john_doe	john.doe@example.com	2022-01-01	USA	[null]	Premium
32	102	jane_smith	jane.smith@example.com	2022-01-02	Canada	[null]	Premium
33	103	michael_jones	michael.jones@example.com	2022-01-03	UK	[null]	Premium
34	104	sarah_brown	sarah.brown@example.com	2022-01-04	Germany	[null]	Premium
35	105	david_miller	david.miller@example.com	2022-01-05	France	[null]	Premium
36	106	emily_davis	emily.davis@example.com	2022-01-06	Japan	[null]	Premium
37	107	chris_wilson	chris.wilson@example.com	2022-01-07	Australia	[null]	Premium
38	108	anna_taylor	anna.taylor@example.com	2022-01-08	Russia	[null]	Premium
39	109	james_white	james.white@example.com	2022-01-09	Brazil	[null]	Premium
40	110	lisa_harris	lisa.harris@example.com	2022-01-10	India	[null]	Premium
41	111	kevin_clark	kevin.clark@example.com	2022-01-11	China	[null]	Premium
42	112	laura_lee	laura.lee@example.com	2022-01-12	Italy	[null]	Premium
43	113	robert_hall	robert.hall@example.com	2022-01-13	Mexico	[null]	Premium
44	114	sophia_wright	sophia.wright@example.com	2022-01-14	Spain	[null]	Premium
45	115	daniel_thompson	daniel.thompson@example.com	2022-01-15	Netherlands	[null]	Premium
46	116	amelia_walker	amelia.walker@example.com	2022-01-16	Sweden	[null]	Premium
47	117	jack_evans	jack.evans@example.com	2022-01-17	Norway	[null]	Premium
48	118	olivia_moore	olivia.moore@example.com	2022-01-18	Denmark	[null]	Premium
49	119	ethan_hill	ethan.hill@example.com	2022-01-19	Finland	[null]	Premium
50	120	isabella_adams	isabella.adams@example.com	2022-01-20	Poland	[null]	Premium

Data Output Messages Notifications Query Query History

```
1 ✓ UPDATE System_Requirements
2   SET ram = ram + 4
3 WHERE game_id = 3;
4 |
```

	requirement_id [PK] integer	game_id integer	os character varying (255)	processor character varying (255)	ram integer	gpu chara
1	1	1	Windows 10	Intel Core i5	12	NVID
2	2	2	Windows 10	Intel Core i7	20	NVID
3	4	4	Windows 10	Intel Core i9	36	NVID
4	5	5	Windows 8	Intel Core i3	8	NVID
5	6	6	Windows 10	AMD Ryzen 7	20	AMD
6	7	7	Windows 7	Intel Core i5	12	NVID
7	8	8	Windows 10	Intel Core i7	20	NVID
8	9	9	Windows 10	AMD Ryzen 5	12	AMD
9	10	10	Windows 8	Intel Core i5	12	NVID
10	11	11	Windows 10	Intel Core i7	20	NVID
11	12	12	Windows 10	AMD Ryzen 3	8	AMD
12	13	13	Windows 10	Intel Core i9	36	NVID
13	14	14	Windows 7	Intel Core i5	12	NVID
14	15	15	Windows 10	AMD Ryzen 7	20	AMD
15	16	16	Windows 10	Intel Core i3	8	NVID
16	17	17	Windows 8	Intel Core i5	12	NVID
17	18	18	Windows 10	AMD Ryzen 5	12	AMD
18	19	19	Windows 7	Intel Core i7	20	NVID
19	20	20	Windows 10	Intel Core i9	36	NVID
20	21	21	Windows 10	Intel Core i5	12	NVID
21	22	22	Windows 10	AMD Ryzen 7	20	AMD
22	22	22	Windows 10	Intel Core i7	20	NVID

Total rows: 50    Query complete 00:00:00.064

Data Output Messages Notifications Query Query History

```
1 ✓ UPDATE Review
2 SET helpful_votes = helpful_votes + 10
3 WHERE rating >= 4.5;
4
```

	review_id [PK] integer	user_id integer	game_id integer	rating integer	comment text	review_date date	helpful_votes integer
1	2	101	1	3	Amazing gameplay and graphics!	2024-06-08	37
2	3	101	11	3	Not recommended. Needs improvements.	2024-06-01	29
3	4	101	21	3	Amazing gameplay and graphics!	2024-05-04	42
4	5	101	31	1	Not recommended. Needs improvements.	2024-05-21	1
5	8	102	12	2	Good, but with some bugs.	2024-07-27	26
6	9	102	22	2	Good, but with some bugs.	2024-05-03	31
7	13	103	13	4	Good, but with some bugs.	2024-07-18	1
8	15	103	33	4	Amazing gameplay and graphics!	2024-08-24	18
9	16	103	43	1	Not recommended. Needs improvements.	2024-04-08	28
10	17	104	4	4	Good, but with some bugs.	2024-07-02	15
11	18	104	14	3	Good, but with some bugs.	2024-05-25	9
12	20	104	34	1	Amazing gameplay and graphics!	2024-12-06	24
13	21	104	44	2	Good, but with some bugs.	2024-05-18	45
14	22	105	5	4	Not recommended. Needs improvements.	2024-11-10	13
15	23	105	15	1	Not recommended. Needs improvements.	2024-11-28	35
16	24	105	25	2	Good, but with some bugs.	2024-04-12	15
17	25	105	35	2	Amazing gameplay and graphics!	2025-01-11	17
18	26	105	45	4	Not recommended. Needs improvements.	2024-03-10	26
19	27	106	6	4	Good, but with some bugs.	2024-05-25	15
20	30	106	36	3	Amazing gameplay and graphics!	2024-10-28	30
21	31	106	46	3	Amazing gameplay and graphics!	2024-12-11	17
22	22	107	7	4	Good, but with some bugs.	2025-01-22	10

Total rows: 50

Query complete 00:00:00.046

```

1 ✓ UPDATE Purchase
2 SET payment_method = 'Credit Card'
3 WHERE payment_method = 'Cash';
4

```

	<code>purchase_id</code> [PK] integer ↗	<code>user_id</code> integer ↗	<code>game_id</code> integer ↗	<code>transaction_date</code> date ↗	<code>amount</code> numeric (10,2) ↗	<code>payment_method</code> character varying (50) ↗	<code>discount_applied</code> boolean ↗
1	1	141	18	2024-10-11	72.22	PayPal	false
2	2	141	18	2024-08-31	31.93	Credit Card	false
3	3	141	18	2025-01-10	58.58	PayPal	false
4	4	141	18	2024-03-23	98.09	Credit Card	true
5	5	141	18	2024-02-26	60.91	Credit Card	false
6	6	141	18	2024-07-28	65.80	Credit Card	true
7	7	141	18	2025-02-03	94.64	PayPal	false
8	8	141	18	2024-07-21	96.14	Credit Card	false
9	9	141	18	2024-03-21	80.73	PayPal	false
10	10	141	18	2024-03-13	44.47	PayPal	true
11	11	141	18	2024-07-16	89.67	Credit Card	true
12	12	141	18	2024-11-04	46.55	Credit Card	false
13	13	141	18	2025-02-09	75.28	Credit Card	false
14	14	141	18	2024-03-03	29.46	PayPal	false
15	15	141	18	2024-07-04	81.68	Credit Card	true
16	16	141	18	2024-04-28	49.18	PayPal	false
17	17	141	18	2024-02-26	80.28	Credit Card	false
18	18	141	18	2024-07-29	99.66	Credit Card	true
19	19	141	18	2024-03-18	52.14	PayPal	false
20	20	141	18	2024-10-24	21.53	Credit Card	false
21	21	141	18	2024-12-18	108.36	Credit Card	true
22	22	141	18	2024-02-21	78.20	PayPal	false

Total rows: 50

Query complete 00:00:00.045

## 5. Data Deletion (5 DELETE queries)

The Delete command deletes the row, table or table we need. In this task, we set various conditions with the help of which we deleted this or that information.

	review_id [PK] integer	user_id integer	game_id integer	rating integer	comment text	review_date date	helpful_votes integer
1	2	101	1	3	Amazing gameplay and graphics!	2024-06-08	37
2	3	101	11	3	Not recommended. Needs improvements.	2024-06-01	29
3	4	101	21	3	Amazing gameplay and graphics!	2024-05-04	42
4	8	102	12	2	Good, but with some bugs.	2024-07-27	26
5	9	102	22	2	Good, but with some bugs.	2024-05-03	31
6	13	103	13	4	Good, but with some bugs.	2024-07-18	1
7	15	103	33	4	Amazing gameplay and graphics!	2024-08-24	18
8	17	104	4	4	Good, but with some bugs.	2024-07-02	15
9	18	104	14	3	Good, but with some bugs.	2024-05-25	9
10	21	104	44	2	Good, but with some bugs.	2024-05-18	45
11	22	105	5	4	Not recommended. Needs improvements.	2024-11-10	13
12	24	105	25	2	Good, but with some bugs.	2024-04-12	15
13	25	105	35	2	Amazing gameplay and graphics!	2025-01-11	17
14	26	105	45	4	Not recommended. Needs improvements.	2024-03-10	26
15	27	106	6	4	Good, but with some bugs.	2024-05-25	15
16	30	106	36	3	Amazing gameplay and graphics!	2024-10-28	30
17	31	106	46	3	Amazing gameplay and graphics!	2024-12-11	17
18	32	107	7	4	Good, but with some bugs.	2025-01-22	12
19	34	107	27	2	Good, but with some bugs.	2025-01-26	27
20	35	107	37	4	Amazing gameplay and graphics!	2024-10-07	11
21	37	108	8	2	Good, but with some bugs.	2024-11-30	27
22	40	108	20	2	Good, but with some bugs.	2024-02-11	17

Total rows: 44    Query complete 00:00:00.073

	review_id [PK] integer	user_id integer	game_id integer	rating integer	comment text	review_date date	helpful_votes integer
1	8	102	12	2	Good, but with some bugs.	2024-07-27	26
2	102	jane_smith	jane.smith@example.com	2022-01-02	Canada	[null]	Premium

10 | 2 | The Witcher 3: Wild Hunt | RPG | 2015-05-19 | 35.99 | 2 | [null] | PC, PS4, Xbox One | [null]

Data Output Messages Notifications Query Query History

```
1 ✓ DELETE FROM System_Requirements  
2 WHERE game_id NOT IN (SELECT game_id FROM Game);  
3
```

	requirement_id [PK] integer	game_id integer	os character varying (255)	processor character varying (255)	ram integer	gpu character varying (255)	storage character varying (50)	directx_version character varying (20)
1	2	2	Windows 10	Intel Core i7	20	NVIDIA GTX 1080	100 GB	DirectX 12
2	4	4	Windows 10	Intel Core i9	36	NVIDIA RTX 3070	150 GB	DirectX 12
3	5	5	Windows 8	Intel Core i3	8	NVIDIA GTX 950	30 GB	DirectX 11
4	6	6	Windows 10	AMD Ryzen 7	20	AMD RX 5700 XT	120 GB	DirectX 12
5	7	7	Windows 7	Intel Core i5	12	NVIDIA GTX 970	60 GB	DirectX 11
6	8	8	Windows 10	Intel Core i7	20	NVIDIA GTX 1660	90 GB	DirectX 12
7	9	9	Windows 10	AMD Ryzen 5	12	AMD RX 560	50 GB	DirectX 11
8	10	10	Windows 8	Intel Core i5	12	NVIDIA GTX 1050 Ti	40 GB	DirectX 11

Data Output Messages Notifications Query Query History

```
1 ✓ DELETE FROM Purchase  
2 WHERE transaction_date < NOW() - INTERVAL '2 years';  
3
```

	purchase_id [PK] integer	user_id integer	game_id integer	transaction_date date	amount numeric (10,2)	payment_method character varying (50)	discount_applied boolean
1	1	141	18	2024-10-11	72.22	PayPal	false
2	2	141	18	2024-08-31	31.93	Credit Card	false
3	3	141	18	2025-01-10	58.58	PayPal	false
4	4	141	18	2024-03-23	98.09	Credit Card	true
5	5	141	18	2024-02-26	60.91	Credit Card	false
6	6	141	18	2024-07-28	65.80	Credit Card	true
7	7	141	18	2025-02-03	94.64	PayPal	false
8	8	141	18	2024-07-21	96.14	Credit Card	false
9	9	141	18	2024-03-21	80.73	PayPal	false
10	10	141	18	2024-03-13	44.47	PayPal	true
11	11	141	18	2024-07-16	89.67	Credit Card	true
12	12	141	18	2024-11-04	46.55	Credit Card	false
13	13	141	18	2025-02-09	75.28	Credit Card	false
14	14	141	18	2024-03-03	29.46	PayPal	false
15	15	141	18	2024-07-04	81.68	Credit Card	true
16	16	141	18	2024-04-28	49.18	PayPal	false
17	17	141	18	2024-02-26	80.28	Credit Card	false
18	18	141	18	2024-07-29	99.66	Credit Card	true
19	19	141	18	2024-03-18	52.14	PayPal	false
20	20	141	18	2024-10-24	21.53	Credit Card	false
21	21	141	18	2024-12-18	108.36	Credit Card	true
22	22	141	18	2024-03-21	70.00	PayPal	false

## 4. SQL JOINS & Data Retrieval (Based on Assignment 4 )

### 1. INNER JOIN (10 pts)

The INNER JOIN operation can be used in any FROM clause. This is the most common type of join. With it, we combined information from two different columns.

	Data Output	Messages	Notifications	Query	Query History
1	▼	<pre>SELECT u.username, g.title, r.rating, r.comment</pre>			
2		<pre>FROM review r</pre>			
3		<pre>INNER JOIN useraccount u ON r.user_id = u.user_id</pre>			
4		<pre>INNER JOIN game g ON r.game_id = g.game_id;</pre>			
5					
	<b>username</b> character varying (100)	<b>title</b> character varying (255)	<b>rating</b> integer	<b>comment</b> text	
1	jane_smith	Uncharted 4: A Thief's End	2	Good, but with some bugs.	
2	jane_smith	The Walking Dead: Season One	2	Good, but with some bugs.	
3	michael_jones	Spider-Man	4	Good, but with some bugs.	
4	michael_jones	Death Stranding	4	Amazing gameplay and graphics!	
5	sarah_brown	Assassin's Creed Valhalla	4	Good, but with some bugs.	
6	sarah_brown	Horizon Zero Dawn	3	Good, but with some bugs.	
7	sarah_brown	Cyberpunk 2077	2	Good, but with some bugs.	
8	david_miller	The Elder Scrolls V: Skyrim	4	Not recommended. Needs improvements.	
9	david_miller	DOOM Eternal	2	Good, but with some bugs.	
10	david_miller	Stellaris	2	Amazing gameplay and graphics!	
11	david_miller	The Sims 4	4	Not recommended. Needs improvements.	
12	emily_davis	FIFA 23	4	Good, but with some bugs.	
13	emily_davis	League of Legends	3	Amazing gameplay and graphics!	
14	emily_davis	Star Wars Jedi: Fallen Order	3	Amazing gameplay and graphics!	
15	chris_wilson	Final Fantasy XV	4	Good, but with some bugs.	
16	chris_wilson	Divinity: Original Sin II	2	Good, but with some bugs.	
17	chris_wilson	Fortnite	4	Amazing gameplay and graphics!	
18	anna_taylor	Dark Souls III	2	Good, but with some bugs.	
19	anna_taylor	STALKER: Shadow of Chernobyl	3	Good, but with some bugs.	
20	anna_taylor	Monster Hunter: World	2	Good, but with some bugs.	
21	james_white	Titanfall 2	4	Not recommended. Needs improvements.	
22	...	...	...	...	...
Total rows: 40		Query complete 00:00:00.049			

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT g.title, p.name AS publisher_name
2   FROM game g
3   INNER JOIN publisher p ON g.publisher_id = p.publisher_id;
4
```

	title character varying (255)	publisher_name character varying (255)
1	Assassin's Creed Valhalla	Epic Games
2	FIFA 23	Epic Games
3	Ghost of Tsushima	Epic Games
4	DOOM Eternal	Epic Games
5	Hades	Epic Games
6	Hitman 3	Epic Games
7	Returnal	Epic Games
8	Cyberpunk 2077	Epic Games
9	Horizon Forbidden West	Epic Games
10	The Witcher 3: Wild Hunt	Epic Games
11	Grand Theft Auto V	Epic Games
12	The Elder Scrolls V: Skyrim	Epic Games
13	Final Fantasy XV	Epic Games
14	Dark Souls III	Epic Games
15	Overwatch	Epic Games
16	Resident Evil 2	Epic Games
17	Metal Gear Solid V	Epic Games
18	Uncharted 4: A Thief's End	Epic Games
19	Spider-Man	Epic Games
20	Horizon Zero Dawn	Epic Games
21	Sekiro: Shadows Die Twice	Epic Games
22	Yakuza: Like a Dragon	Epic Games

Total rows: 49    Query complete 00:00:00.038

## 2. FULL JOIN (15 pts)

FULL JOIN allows you to see all rows from tables, even if they do not match the specified field or condition. Using this operator, we searched for all rows that did not match each other.

Data Output Messages Notifications Query Query History

```
1 ✓ SELECT g.title, s.ram, s.processor
2   FROM game g
3   FULL JOIN system_requirements s ON g.game_id = s.game_id;
4
```

	title character varying (255)	ram integer	processor character varying (255)
1	The Witcher 3: Wild Hunt	20	Intel Core i7
2	Assassin's Creed Valhalla	36	Intel Core i9
3	The Elder Scrolls V: Skyrim	8	Intel Core i3
4	FIFA 23	20	AMD Ryzen 7
5	Final Fantasy XV	12	Intel Core i5
6	Dark Souls III	20	Intel Core i7
7	Overwatch	12	AMD Ryzen 5
8	Resident Evil 2	12	Intel Core i5
9	Metal Gear Solid V	20	Intel Core i7
10	Uncharted 4: A Thief's End	8	AMD Ryzen 3
11	Spider-Man	36	Intel Core i9
12	Horizon Zero Dawn	12	Intel Core i5
13	Sekiro: Shadows Die Twice	20	AMD Ryzen 7
14	Ghost of Tsushima	8	Intel Core i3
15	Xenoblade Chronicles 2	12	Intel Core i5
16	Call of Duty: Black Ops 4	12	AMD Ryzen 5
17	Titanfall 2	20	Intel Core i7
18	Pillars of Eternity II	36	Intel Core i9
19	Dragon Age: Inquisition	12	Intel Core i5
20	The Walking Dead: Season One	20	AMD Ryzen 7
21	Control	20	Intel Core i7
22	Death Stranding	12	AMD Ryzen 5

Total rows: 49    Query complete 00:00:00.046

Data Output Messages Notifications Query Query History

```
1 ▾ SELECT u.username, r.comment, r.rating  
2   FROM useraccount u  
3   FULL JOIN review r ON u.user_id = r.user_id;  
4 |
```

	username character varying (100) 	comment text 	rating integer 
1	jane_smith	Good, but with some bugs.	2
2	jane_smith	Good, but with some bugs.	2
3	michael_jones	Good, but with some bugs.	4
4	michael_jones	Amazing gameplay and graphics!	4
5	sarah_brown	Good, but with some bugs.	4
6	sarah_brown	Good, but with some bugs.	3
7	sarah_brown	Good, but with some bugs.	2
8	david_miller	Not recommended. Needs improvements.	4
9	david_miller	Good, but with some bugs.	2
10	david_miller	Amazing gameplay and graphics!	2
11	david_miller	Not recommended. Needs improvements.	4
12	emily_davis	Good, but with some bugs.	4
13	emily_davis	Amazing gameplay and graphics!	3
14	emily_davis	Amazing gameplay and graphics!	3
15	chris_wilson	Good, but with some bugs.	4
16	chris_wilson	Good, but with some bugs.	2
17	chris_wilson	Amazing gameplay and graphics!	4
18	anna_taylor	Good, but with some bugs.	2
19	anna_taylor	Good, but with some bugs.	3
20	anna_taylor	Good, but with some bugs.	2
21	james_white	Not recommended. Needs improvements.	4
22	.....	.....	2

Total rows: 80    Query complete 00:00:00.046

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT p.purchase_id, g.title, p.amount  
2 FROM purchase p  
3 FULL JOIN game g ON p.game_id = g.game_id;  
4
```

	purchase_id integer	title character varying (255)	amount numeric (10,2)
1	1	Call of Duty: Black Ops 4	72.22
2	2	Call of Duty: Black Ops 4	31.93
3	3	Call of Duty: Black Ops 4	58.58
4	4	Call of Duty: Black Ops 4	98.09
5	5	Call of Duty: Black Ops 4	60.91
6	6	Call of Duty: Black Ops 4	65.80
7	7	Call of Duty: Black Ops 4	94.64
8	8	Call of Duty: Black Ops 4	96.14
9	9	Call of Duty: Black Ops 4	80.73
10	10	Call of Duty: Black Ops 4	44.47
11	11	Call of Duty: Black Ops 4	89.67
12	12	Call of Duty: Black Ops 4	46.55
13	13	Call of Duty: Black Ops 4	75.28
14	14	Call of Duty: Black Ops 4	29.46
15	15	Call of Duty: Black Ops 4	81.68
16	16	Call of Duty: Black Ops 4	49.18
17	17	Call of Duty: Black Ops 4	80.28
18	18	Call of Duty: Black Ops 4	99.66
19	19	Call of Duty: Black Ops 4	52.14
20	20	Call of Duty: Black Ops 4	21.53
21	21	Call of Duty: Black Ops 4	108.36
22	22	Call of Duty: Black Ops 4	70.00

Total rows: 98    Query complete 00:00:00.053

### 3. LEFT JOIN (15 pts)

The LEFT JOIN operation creates a left outer join. A left outer join selects all records from the first (left) table, even if they do not match records in the second (right) table.

In this task, we created a left outer join

Data Output Messages Notifications Query <u>Query</u> Query History		
1 ✓ <code>SELECT g.title, p.name AS publisher_name</code> 2 <code>FROM game g</code> 3 <code>LEFT JOIN publisher p ON g.publisher_id = p.publisher_id;</code> 4		
	<code>title</code> character varying (255)	<code>publisher_name</code> character varying (255)
1	Assassin's Creed Valhalla	Epic Games
2	FIFA 23	Epic Games
3	Ghost of Tsushima	Epic Games
4	DOOM Eternal	Epic Games
5	Hades	Epic Games
6	Hitman 3	Epic Games
7	Returnal	Epic Games
8	Cyberpunk 2077	Epic Games
9	Horizon Forbidden West	Epic Games
10	The Witcher 3: Wild Hunt	Epic Games
11	Grand Theft Auto V	Epic Games
12	The Elder Scrolls V: Skyrim	Epic Games
13	Final Fantasy XV	Epic Games
14	Dark Souls III	Epic Games
15	Overwatch	Epic Games
16	Resident Evil 2	Epic Games
17	Metal Gear Solid V	Epic Games
18	Uncharted 4: A Thief's End	Epic Games
19	Spider-Man	Epic Games
20	Horizon Zero Dawn	Epic Games
21	Sekiro: Shadows Die Twice	Epic Games
22	Yakuza: Like a Dragon	Epic Games
Total rows: 49		Query complete 00:00:00.046

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT u.username, p.purchase_id, p.amount  
2 FROM useraccount u  
3 LEFT JOIN purchase p ON u.user_id = p.user_id;
```

	username character varying (100) 	purchase_id integer 	amount numeric (10,2) 
1	isaac_bailey	1	72.22
2	isaac_bailey	2	31.93
3	isaac_bailey	3	58.58
4	isaac_bailey	4	98.09
5	isaac_bailey	5	60.91
6	isaac_bailey	6	65.80
7	isaac_bailey	7	94.64
8	isaac_bailey	8	96.14
9	isaac_bailey	9	80.73
10	isaac_bailey	10	44.47
11	isaac_bailey	11	89.67
12	isaac_bailey	12	46.55
13	isaac_bailey	13	75.28
14	isaac_bailey	14	29.46
15	isaac_bailey	15	81.68
16	isaac_bailey	16	49.18
17	isaac_bailey	17	80.28
18	isaac_bailey	18	99.66
19	isaac_bailey	19	52.14
20	isaac_bailey	20	21.53
21	isaac_bailey	21	108.36
22	.....	22	70.00

Total rows: 98 | Query complete 00:00:00.060

Data Output Messages Notifications Query Query History

```
1 ✓ SELECT g.title, r.comment, r.rating  
2   FROM game g  
3   LEFT JOIN review r ON g.game_id = r.game_id;  
4
```

	title character varying (255)	comment text	rating integer
1	Uncharted 4: A Thief's End	Good, but with some bugs.	2
2	The Walking Dead: Season One	Good, but with some bugs.	2
3	Spider-Man	Good, but with some bugs.	4
4	Death Stranding	Amazing gameplay and graphics!	4
5	Assassin's Creed Valhalla	Good, but with some bugs.	4
6	Horizon Zero Dawn	Good, but with some bugs.	3
7	Cyberpunk 2077	Good, but with some bugs.	2
8	The Elder Scrolls V: Skyrim	Not recommended. Needs improvements.	4
9	DOOM Eternal	Good, but with some bugs.	2
10	Stellaris	Amazing gameplay and graphics!	2
11	The Sims 4	Not recommended. Needs improvements.	4
12	FIFA 23	Good, but with some bugs.	4
13	League of Legends	Amazing gameplay and graphics!	3
14	Star Wars Jedi: Fallen Order	Amazing gameplay and graphics!	3
15	Final Fantasy XV	Good, but with some bugs.	4
16	Divinity: Original Sin II	Good, but with some bugs.	2
17	Fortnite	Amazing gameplay and graphics!	4
18	Dark Souls III	Good, but with some bugs.	2
19	STALKER: Shadow of Chernobyl	Good, but with some bugs.	3
20	Monster Hunter: World	Good, but with some bugs.	2
21	Titanfall 2	Not recommended. Needs improvements.	4
22	Grand Theft Auto V	Good, but with some bugs.	2

Total rows: 49    Query complete 00:00:00.076

#### 4. RIGHT JOIN (15 pts)

The RIGHT JOIN operation creates a right outer join. A right outer join selects all records from the second (right) table, even if they do not match records in the first (left) table.

Data Output		Messages	Notifications	Query	Query History
1	▼	<pre>1 ✓ SELECT p.name AS publisher_name, g.title 2   FROM game g 3   RIGHT JOIN publisher p ON g.publisher_id = p.publisher_id;</pre>			
1		<b>publisher_name</b> character varying (255)	🔒	<b>title</b> character varying (255)	🔒
1		Epic Games		Assassin's Creed Valhalla	
2		Epic Games		FIFA 23	
3		Epic Games		Ghost of Tsushima	
4		Epic Games		DOOM Eternal	
5		Epic Games		Hades	
6		Epic Games		Hitman 3	
7		Epic Games		Returnal	
8		Epic Games		Cyberpunk 2077	
9		Epic Games		Horizon Forbidden West	
10		Epic Games		The Witcher 3: Wild Hunt	
11		Epic Games		Grand Theft Auto V	
12		Epic Games		The Elder Scrolls V: Skyrim	
13		Epic Games		Final Fantasy XV	
14		Epic Games		Dark Souls III	
15		Epic Games		Overwatch	
16		Epic Games		Resident Evil 2	
17		Epic Games		Metal Gear Solid V	
18		Epic Games		Uncharted 4: A Thief's End	
19		Epic Games		Spider-Man	
20		Epic Games		Horizon Zero Dawn	
21		Epic Games		Sekiro: Shadows Die Twice	
22		Epic Games		Yakuza: Like a Dragon	
Total rows: 98		Query complete 00:00:00.047			

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT p.purchase_id, u.username, p.amount  
2 FROM purchase p  
3 RIGHT JOIN useraccount u ON p.user_id = u.user_id;  
4 |
```

	<b>purchase_id</b> integer	<b>username</b> character varying (100)	<b>amount</b> numeric (10,2)
1	1	isaac_bailey	72.22
2	2	isaac_bailey	31.93
3	3	isaac_bailey	58.58
4	4	isaac_bailey	98.09
5	5	isaac_bailey	60.91
6	6	isaac_bailey	65.80
7	7	isaac_bailey	94.64
8	8	isaac_bailey	96.14
9	9	isaac_bailey	80.73
10	10	isaac_bailey	44.47
11	11	isaac_bailey	89.67
12	12	isaac_bailey	46.55
13	13	isaac_bailey	75.28
14	14	isaac_bailey	29.46
15	15	isaac_bailey	81.68
16	16	isaac_bailey	49.18
17	17	isaac_bailey	80.28
18	18	isaac_bailey	99.66
19	19	isaac_bailey	52.14
20	20	isaac_bailey	21.53
21	21	isaac_bailey	108.36
22	22	.....	70.20

Total rows: 98

Query complete 00:00:00.061

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT s.processor, s.ram, g.title
2 FROM system_requirements s
3 RIGHT JOIN game g ON s.game_id = g.game_id;
4
```

	processor character varying (255) 	ram integer 	title character varying (255) 
1	Intel Core i7	20	The Witcher 3: Wild Hunt
2	Intel Core i9	36	Assassin's Creed Valhalla
3	Intel Core i3	8	The Elder Scrolls V: Skyrim
4	AMD Ryzen 7	20	FIFA 23
5	Intel Core i5	12	Final Fantasy XV
6	Intel Core i7	20	Dark Souls III
7	AMD Ryzen 5	12	Overwatch
8	Intel Core i5	12	Resident Evil 2
9	Intel Core i7	20	Metal Gear Solid V
10	AMD Ryzen 3	8	Uncharted 4: A Thief's End
11	Intel Core i9	36	Spider-Man
12	Intel Core i5	12	Horizon Zero Dawn
13	AMD Ryzen 7	20	Sekiro: Shadows Die Twice
14	Intel Core i3	8	Ghost of Tsushima
15	Intel Core i5	12	Xenoblade Chronicles 2
16	AMD Ryzen 5	12	Call of Duty: Black Ops 4
17	Intel Core i7	20	Titanfall 2
18	Intel Core i9	36	Pillars of Eternity II
19	Intel Core i5	12	Dragon Age: Inquisition
20	AMD Ryzen 7	20	The Walking Dead: Season One
21	Intel Core i7	20	Control
22	AMD Ryzen 5	12	Death Stranding

Total rows: 49

Query complete 00:00:00.050

## 5. CROSS JOIN (15 pts)

Each row of one table is joined with each row of the second table, thereby yielding all possible combinations of rows of the two tables.

	Data	Output	Messages	Notifications	Query	Query History
1	▼	<pre>1 ✓ SELECT u.username, g.title 2   FROM useraccount u 3   CROSS JOIN game g;</pre>				
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						
21						
22						
Total rows: 2401		Query complete 00:00:00.054				

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT g.title, p.name AS publisher_name
2   FROM game g
3   CROSS JOIN publisher p;
4
```

	title character varying (255) 	publisher_name character varying (255) 
1	Assassin's Creed Valhalla	Valve
2	Assassin's Creed Valhalla	CD Projekt
3	Assassin's Creed Valhalla	Rockstar Games
4	Assassin's Creed Valhalla	Ubisoft
5	Assassin's Creed Valhalla	Bethesda Softworks
6	Assassin's Creed Valhalla	EA Sports
7	Assassin's Creed Valhalla	Square Enix
8	Assassin's Creed Valhalla	Capcom
9	Assassin's Creed Valhalla	Blizzard Entertainment
10	Assassin's Creed Valhalla	Konami
11	Assassin's Creed Valhalla	Bandai Namco
12	Assassin's Creed Valhalla	Sony Interactive Entertainment
13	Assassin's Creed Valhalla	Microsoft Studios
14	Assassin's Creed Valhalla	Nintendo
15	Assassin's Creed Valhalla	Sega
16	Assassin's Creed Valhalla	Warner Bros. Interactive
17	Assassin's Creed Valhalla	Deep Silver
18	Assassin's Creed Valhalla	Activision
19	Assassin's Creed Valhalla	THQ Nordic
20	Assassin's Creed Valhalla	Focus Home Interactive
21	Assassin's Creed Valhalla	Obsidian Entertainment
22	Assassin's Creed Valhalla	Paradox Interactive

Total rows: 2450 | Query complete 00:00:00.055

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT p.purchase_id, r.comment  
2 FROM purchase p  
3 CROSS JOIN review r;  
4 |
```

	purchase_id integer 	comment text 
1	1	Good, but with some bugs.
2	1	Good, but with some bugs.
3	1	Good, but with some bugs.
4	1	Amazing gameplay and graphics!
5	1	Good, but with some bugs.
6	1	Good, but with some bugs.
7	1	Good, but with some bugs.
8	1	Not recommended. Needs improvements.
9	1	Good, but with some bugs.
10	1	Amazing gameplay and graphics!
11	1	Not recommended. Needs improvements.
12	1	Good, but with some bugs.
13	1	Amazing gameplay and graphics!
14	1	Amazing gameplay and graphics!
15	1	Good, but with some bugs.
16	1	Good, but with some bugs.
17	1	Amazing gameplay and graphics!
18	1	Good, but with some bugs.
19	1	Good, but with some bugs.
20	1	Good, but with some bugs.
21	1	Not recommended. Needs improvements.
22	1	Good, but with some bugs.

Total rows: 2000

Query complete 00:00:00.072

## 6. NATURAL JOIN (15 pts)

With this operator we compared the data in all columns, and it calculated which of them contained similar information.

```
Data Output Messages Notifications Query Query History
1 ✓ SELECT *
2   FROM game NATURAL JOIN publisher;
3 |
```

	publisher_id	game_id	title	genre	release_date	price	developer_id	platform
	integer	integer	character varying (255)	character varying (100)	date	numeric (10,2)	integer	character varying (100)
1	38	4	Assassin's Creed Valhalla	Action	2020-11-10	69.99	4	PC, PS5, Xbox
2	38	6	FIFA 23	Sports	2022-09-27	59.99	6	PC, PS5, Xbox
3	38	16	Ghost of Tsushima	Action	2020-07-17	69.99	16	PS4, PS5
4	38	25	DOOM Eternal	FPS	2020-03-20	59.99	27	PC, PS4, Xbox
5	38	30	Hades	Rogue-like	2020-09-17	24.99	32	PC, Nintendo Switch
6	38	34	Hitman 3	Stealth	2021-01-20	69.99	37	PC, PS5, Xbox
7	38	40	Returnal	Rogue-like	2021-04-30	69.99	50	PC, PS5
8	38	44	Cyberpunk 2077	RPG	2020-12-10	59.99	2	PC, PS5, Xbox
9	38	47	Horizon Forbidden West	RPG	2022-02-18	69.99	14	PC, PS5
10	38	2	The Witcher 3: Wild Hunt	RPG	2015-05-19	35.99	2	PC, PS4, Xbox One
11	38	3	Grand Theft Auto V	Action	2013-09-17	53.99	3	PC, PS4, PS5, Xbox
12	38	5	The Elder Scrolls V: Skyrim	RPG	2011-11-11	26.99	5	PC, PS4, Xbox
13	38	7	Final Fantasy XV	RPG	2016-11-29	44.99	7	PC, PS4, Xbox
14	38	8	Dark Souls III	RPG	2016-03-24	35.99	15	PC, PS4, Xbox
15	38	9	Overwatch	FPS	2016-05-24	35.99	9	PC, PS4, Xbox
16	38	10	Resident Evil 2	Horror	2019-01-25	53.99	10	PC, PS4, Xbox
17	38	11	Metal Gear Solid V	Action	2015-09-01	26.99	36	PC, PS4, Xbox
18	38	12	Uncharted 4: A Thief's End	Adventure	2016-05-10	26.99	12	PS4
19	38	13	Spider-Man	Action	2018-09-07	44.99	13	PS4, PS5
20	38	14	Horizon Zero Dawn	RPG	2017-02-28	35.99	14	PC, PS4
21	38	15	Sekiro: Shadows Die Twice	Action	2019-03-22	53.99	15	PC, PS4, Xbox

Total rows: 49    Query complete 00:00:00.049    CRLF

Data Output Messages Notifications **Query** Query History

```
1 ▼ SELECT *
2   FROM review NATURAL JOIN useraccount;
3
```

	user_id	review_id	game_id	rating	comment	review_date	helpful_votes	username	email	regis
	integer	integer	integer	integer	text	date	integer	character varying (100)	character varying (255)	date
1	102	8	12	2	Good, but with some bugs.	2024-07-27	26	jane_smith	jane.smith@example.com	202
2	102	9	22	2	Good, but with some bugs.	2024-05-03	31	jane_smith	jane.smith@example.com	202
3	103	13	13	4	Good, but with some bugs.	2024-07-18	1	michael_jones	michael.jones@example.com	202
4	103	15	33	4	Amazing gameplay and graphics!	2024-08-24	18	michael_jones	michael.jones@example.com	202
5	104	17	4	4	Good, but with some bugs.	2024-07-02	15	sarah_brown	sarah.brown@example.com	202
6	104	18	14	3	Good, but with some bugs.	2024-05-25	9	sarah_brown	sarah.brown@example.com	202
7	104	21	44	2	Good, but with some bugs.	2024-05-18	45	sarah_brown	sarah.brown@example.com	202
8	105	22	5	4	Not recommended. Needs improvements.	2024-11-10	13	david_miller	david.miller@example.com	202
9	105	24	25	2	Good, but with some bugs.	2024-04-12	15	david_miller	david.miller@example.com	202
10	105	25	35	2	Amazing gameplay and graphics!	2025-01-11	17	david_miller	david.miller@example.com	202
11	105	26	45	4	Not recommended. Needs improvements.	2024-03-10	26	david_miller	david.miller@example.com	202
12	106	27	6	4	Good, but with some bugs.	2024-05-25	15	emily_davis	emily.davis@example.com	202
13	106	30	36	3	Amazing gameplay and graphics!	2024-10-28	30	emily_davis	emily.davis@example.com	202
14	106	31	46	3	Amazing gameplay and graphics!	2024-12-11	17	emily_davis	emily.davis@example.com	202
15	107	32	7	4	Good, but with some bugs.	2025-01-22	12	chris_wilson	chris.wilson@example.com	202
16	107	34	27	2	Good, but with some bugs.	2025-01-26	27	chris_wilson	chris.wilson@example.com	202
17	107	35	37	4	Amazing gameplay and graphics!	2024-10-07	11	chris_wilson	chris.wilson@example.com	202
18	108	37	8	2	Good, but with some bugs.	2024-11-30	27	anna_taylor	anna.taylor@example.com	202
19	108	40	38	3	Good, but with some bugs.	2024-02-11	17	anna_taylor	anna.taylor@example.com	202
20	108	41	48	2	Good, but with some bugs.	2024-08-12	26	anna_taylor	anna.taylor@example.com	202
21	109	43	19	4	Not recommended. Needs improvements.	2024-03-27	44	james_white	james.white@example.com	202

Total rows: 40 Query complete 00:00:00.049

CRLF Ln 3, Col 1

Data Output Messages Notifications **Query** Query History

```
1 ▼ SELECT *
2   FROM purchase NATURAL JOIN game;
3
```

	game_id	purchase_id	user_id	transaction_date	amount	payment_method	discount_applied	title	genre	release_date	price
	integer	integer	integer	date	numeric (10,2)	character varying (50)	boolean	character varying (255)	character varying (100)	date	numeric (10,2)
1	18	1	141	2024-10-11	72.22	PayPal	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
2	18	2	141	2024-08-31	31.93	Credit Card	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
3	18	3	141	2025-01-10	58.58	PayPal	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
4	18	4	141	2024-03-23	98.09	Credit Card	true	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
5	18	5	141	2024-02-26	60.91	Credit Card	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
6	18	6	141	2024-07-28	65.80	Credit Card	true	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
7	18	7	141	2025-02-03	94.64	PayPal	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
8	18	8	141	2024-07-21	96.14	Credit Card	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
9	18	9	141	2024-03-21	80.73	PayPal	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
10	18	10	141	2024-03-13	44.47	PayPal	true	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
11	18	11	141	2024-07-16	89.67	Credit Card	true	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
12	18	12	141	2024-11-04	46.55	Credit Card	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
13	18	13	141	2025-02-09	75.28	Credit Card	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
14	18	14	141	2024-03-03	29.46	PayPal	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
15	18	15	141	2024-07-04	81.68	Credit Card	true	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
16	18	16	141	2024-04-28	49.18	PayPal	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
17	18	17	141	2024-02-26	80.28	Credit Card	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
18	18	18	141	2024-07-29	99.66	Credit Card	true	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
19	18	19	141	2024-03-18	52.14	PayPal	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
20	18	20	141	2024-10-24	21.53	Credit Card	false	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99
21	18	21	141	2024-12-18	108.36	Credit Card	true	Call of Duty: Black Ops 4	FPS	2018-10-12	53.99

Total rows: 50 Query complete 00:00:00.055

CRLF Ln 3, Col 1

## 7. SELF JOIN (15 pts)

SELF JOIN - Allows you to join a table with itself to perform various tasks such as comparing records and finding related data.

	Data Output	Messages	Notifications	Query	Query History
1 ✓ <code>SELECT a.username AS user1, b.username AS user2, a.email</code>					
2	<code>FROM useraccount a</code>				
3	<code>JOIN useraccount b ON a.email LIKE '%'    split_part(b.email, '@', 2);</code>				
4					
	user1 character varying (100) 	user2 character varying (100) 	email character varying (255) 		
45	alex_martin	amelia_walker	alex.martin@example.com		
46	alex_martin	jack_evans	alex.martin@example.com		
47	alex_martin	olivia_moore	alex.martin@example.com		
48	alex_martin	ethan_hill	alex.martin@example.com		
49	alex_martin	isabella_adams	alex.martin@example.com		
50	mia_scott	alex_martin	mia.scott@example.com		
51	mia_scott	mia_scott	mia.scott@example.com		
52	mia_scott	lucas_kelly	mia.scott@example.com		
53	mia_scott	grace_carter	mia.scott@example.com		
54	mia_scott	logan_brown	mia.scott@example.com		
55	mia_scott	zoe_anderson	mia.scott@example.com		
56	mia_scott	henry_taylor	mia.scott@example.com		
57	mia_scott	ellie_wilson	mia.scott@example.com		
58	mia_scott	ryan_lee	mia.scott@example.com		
59	mia_scott	chloe_jackson	mia.scott@example.com		
60	mia_scott	joshua_williams	mia.scott@example.com		
61	mia_scott	eva_morris	mia.scott@example.com		
62	mia_scott	andrew_hall	mia.scott@example.com		
63	mia_scott	maya_robinson	mia.scott@example.com		
64	mia_scott	liam_hill	mia.scott@example.com		
65	mia_scott	sienna_clark	mia.scott@example.com		
..	.....	.....	.....	.....	.....
Total rows: 2401		Query complete 00:00:00.069			

Data Output Messages Notifications Query Query History

```
1 ✓ SELECT a.title AS game1, b.title AS game2, a.release_date
2   FROM game a
3   JOIN game b ON a.release_date = b.release_date AND a.game_id <> b.game_id;
4 |
```

	game1 character varying (255) 	game2 character varying (255) 	release_date date 
1	Game A	Game B	2022-01-15
2	Game B	Game A	2022-01-15
3	Game C	Game D	2023-03-20
4	Game D	Game C	2023-03-20
5	Game E	Game F	2023-05-10
6	Game F	Game E	2023-05-10

Data Output Messages Notifications Query Query History

```
1 ✓ SELECT a.user_id, b.user_id, a.game_id, a.rating AS rating_a, b.rating AS rating_b
2   FROM review a
3   JOIN review b ON a.game_id = b.game_id AND a.user_id <> b.user_id;
4 |
```

	user_id integer 	user_id integer 	game_id integer 	rating_a integer 	rating_b integer 
1	104	121	4	4	5
2	105	122	5	4	4
3	124	123	2	5	5
4	124	102	2	5	5
5	102	123	2	5	5
6	102	124	2	5	5
7	103	126	3	6	3
8	103	125	3	6	4
9	121	104	4	5	4
10	122	105	5	4	4
11	123	102	2	5	5
12	123	124	2	5	5
13	125	126	3	4	3
14	125	103	3	4	6
15	126	125	3	3	4
16	126	103	3	3	6

## 5. Query Execution & Results Analysis (Based on Assignment 5)

### A. Grouping Data (GROUP BY Clause)

1. Here I wrote a program which calculates group of data ewith help “SUM()” operator.

Data Output			Messages	Notifications	Query	Query History
1	v	SELECT game_id, COUNT(*) AS total_reviews				
2		FROM review				
3		GROUP BY game_id;				
4						
	game_id	total_reviews				
1	42	1				
2	29	1				
3	4	2				
4	46	1				
5	32	1				
6	10	1				
7	7	1				
8	35	1				
9	45	1				
10	38	1				
11	6	1				
12	26	1				
13	48	1				
14	12	1				
15	39	1				
16	24	1				
17	19	1				
18	36	1				
19	25	1				
20	30	1				
21	50	1				
22	40	1				
Total rows: 40		Query complete 00:00:00.054				

2. “AVG” operator group two culums and calculetes they are together.

	user_id	game_id	avg_rating	
	integer	integer	numeric	
1	124	2	5.000000000000000	
2	109	29	6.000000000000000	
3	107	47	6.000000000000000	
4	102	32	5.000000000000000	
5	104	4	4.000000000000000	
6	122	5	4.000000000000000	
7	125	3	4.000000000000000	
8	106	26	5.000000000000000	
9	104	44	2.000000000000000	
10	105	25	2.000000000000000	
11	110	50	4.000000000000000	
12	126	3	3.000000000000000	
13	107	17	5.000000000000000	
14	110	10	3.000000000000000	
15	109	49	2.000000000000000	
16	106	36	3.000000000000000	
17	103	23	5.000000000000000	
18	102	22	2.000000000000000	
19	108	48	2.000000000000000	
20	102	12	2.000000000000000	
21	105	35	2.000000000000000	
22	110	20	5.000000000000000	
Total rows: 46		Query complete 00:00:00.057		

3. That operator filtering data which count <5, and output for us some data.

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT game_id, COUNT(*) AS review_count
2   FROM review
3   GROUP BY game_id
4   HAVING COUNT(*) < 5;
5
```

	game_id integer	review_count bigint
1	42	1
2	29	1
3	4	2
4	46	1
5	32	1
6	10	1
7	7	1
8	35	1
9	45	1
10	38	1
11	6	1
12	26	1
13	48	1
14	12	1
15	39	1
16	24	1
17	19	1
18	36	1
19	25	1
20	30	1
21	50	1
22	40	1

Total rows: 40      Query complete 00:00:00.049

4. Here we are group data from 2 different tables and calculate together.

Data Output	Messages	Notifications	Query	Query History
1 ↴ <code>SELECT g.title, COUNT(r.review_id) AS total_reviews</code>				
2 <code>FROM game g</code>				
3 <code>LEFT JOIN review r ON g.game_id = r.game_id</code>				
4 <code>GROUP BY g.title;</code>				
	title character varying (255)		total_reviews bigint	
1	Stellaris		1	
2	Dragon Age: Inquisition		0	
3	Game F		0	
4	Bayonetta 2		1	
5	Hitman 3		0	
6	Assassin's Creed Valhalla		2	
7	Game B		0	
8	Hades		1	
9	Spider-Man		1	
10	Call of Duty: Black Ops 4		1	
11	Divinity: Original Sin II		1	
12	Death Stranding		1	
13	Ghost of Tsushima		1	
14	Grand Theft Auto V		3	
15	No Man's Sky		1	
16	Hollow Knight		0	
17	Game E		0	
18	Dishonored 2		1	
19	Fortnite		1	
20	The Elder Scrolls V: Skyrim		2	
21	Cuphead		1	
22	Destiny 2		1	
Total rows: 55		Query complete 00:00:00.061		

## B. Advanced Grouping & Set Operations

1. Filter a column using a single-value subquery.

Data Output Messages Notifications Query Query History

```
1 ✓ SELECT review_id, user_id, game_id, rating
2   FROM review
3 WHERE rating = (SELECT MAX(rating) FROM review);
4
```

Data Output Messages Notifications Query Query Histc

	review_id [PK] integer	user_id integer	game_id integer	rating integer
1	12	103	3	6
2	19	104	24	6
3	36	107	47	6
4	44	109	29	6

## 2. IN Subquery – Use an IN() subquery to filter rows based on multiple values

	game_id [PK] integer	title character varying (255)	publisher_id integer
1	4	Assassin's Creed Valhalla	38
2	6	FIFA 23	38
3	16	Ghost of Tsushima	38
4	25	DOOM Eternal	38
5	30	Hades	38
6	34	Hitman 3	38
7	40	Returnal	38
8	44	Cyberpunk 2077	38
9	47	Horizon Forbidden West	38
10	2	The Witcher 3: Wild Hunt	38
11	3	Grand Theft Auto V	38
12	5	The Elder Scrolls V: Skyrim	38
13	7	Final Fantasy XV	38
14	8	Dark Souls III	38
15	9	Overwatch	38
16	10	Resident Evil 2	38
17	11	Metal Gear Solid V	38
18	12	Uncharted 4: A Thief's End	38
19	13	Spider-Man	38
20	14	Horizon Zero Dawn	38
21	15	Sekiro: Shadows Die Twice	38
22	17	Yakuza: Like a Dragon	38

Total rows: 53      Query complete 00:00:00.089

3. Correlated Subquery – Implement a WHERE condition that depends on an outer query.

Data Output Messages Notifications Query Query History

```
1 ✓ SELECT DISTINCT r1.user_id
2   FROM review r1
3 WHERE (SELECT COUNT(*) FROM review r2 WHERE r2.user_id = r1.user_id) > 1;
4 |
```

	user_id
1	102
2	103
3	104
4	105
5	106
6	107
7	108
8	109
9	110

4. EXISTS Subquery – Return results based on whether the subquery has at least one match.

Data Output Messages Notifications Query Query History

```
1 ▾ SELECT g.game_id, g.title
2   FROM game g
3   WHERE EXISTS (SELECT 1 FROM review r WHERE r.game_id = g.game_id);
4
```

	game_id [PK] integer	title character varying (255)
1	4	Assassin's Creed Valhalla
2	6	FIFA 23
3	16	Ghost of Tsushima
4	25	DOOM Eternal
5	30	Hades
6	44	Cyberpunk 2077
7	47	Horizon Forbidden West
8	2	The Witcher 3: Wild Hunt
9	3	Grand Theft Auto V
10	5	The Elder Scrolls V: Skyrim
11	7	Final Fantasy XV
12	8	Dark Souls III
13	10	Resident Evil 2
14	12	Uncharted 4: A Thief's End
15	13	Spider-Man
16	14	Horizon Zero Dawn
17	17	Xenoblade Chronicles 2
18	18	Call of Duty: Black Ops 4
19	19	Titanfall 2
20	20	Pillars of Eternity II
21	22	The Walking Dead: Seaso...
22	22	Control

Total rows: 40      Query complete 00:00:00.059

## 5. ALL Subquery – Compare a column against all values returned by a subquery.

Data Output Messages Notifications Query Query History

```
1 ▾ SELECT r.review_id, r.user_id, r.game_id, r.rating
2   FROM review r
3   JOIN (
4       SELECT game_id, AVG(rating) AS avg_rating
5         FROM review
6        GROUP BY game_id
7   ) avg_ratings ON r.game_id = avg_ratings.game_id
8   WHERE r.rating > avg_ratings.avg_rating;
9
```

	review_id [PK] integer	user_id integer	game_id integer	rating integer
1	12	103	3	6
2	211	121	4	5

## 6. Row Subquery – Compare multiple columns against subquery results using = AS <> operators.

The screenshot shows a database interface with a query editor and a results grid. The query editor contains the following SQL code:

```
1 v SELECT g.title, g.release_date, g.publisher_id
2   FROM game g
3   JOIN (
4     SELECT publisher_id, MIN(release_date) AS earliest_date
5       FROM game
6      GROUP BY publisher_id
7   ) sub ON g.publisher_id = sub.publisher_id AND g.release_date = sub.earliest_date;
8 |
```

The results grid displays the following data:

	<b>title</b> character varying (255)	<b>release_date</b> date	<b>publisher_id</b> integer
1	Game A	2022-01-15	1
2	Game B	2022-01-15	2
3	Game C	2023-03-20	3
4	Game D	2023-03-20	4
5	Game E	2023-05-10	5
6	Game F	2023-05-10	6
7	World of Warcraft	2004-11-23	38

## 6.Advanced SQL Techniques (Based on Assignment 6 )

### A. Window Functions

#### 1. In task 1 I wrote a code which calculated 2 rows with help LAG() and LEAD()

Data Output Messages Notifications Query Query Query History

```
1 ✓ SELECT review_id, game_id, user_id, rating,
2      LAG(rating) OVER (PARTITION BY game_id ORDER BY review_id) AS prev_rating,
3      rating - LAG(rating) OVER (PARTITION BY game_id ORDER BY review_id) AS rating_diff
4  FROM review;
5
```

	review_id [PK] integer	game_id integer	user_id integer	rating integer	prev_rating integer	rating_diff integer
1	7	2	102	5	[null]	[null]
2	233	2	123	5	5	0
3	244	2	124	5	5	0
4	12	3	103	6	[null]	[null]
5	255	3	125	4	6	-2
6	266	3	126	3	4	-1
7	17	4	104	4	[null]	[null]
8	211	4	121	5	4	1
9	22	5	105	4	[null]	[null]
10	222	5	122	4	4	0
11	27	6	106	4	[null]	[null]
12	32	7	107	4	[null]	[null]
13	37	8	108	2	[null]	[null]
14	47	10	110	3	[null]	[null]
15	8	12	102	2	[null]	[null]
16	13	13	103	4	[null]	[null]
17	18	14	104	3	[null]	[null]
18	28	16	106	5	[null]	[null]
19	33	17	107	5	[null]	[null]
20	38	18	108	5	[null]	[null]
21	43	19	109	4	[null]	[null]
22	40	20	110	2	[null]	[null]

Total rows: 46    Query complete 00:00:00.054

2. IN task 2 I calculated percentage\_contribution with operators SUM() and OVER(), that was count in decimal numbers

Data Output Messages Notifications Query Query Query History

```

1 ✓ SELECT game_id, COUNT(review_id) AS total_reviews,
2     COUNT(review_id) * 100.0 / SUM(COUNT(review_id)) OVER () AS percentage_contribution
3     FROM review
4     GROUP BY game_id;
5

```

	game_id integer	total_reviews bigint	percentage_contribution numeric
1	42	1	2.1739130434782609
2	29	1	2.1739130434782609
3	4	2	4.3478260869565217
4	46	1	2.1739130434782609
5	32	1	2.1739130434782609
6	10	1	2.1739130434782609
7	7	1	2.1739130434782609
8	35	1	2.1739130434782609
9	45	1	2.1739130434782609
10	38	1	2.1739130434782609
11	6	1	2.1739130434782609
12	26	1	2.1739130434782609
13	48	1	2.1739130434782609
14	12	1	2.1739130434782609
15	39	1	2.1739130434782609
16	24	1	2.1739130434782609
17	19	1	2.1739130434782609
18	36	1	2.1739130434782609
19	25	1	2.1739130434782609
20	30	1	2.1739130434782609
21	50	1	2.1739130434782609
22	40	1	2.1739130434782609
Total rows: 40		Query complete 00:00:00.066	

3. In task 3 I need to show averages over a 3 columns. I use operator AVG()

Data Output Messages Notifications Query Query History

```
1 ✓ SELECT review_id, game_id, rating,
2          AVG(rating) OVER (PARTITION BY game_id ORDER BY review_id ROWS BETWEEN 2 PRECEDING AND CURRENT ROW) AS moving_avg
3      FROM review;
4  |
```

	review_id [PK] integer	game_id integer	rating integer	moving_avg numeric
1	7	2	5	5.00000000000000000000
2	233	2	5	5.00000000000000000000
3	244	2	5	5.00000000000000000000
4	12	3	6	6.00000000000000000000
5	255	3	4	5.00000000000000000000
6	266	3	3	4.33333333333333333333
7	17	4	4	4.00000000000000000000
8	211	4	5	4.50000000000000000000
9	22	5	4	4.00000000000000000000
10	222	5	4	4.00000000000000000000
11	27	6	4	4.00000000000000000000
12	32	7	4	4.00000000000000000000
13	37	8	2	2.00000000000000000000
14	47	10	3	3.00000000000000000000
15	8	12	2	2.00000000000000000000
16	13	13	4	4.00000000000000000000
17	18	14	3	3.00000000000000000000
18	28	16	5	5.00000000000000000000
19	33	17	5	5.00000000000000000000
20	38	18	5	5.00000000000000000000
21	43	19	4	4.00000000000000000000
22	40	20	3	3.00000000000000000000

Total rows: 46    Query complete 00:00:00.078

## B.Advanced Grouping & Set Operations

### 1. GROUPING SETS (Sum, Count, Grand Total calculations).

In this task uses GROUPING SETS and the COALESCE function to group review data by game\_id, user\_id, and calculate the total count of reviews.

Data Output Messages Notifications Query Query History

```
1 ▼ SELECT
2     COALESCE(CAST(game_id AS TEXT), 'TOTAL') AS game_id,
3     COALESCE(CAST(user_id AS TEXT), 'TOTAL') AS user_id,
4     COUNT(review_id) AS review_count
5 FROM review
6 GROUP BY GROUPING SETS ((game_id), (user_id), ());
7
```

	game_id 	user_id 	review_count 
1	TOTAL	TOTAL	46
2	42	TOTAL	1
3	29	TOTAL	1
4	4	TOTAL	2
5	46	TOTAL	1
6	32	TOTAL	1
7	10	TOTAL	1
8	7	TOTAL	1
9	35	TOTAL	1
10	45	TOTAL	1
11	38	TOTAL	1
12	6	TOTAL	1
13	26	TOTAL	1
14	48	TOTAL	1
15	12	TOTAL	1
16	39	TOTAL	1
17	24	TOTAL	1
18	19	TOTAL	1
19	36	TOTAL	1
20	25	TOTAL	1
21	30	TOTAL	1
22	50	TOTAL	1

Total rows: 56    Query complete 00:00:00.049

**2. CUBE (All possible totals and subtotals for three columns).**  
I uses the CUBE operator to generate all possible combinations of grouping for game\_id, user\_id, and rating from the review table.

Data Output Messages Notifications Query Query History

```
1 ✓ SELECT game_id, user_id, rating, COUNT(*) AS review_count
2   FROM review
3   GROUP BY CUBE (game_id, user_id, rating);
4 |
```

	game_id integer	user_id integer	rating integer	review_count bigint
1	[null]	[null]	[null]	46
2	4	121	5	1
3	8	108	2	1
4	3	125	4	1
5	7	107	4	1
6	47	107	6	1
7	16	106	5	1
8	27	107	2	1
9	12	102	2	1
10	20	110	2	1
11	36	106	3	1
12	3	103	6	1
13	26	106	5	1
14	5	105	4	1
15	18	108	5	1
16	45	105	4	1
17	4	104	4	1
18	37	107	4	1
19	2	102	5	1
20	49	109	2	1
21	33	103	4	1
22	24	104	6	1

Total rows: 231      Query complete 00:00:00.073

### 3. ROLLUP (Hierarchical totals for two columns).

This query uses the ROLLUP operator to generate hierarchical totals for game\_id and user\_id from the review table.

Data Output Messages Notifications Query Query History

```
1 v SELECT game_id, user_id, COUNT(review_id) AS review_count
2   FROM review
3   GROUP BY ROLLUP (game_id, user_id);
```

	game_id integer	user_id integer	review_count bigint
1	[null]	[null]	46
2	26	106	1
3	25	105	1
4	23	103	1
5	37	107	1
6	17	107	1
7	29	109	1
8	42	102	1
9	36	106	1
10	50	110	1
11	3	126	1
12	13	103	1
13	35	105	1
14	33	103	1
15	10	110	1
16	19	109	1
17	18	108	1
18	48	108	1
19	14	104	1
20	28	108	1
21	47	107	1
22	40	100	1

Total rows: 87

Query complete 00:00:00.073

#### 4. INTERSECT (Combining filtered queries).

Here i retrieves all games from the game table where the genre includes "Actions" or "Sports" in its name.



The screenshot shows a PostgreSQL pgAdmin interface. At the top, there are tabs for Data Output, Messages, Notifications, Query (which is selected), and Query History. Below the tabs is a code editor with two lines of SQL:

```
1 SELECT game_id, genre FROM game WHERE genre LIKE '%Actions%' OR genre LIKE '%Sports%';
2
```

Below the code editor is a toolbar with various icons: a plus sign, a file, a dropdown, a clipboard, another dropdown, a trash can, a database icon, a download arrow, a graph, and the word "SQL".

The main area displays a table with three columns: game\_id, genre, and a third column which appears to be a primary key indicator. The table has one row of data:

	game_id [PK] integer	genre character varying (100)
1	6	Sports