User_ID	Username	Email	Game_ID	Game_Name	Review	Transaction_Data	Amount
1	Dimas123	dimas@mail.com	101	Adventure Land	Awesome	2024-12-01	50000

1NF

Tbl User

User_ID	Username	Email
1	Dimas123	dimas@mail.com

Tbl Games

Game_ID	Game_Name	Gendre

Tbl Review

Review_ID	User_ID	Game_ID	Review

Tbl Transaction

Transaction_ID	User_ID	Game_ID	Transaction_Date	Amount

2NF

Tbl User

User_ID	Username	Email

Tbl Games

Game_ID	Game_Name	Gendre

Tbl Review

Review_ID	User_ID	Game_ID	Review

Tbl Transaction

Transaction_ID	User_ID	Game_ID	Transaction_Date	Amount

3NF

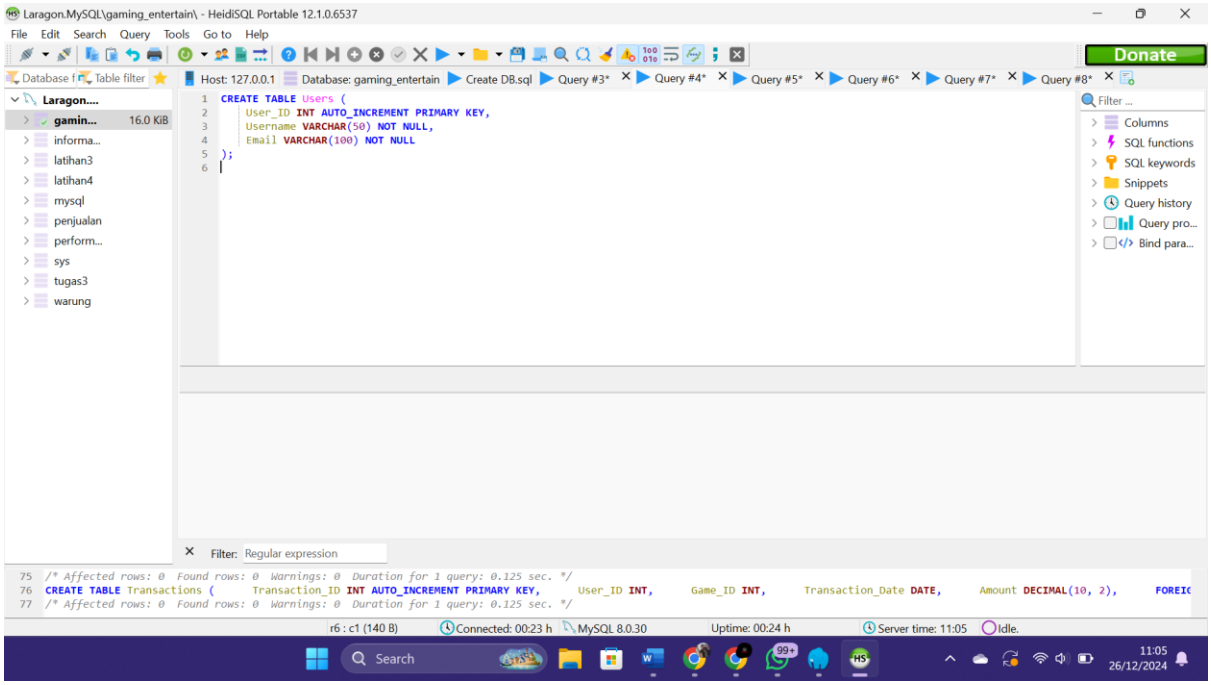
Tbl Gendre

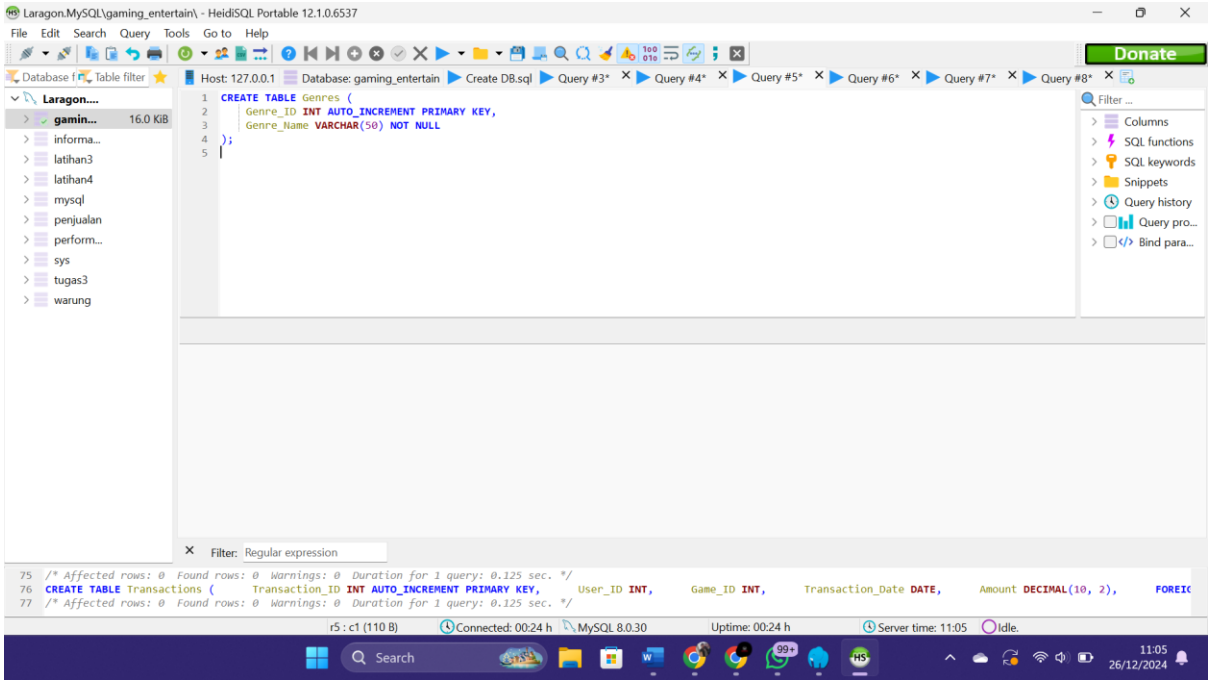
Gendre_ID	Gendre_Name

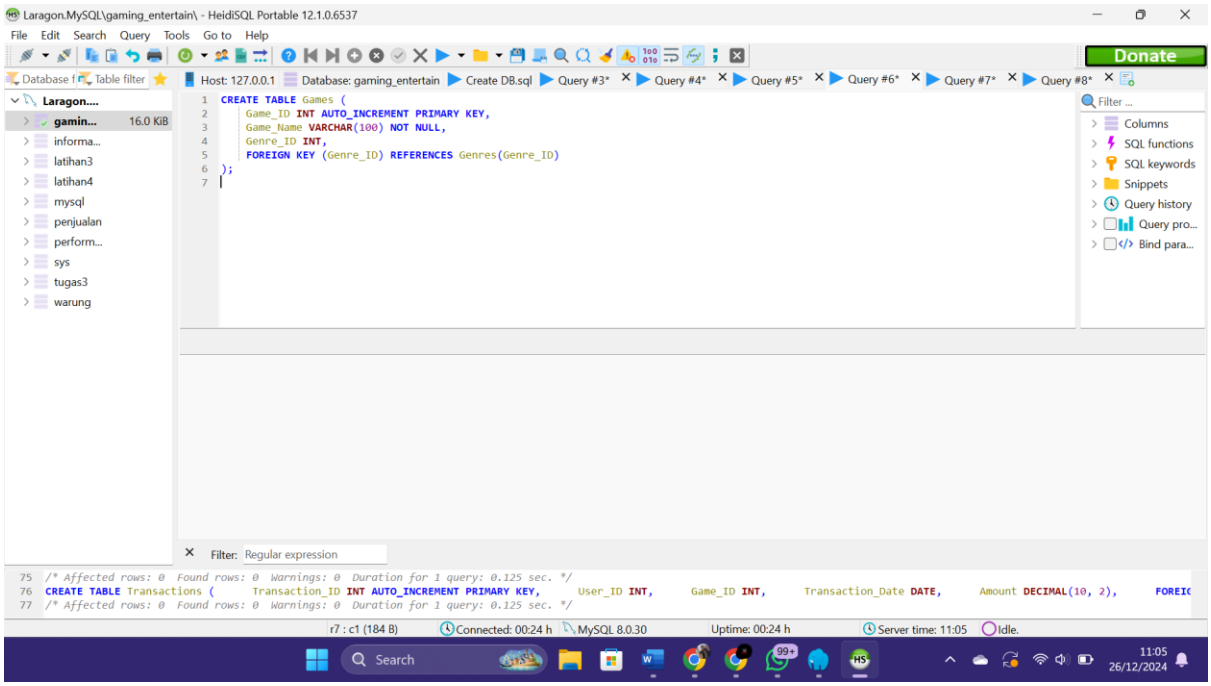
Tbl Games

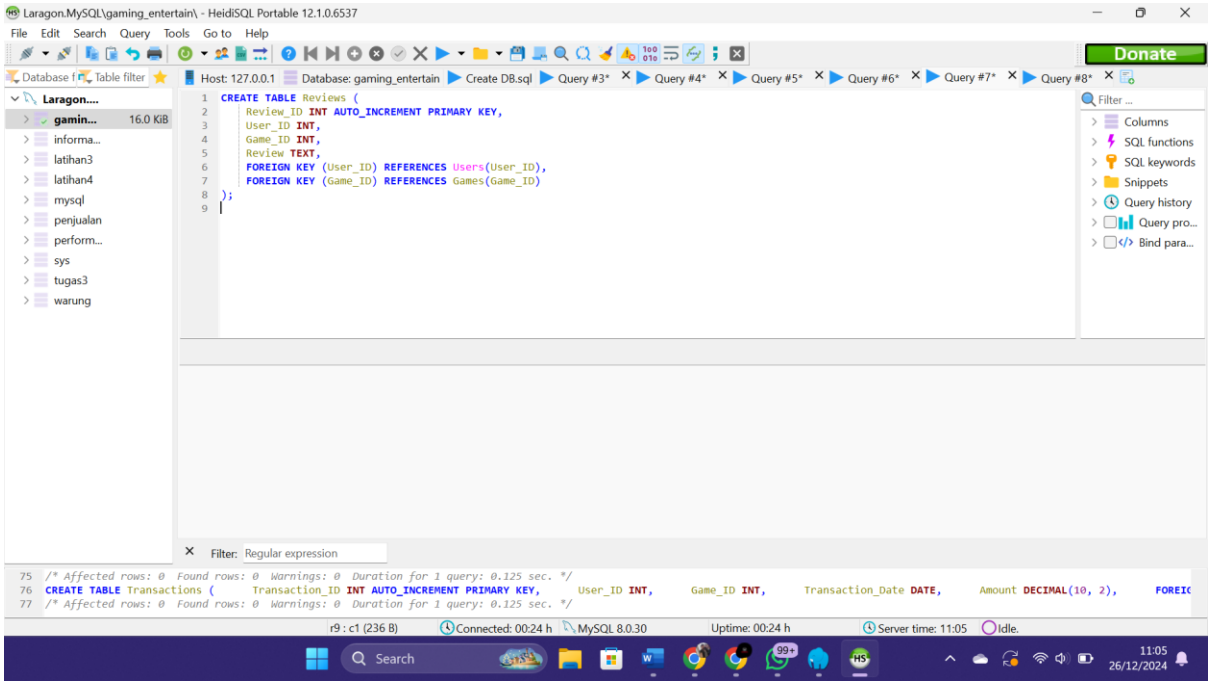
Game_ID	Game_Name	Gendre_ID

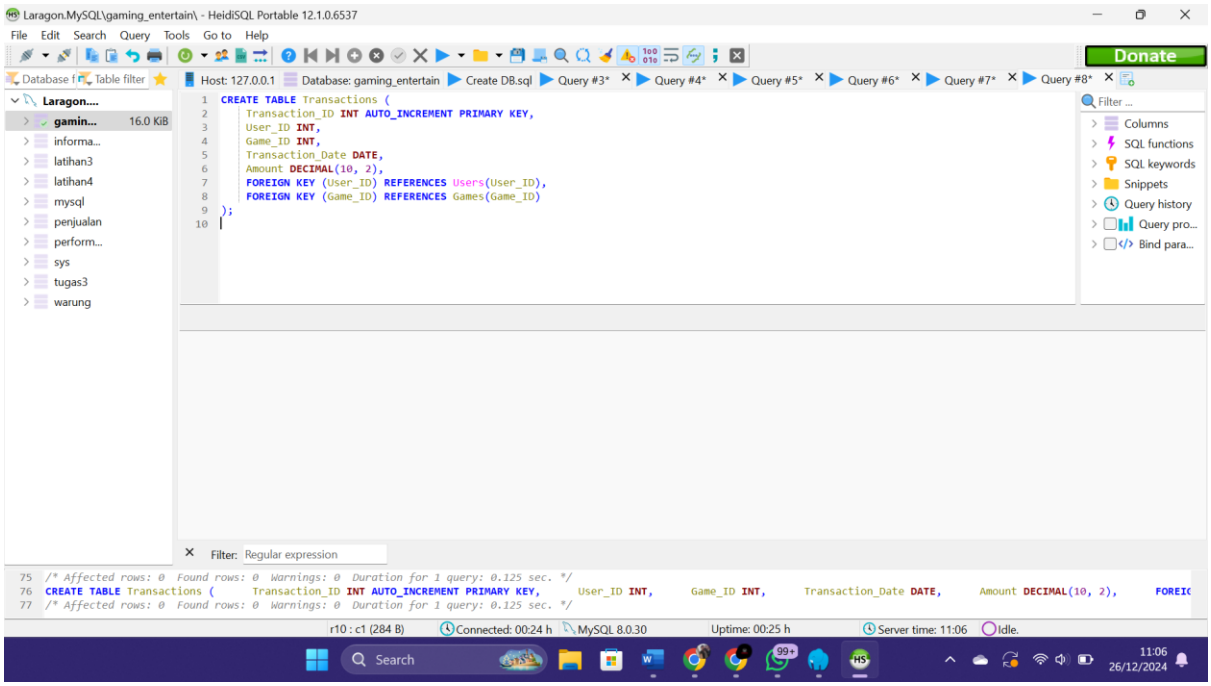
Struktur tabel



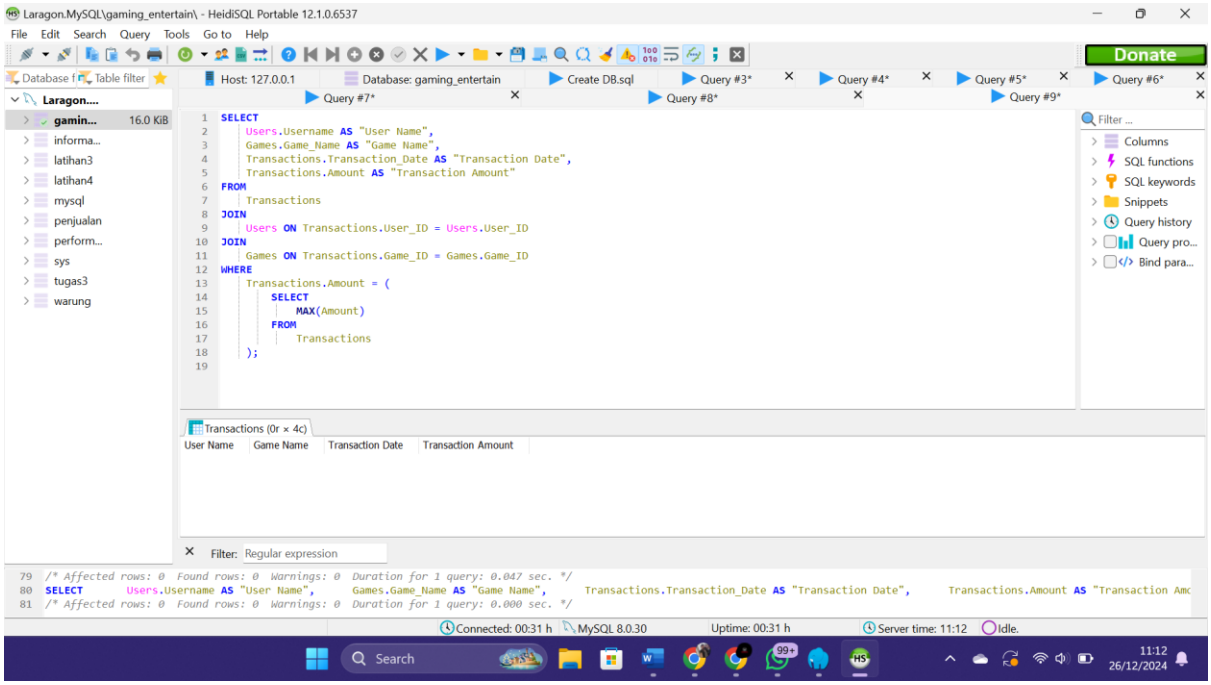



Laragon.MySQL\gaming_entertain\ - HeidiSQL Portable 12.1.0.6537
File Edit Search Query Tools Go to Help
Donate
Host: 127.0.0.1 Database: gaming_entertain Create DB.sql Query #3* Query #4* Query #5* Query #6* Query #7* Query #8*
CREATE TABLE Games (
Game_ID INT AUTO_INCREMENT PRIMARY KEY,
Game_Name VARCHAR(100) NOT NULL,
Genre_ID INT,
FOREIGN KEY (Genre_ID) REFERENCES Genres(Genre_ID)
);
Laragon....
gamin... 16.0 KiB
informa...
latihan3
latihan4
mysql
penjualan
perform...
sys
tugas3
warung
Columns
SQL functions
SQL keywords
Snippets
Query history
Filter: Regular expression
/* Affected rows: 0 Found rows: 0 Warnings: 0 Duration for 1 query: 0.125 sec. */
CREATE TABLE Transactions (Transaction_ID INT AUTO_INCREMENT PRIMARY KEY, User_ID INT, Game_ID INT, Transaction_Date DATE, Amount DECIMAL(10, 2),
/* Affected rows: 0 Found rows: 0 Warnings: 0 Duration for 1 query: 0.125 sec. */
r7 : c1 (184 B) Connected: 00:24 h MySQL 8.0.30 Uptime: 00:24 h Server time: 11:05 Idle.
Search
11:05
26/12/2024

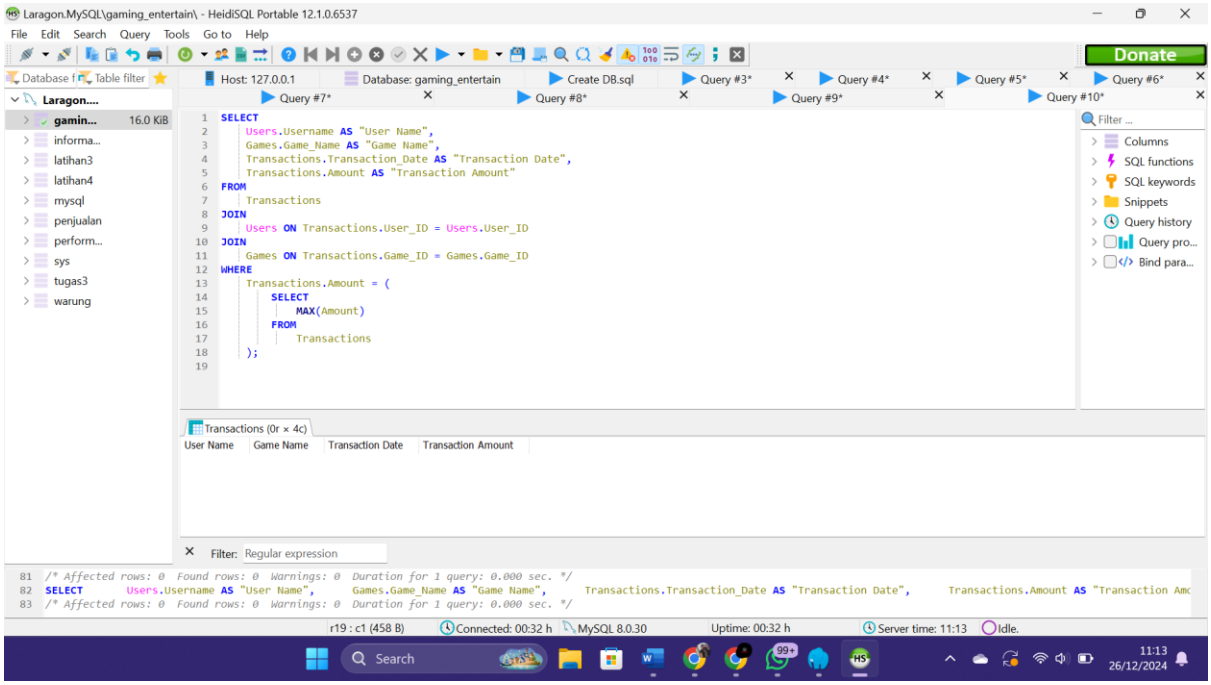

Laragon.MySQL\gaming_entertain\ - HeidiSQL Portable 12.1.0.6537
File Edit Search Query Tools Go to Help
Donate
Host: 127.0.0.1 Database: gaming_entertain Create DB.sql Query #3* Query #4* Query #5* Query #6* Query #7* Query #8*
CREATE TABLE Reviews (
Review_ID INT AUTO_INCREMENT PRIMARY KEY,
User_ID INT,
Game_ID INT,
Review TEXT,
FOREIGN KEY (User_ID) REFERENCES Users(User_ID),
FOREIGN KEY (Game_ID) REFERENCES Games(Game_ID)
);
Laragon....
gamin... 16.0 KiB
informa...
latihan3
latihan4
mysql
penjualan
perform...
sys
tugas3
warung
Columns
SQL functions
SQL keywords
Snippets
Query history
Filter: Regular expression
/* Affected rows: 0 Found rows: 0 Warnings: 0 Duration for 1 query: 0.125 sec. */
CREATE TABLE Transactions (Transaction_ID INT AUTO_INCREMENT PRIMARY KEY, User_ID INT, Game_ID INT, Transaction_Date DATE, Amount DECIMAL(10, 2),
/* Affected rows: 0 Found rows: 0 Warnings: 0 Duration for 1 query: 0.125 sec. */
r9 : c1 (236 B) Connected: 00:24 h MySQL 8.0.30 Uptime: 00:24 h Server time: 11:05 Idle.
Search
11:05
26/12/2024



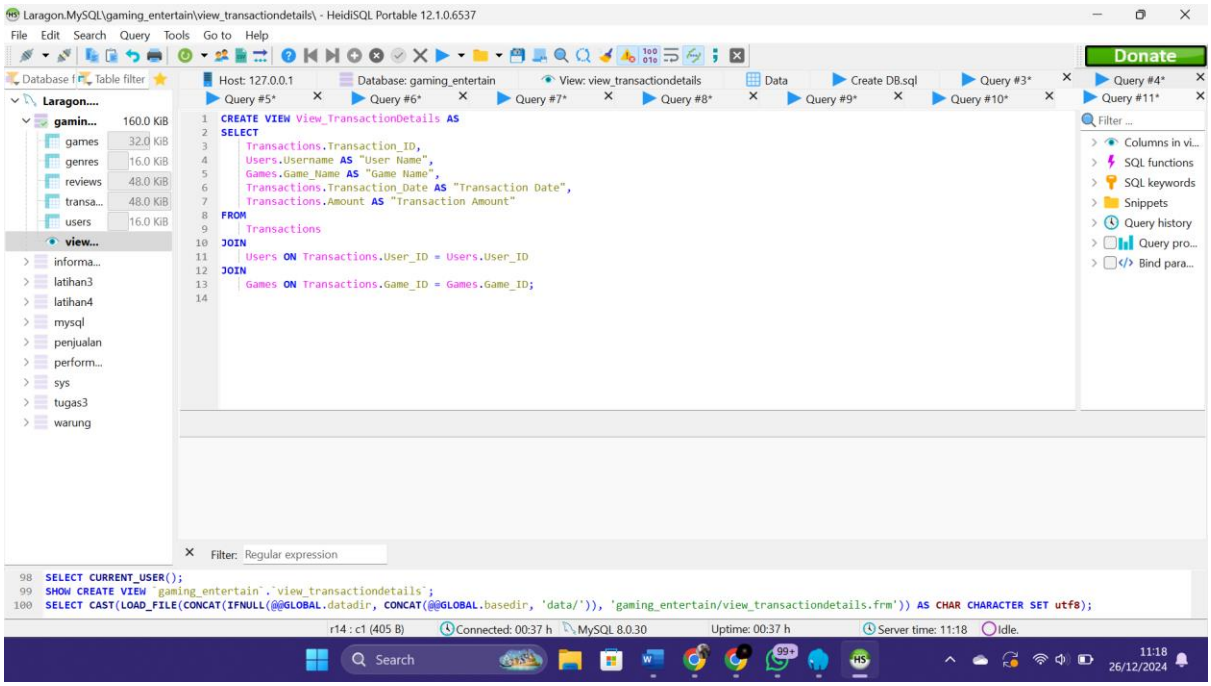
Nomor 2. Menggunakan Join

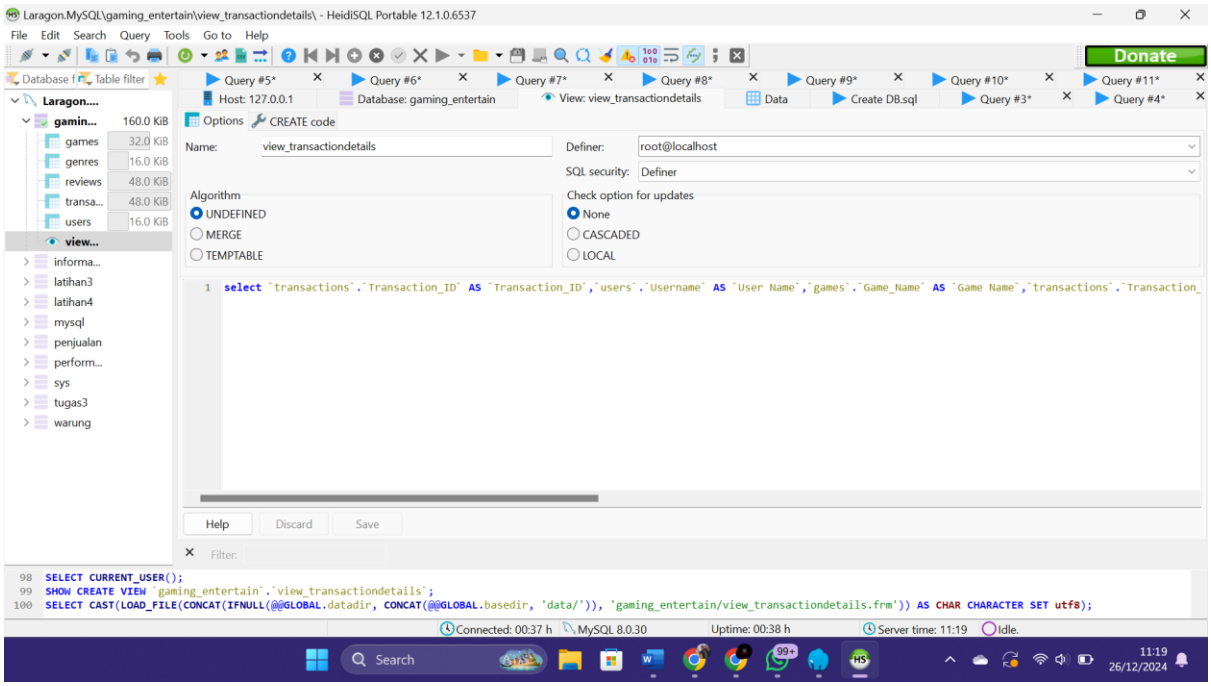


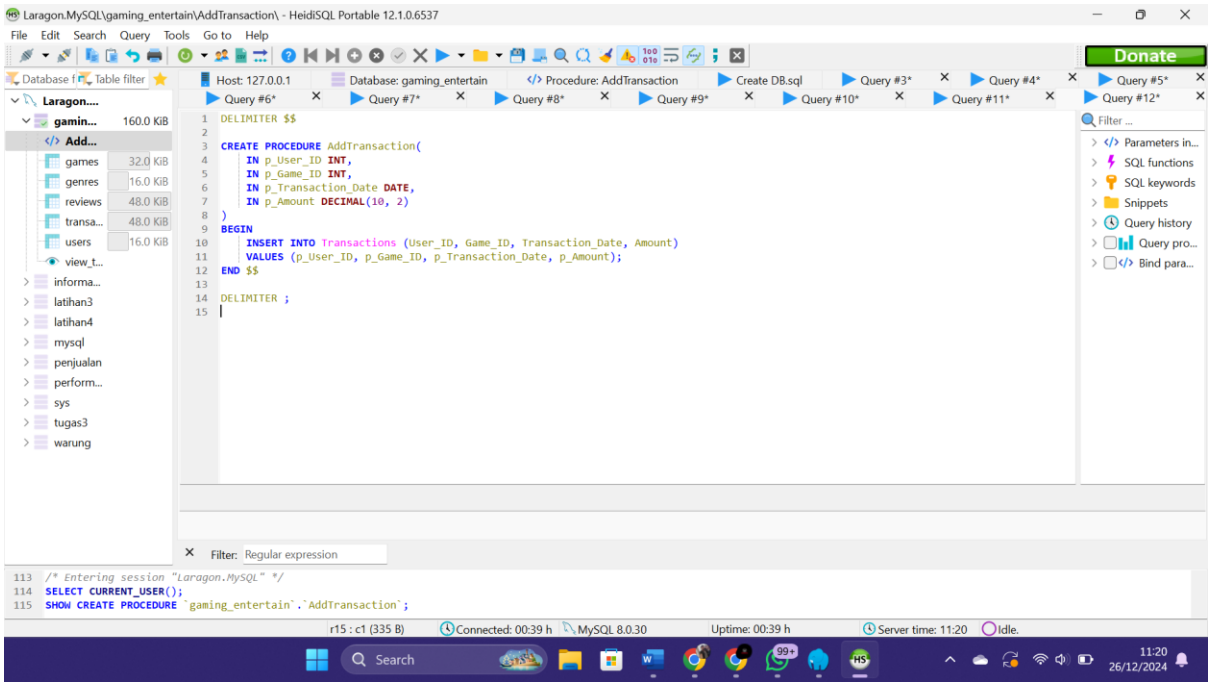
Menggunakan Subquery

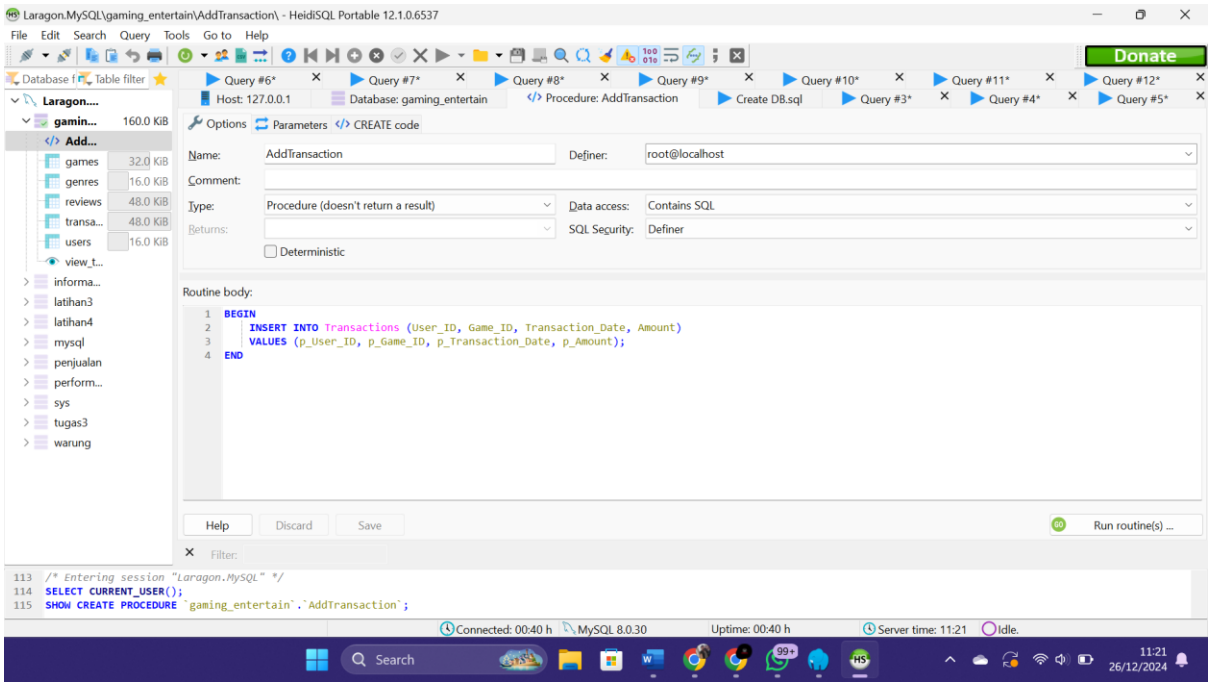


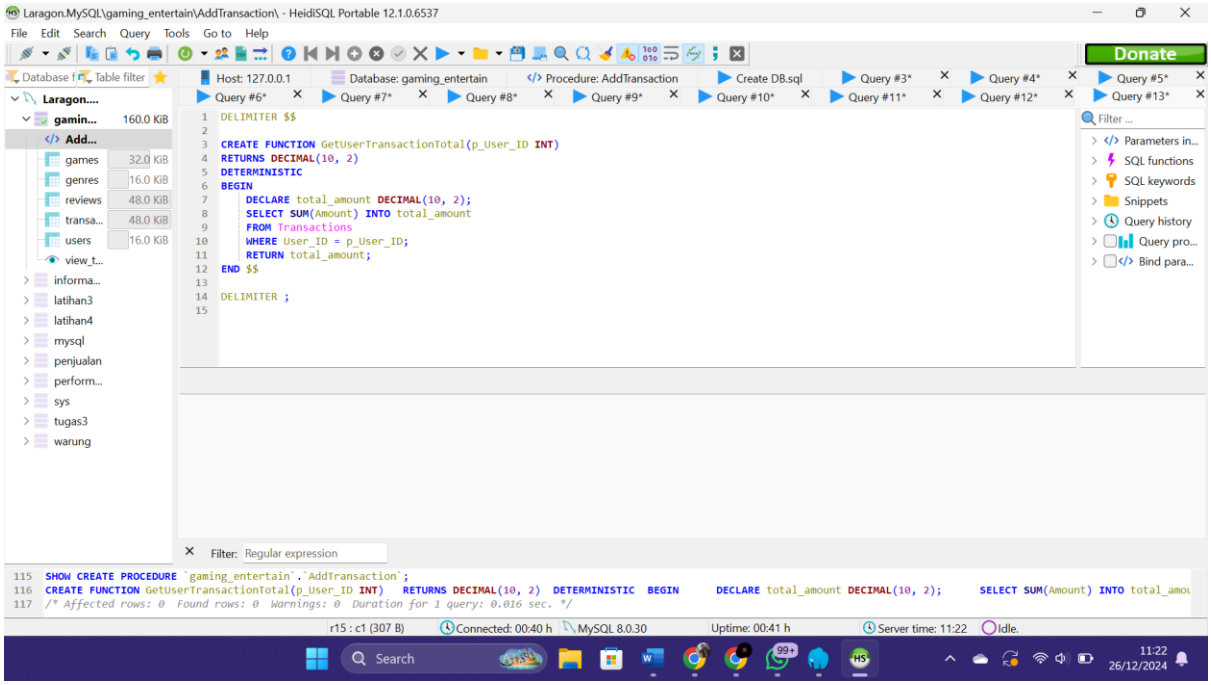
NOMOR 3

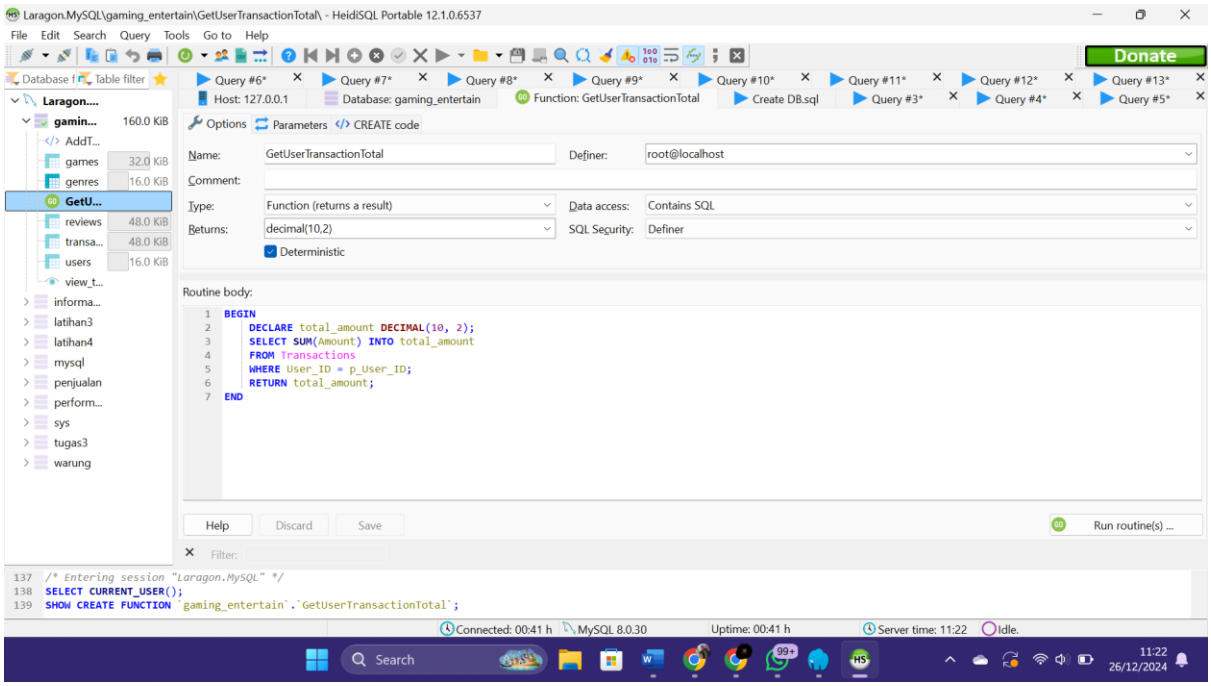



Laragon.MySQL\gaming_entertain\view_transactiondetails\ - HeidiSQL Portable 12.1.0.6537
File Edit Search Query Tools Go to Help
Donate
Query #5* Query #6* Query #7* Query #8* Query #9* Query #10* Query #11*
Host: 127.0.0.1 Database: gaming_entertain View: view_transactiondetails Data Create DB.sql Query #3* Query #4*
Options CREATE code
Name: view_transactiondetails
Definer: root@localhost
SQL security: Definer
Algorithm
UNDEFINED
MERGE
TEMPTABLE
Check option for updates
None
CASCADED
LOCAL
select `transactions`.`Transaction_ID` AS `Transaction_ID`,`users`.`Username` AS `User Name`,`games`.`Game_Name` AS `Game Name`,`transactions`.`Transaction
Help Discard Save
SELECT CURRENT_USER();
SHOW CREATE VIEW `gaming_entertain`.`view_transactiondetails`;
SELECT CAST(LOAD_FILE(CONCAT(IFNULL(@@GLOBAL.datadir, CONCAT(@@GLOBAL.basedir, 'data/')), 'gaming_entertain/view_transactiondetails.frm')) AS CHAR CHARACTER SET utf8);
Connected: 00:37 h MySQL 8.0.30 Uptime: 00:38 h Server time: 11:19 Idle.
Search
11:19
26/12/2024


Laragon.MySQL\gaming_entertain\AddTransaction\ - HeidiSQL Portable 12.1.0.6537
File Edit Search Query Tools Go to Help
Donate
Host: 127.0.0.1 Database: gaming_entertain Procedure: AddTransaction Create DB.sql Query #3* Query #4* Query #5*
Query #6* Query #7* Query #8* Query #9* Query #10* Query #11* Query #12*
DELIMITER $$
CREATE PROCEDURE AddTransaction(
IN p_User_ID INT,
IN p_Game_ID INT,
IN p_Transaction_Date DATE,
IN p_Amount DECIMAL(10, 2)
)
BEGIN
INSERT INTO Transactions (User_ID, Game_ID, Transaction_Date, Amount)
VALUES (p_User_ID, p_Game_ID, p_Transaction_Date, p_Amount);
END $$
DELIMITER ;
Parameters in...
SQL functions
SQL keywords
Snippets
Query history
Query pro...
Bind para...
Filter: Regular expression
/* Entering session "Laragon.MySQL" */
SELECT CURRENT_USER();
SHOW CREATE PROCEDURE `gaming_entertain`.`AddTransaction`;
r15 : c1 (335 B) Connected: 00:39 h MySQL 8.0.30 Uptime: 00:39 h Server time: 11:20 Idle.
Search
11:20
26/12/2024


Laragon.MySQL\gaming_entertain\AddTransaction\ - HeidiSQL Portable 12.1.0.6537
File Edit Search Query Tools Go to Help
Donate
Query #6* Query #7* Query #8* Query #9* Query #10* Query #11* Query #12*
Host: 127.0.0.1 Database: gaming_entertain Procedure: AddTransaction Create DB.sql Query #3* Query #4* Query #5*
Options Parameters CREATE code
Name: AddTransaction
Definer: root@localhost
Comment:
Type: Procedure (doesn't return a result)
Data access: Contains SQL
Returns:
SQL Security: Definer
Deterministic
Routine body:
BEGIN
INSERT INTO Transactions (User_ID, Game_ID, Transaction_Date, Amount)
VALUES (p_User_ID, p_Game_ID, p_Transaction_Date, p_Amount);
END
Help Discard Save Run routine(s) ...
/* Entering session "Laragon.MySQL" */
SELECT CURRENT_USER();
SHOW CREATE PROCEDURE `gaming_entertain`.`AddTransaction`;
Connected: 00:40 h MySQL 8.0.30 Uptime: 00:40 h Server time: 11:21 Idle.
11:21
26/12/2024


Laragon.MySQL\gaming_entertain\AddTransaction\ - HeidiSQL Portable 12.1.0.6537
File Edit Search Query Tools Go to Help
Donate
Host: 127.0.0.1 Database: gaming_entertain Procedure: AddTransaction Create DB.sql Query #3* Query #4* Query #5*
Query #6* Query #7* Query #8* Query #9* Query #10* Query #11* Query #12* Query #13*
DELIMITER $$
CREATE FUNCTION GetUserTransactionTotal(p_User_ID INT)
RETURNS DECIMAL(10, 2)
DETERMINISTIC
BEGIN
DECLARE total_amount DECIMAL(10, 2);
SELECT SUM(Amount) INTO total_amount
FROM Transactions
WHERE User_ID = p_User_ID;
RETURN total_amount;
END $$
DELIMITER ;
Parameters in...
SQL functions
SQL keywords
Snippets
Query history
Query pro...
Bind para...
Filter: Regular expression
SHOW CREATE PROCEDURE `gaming_entertain`.`AddTransaction`;
CREATE FUNCTION GetUserTransactionTotal(p_User_ID INT) RETURNS DECIMAL(10, 2) DETERMINISTIC BEGIN DECLARE total_amount DECIMAL(10, 2); SELECT SUM(Amount) INTO total_amou
/* Affected rows: 0 Found rows: 0 Warnings: 0 Duration for 1 query: 0.016 sec. */
r15 : c1 (307 B) Connected: 00:40 h MySQL 8.0.30 Uptime: 00:41 h Server time: 11:22 Idle.
11:22
26/12/2024



1. View

Nama View: View_TransactionDetails

Kasus Penggunaan:

Admin ingin melihat laporan transaksi secara terperinci untuk menganalisis data penjualan, seperti siapa yang membeli game, game apa yang dibeli, dan jumlah pembayaran.

SELECT * FROM View_TransactionDetails;

2. Stored Procedure

Nama Procedure: AddTransaction

Kasus Penggunaan:

Saat pengguna membeli game, aplikasi secara otomatis memanggil prosedur ini untuk menyimpan data transaksi ke dalam database.

CALL AddTransaction(1, 3, '2024-12-26', 150000);

3. Function

Nama Function: GetUserTransactionTotal

Kasus Penggunaan:

Fitur di aplikasi memungkinkan pengguna melihat total pengeluaran mereka pada transaksi pembelian game

SELECT GetUserTransactionTotal(2) AS Total_Transactions;

4. Trigger (Tambahan Pilihan)

Nama Trigger: AfterInsertTransaction

Kasus Penggunaan:

Setiap kali transaksi baru ditambahkan, sistem otomatis mencatat log untuk audit atau laporan.

Berguna untuk audit keamanan atau debugging jika terjadi kesalahan pada transaksi.

5. Event (Tambahan Pilihan)

Nama Event: DeleteOldTransactions

Kasus Penggunaan:

Aplikasi secara otomatis menghapus transaksi lama (lebih dari satu tahun) untuk menghemat ruang penyimpanan dan menjaga kinerja database.