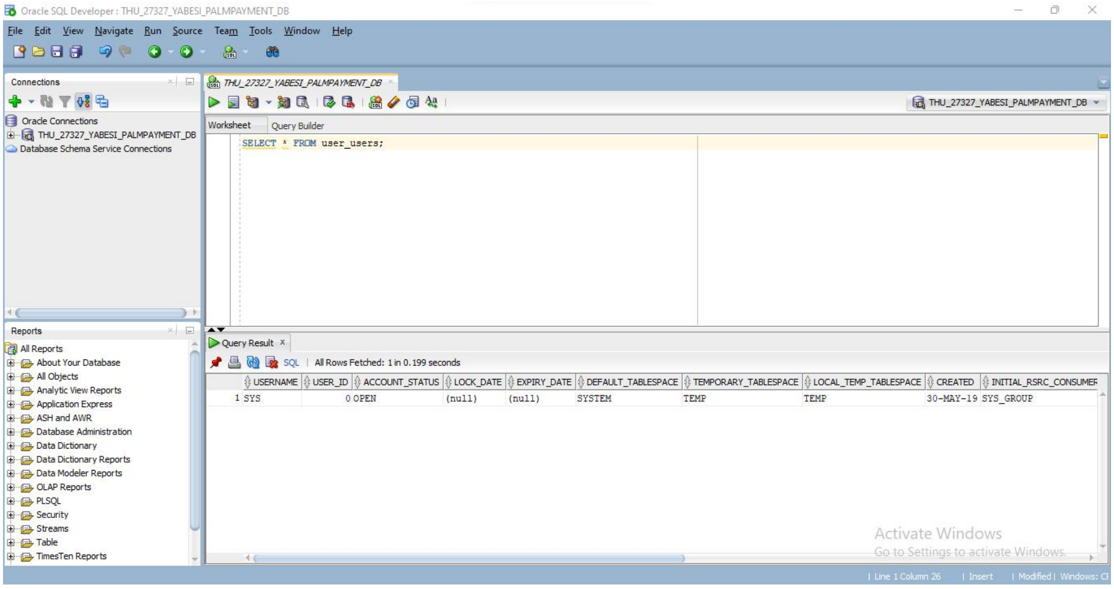
PHASE 5

```
----- creating tables and inserting data and Setting constraints such as NOT NULL,
UNIQUE, and CHECK to ensure data
validity and consistency.-------



CREATE TABLE Client (
   UserID INT PRIMARY KEY NOT NULL,
   Name VARCHAR2(100) NOT NULL CHECK (LENGTH(Name) > 2),
   Email VARCHAR2(100) UNIQUE NOT NULL,
   PhoneNumber VARCHAR2(15) UNIQUE NOT NULL CHECK
(REGEXP_LIKE(PhoneNumber, '^[0-9]+$')),
   BiometricData BLOB NOT NULL CHECK (LENGTH(BiometricData) > 0),
   AccountID INT NOT NULL
);

INSERT ALL
   INTO Client (UserID, Name, Email, PhoneNumber, BiometricData, AccountID)
   VALUES (1, 'IYUKURI YABESI', 'IYUKURI@example.com', '1234567890',
HEXTORAW('1F2E3D4C5B6A7980'), 1)

   INTO Client (UserID, Name, Email, PhoneNumber, BiometricData, AccountID)
   VALUES (2, 'RUKUNDO INNOCENT', 'RUKUNDO@example.com', '0987654321',
HEXTORAW('9A8B7C6D5E4F3210'), 2)

   INTO Client (UserID, Name, Email, PhoneNumber, BiometricData, AccountID)
   VALUES (3, 'Michael Johnson', 'michael@example.com', '1230984567',
HEXTORAW('ABCDEF1234567890'), 3)
SELECT * FROM DUAL;


CREATE TABLE Account (
   AccountID INT PRIMARY KEY NOT NULL,
   UserID INT NOT NULL,
   AccountNumber VARCHAR(20) UNIQUE NOT NULL CHECK
(LENGTH(AccountNumber) = 12),
   AccountBalance DECIMAL(10, 2) DEFAULT 0.00 CHECK (AccountBalance >= 0),
   DateCreated DATE NOT NULL,
   FOREIGN KEY (UserID) REFERENCES Client(UserID)
);

INSERT ALL
   INTO Account (AccountID, UserID, AccountNumber, AccountBalance, DateCreated)
   VALUES (1, 1, 'ACC123456789', 500.00, SYSDATE)

   INTO Account (AccountID, UserID, AccountNumber, AccountBalance, DateCreated)
   VALUES (2, 2, 'ACC098765432', 1500.00, SYSDATE)

   INTO Account (AccountID, UserID, AccountNumber, AccountBalance, DateCreated)
   VALUES (3, 3, 'ACC112233445', 2500.00, SYSDATE)
SELECT * FROM DUAL;

CREATE TABLE Merchant (
   MerchantID INT PRIMARY KEY NOT NULL,
   MerchantName VARCHAR(100) NOT NULL CHECK (LENGTH(MerchantName) >= 3),
   Location VARCHAR(255) NOT NULL CHECK (LENGTH(Location) > 3)
);

INSERT INTO Merchant (MerchantID, MerchantName, Location) VALUES (1,
'ElectroMart', 'Kigali');
INSERT INTO Merchant (MerchantID, MerchantName, Location) VALUES (2,
'FoodZone', 'Butare');
INSERT INTO Merchant (MerchantID, MerchantName, Location) VALUES (3,
'BookHub', 'Kigali');

CREATE TABLE Transaction (
   TransactionID INT PRIMARY KEY NOT NULL,
   AccountID INT NOT NULL,
   MerchantID INT NOT NULL,
   TransactionAmount DECIMAL(10, 2) NOT NULL CHECK (TransactionAmount > 0),
   TransactionDate DATE NOT NULL,
   FOREIGN KEY (AccountID) REFERENCES Account(AccountID),
   FOREIGN KEY (MerchantID) REFERENCES Merchant(MerchantID)
);


INSERT INTO Transaction (TransactionID, AccountID, MerchantID,
TransactionAmount, TransactionDate)
VALUES (1, 1, 1, 200.00, SYSDATE);

INSERT INTO Transaction (TransactionID, AccountID, MerchantID,
TransactionAmount, TransactionDate)
VALUES (2, 2, 2, 150.00, SYSDATE);

INSERT INTO Transaction (TransactionID, AccountID, MerchantID,
TransactionAmount, TransactionDate)
VALUES (3, 3, 3, 300.00, SYSDATE);
```

```
CREATE TABLE Payment_Gateway (
   PaymentID INT PRIMARY KEY NOT NULL,
   UserID INT NOT NULL,
   PaymentMethod VARCHAR(50) NOT NULL CHECK (PaymentMethod IN ('Credit
Card', 'Cash', 'Mobile Payment')),
   PaymentAmount DECIMAL(10, 2) NOT NULL CHECK (PaymentAmount > 0),
   PaymentDate DATE NOT NULL,
   FOREIGN KEY (UserID) REFERENCES Client(UserID)
);



INSERT INTO Payment_Gateway (PaymentID, UserID, PaymentMethod,
PaymentAmount, PaymentDate)
VALUES (1, 1, 'Credit Card', 250.00, SYSDATE);

INSERT INTO Payment_Gateway (PaymentID, UserID, PaymentMethod,
PaymentAmount, PaymentDate)
VALUES (2, 2, 'Cash', 100.00, SYSDATE);

INSERT INTO Payment_Gateway (PaymentID, UserID, PaymentMethod,
PaymentAmount, PaymentDate)
VALUES (3, 3, 'Mobile Payment', 300.00, SYSDATE);



CREATE TABLE Scanner (
   ScannerID INT PRIMARY KEY NOT NULL,
   Location VARCHAR(100) NOT NULL,
   ScannerType VARCHAR(50) NOT NULL CHECK (ScannerType IN ('Entry', 'Exit'))
);


INSERT INTO Scanner (ScannerID, Location, ScannerType) VALUES (1, 'Main Gate',
'Entry');

INSERT INTO Scanner (ScannerID, Location, ScannerType) VALUES (2, 'Exit Gate',
'Exit');
```

EMPTY TABLE

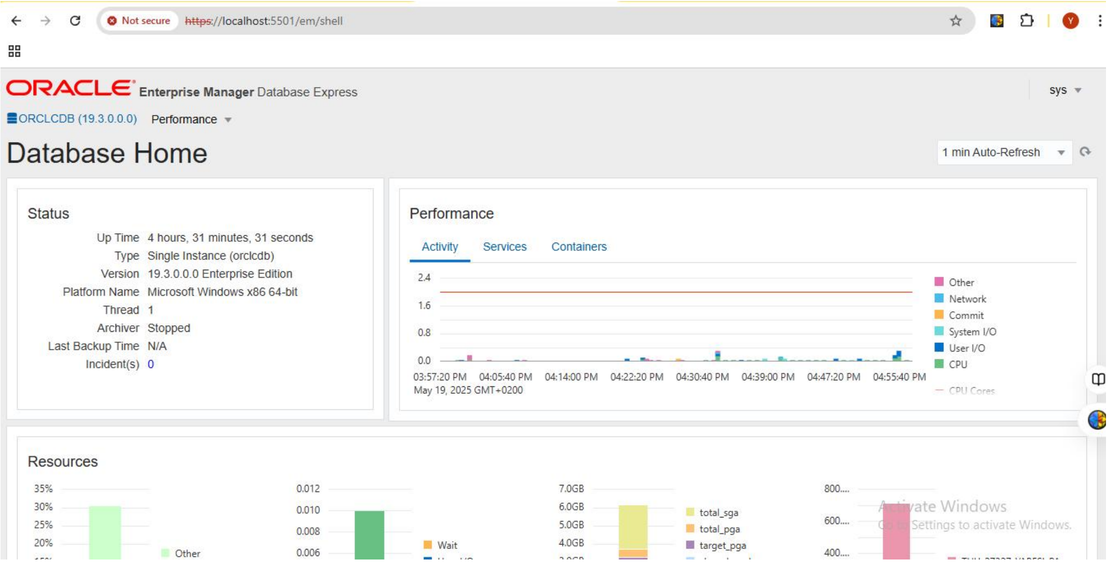


```
SELECT * FROM user_users;
```

All Rows Fetched: 1 in 0.199 seconds

	USERNAME	USER_ID	ACCOUNT_STATUS	LOCK_DATE	EXPIRY_DATE	DEFAULT_TABLESPACE	TEMPORARY_TABLESPACE	LOCAL_TEMP_TABLESPACE	CREATED	INITIAL_RSRC_CONSUMER
1	SYS	0	OPEN	(null)	(null)	SYSTEM	TEMP	TEMP	30-MAY-19	SYS_GROUP

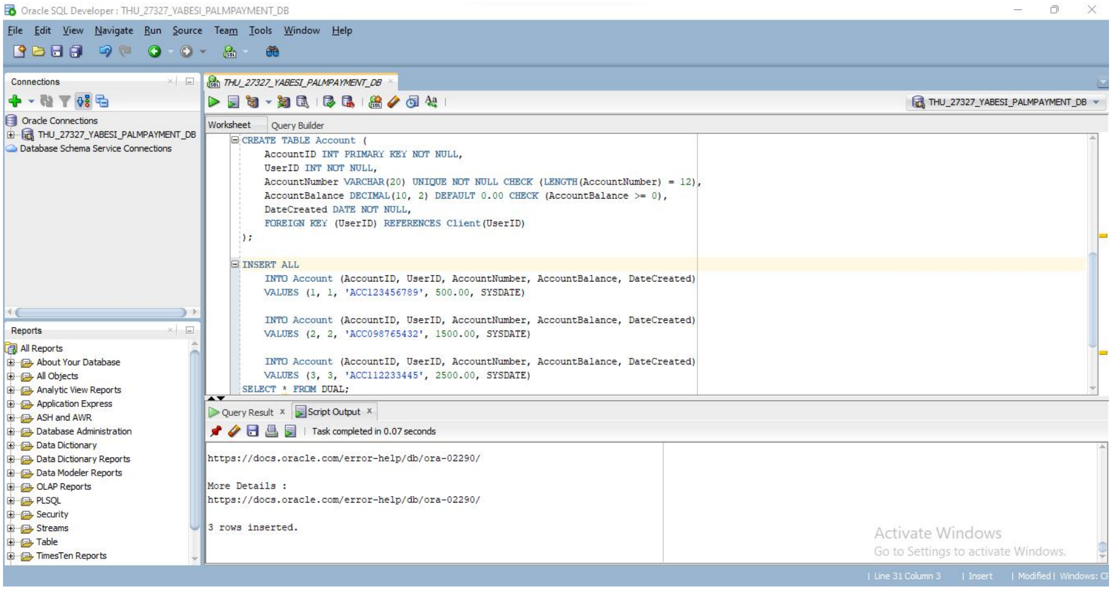
<u>USER SYS CONNECTED TO SYS</u>



Status

Up Time 4 hours, 31 minutes, 31 seconds
Type Single Instance (orclcdb)
Version 19.3.0.0.0 Enterprise Edition
Platform Name Microsoft Windows x86 64-bit
Thread 1
Archiver Stopped
Last Backup Time N/A
Incident(s) 0

DATA BEING INSERTED INTO THE TABLES



```
CREATE TABLE Account (
    AccountID INT PRIMARY KEY NOT NULL,
    UserID INT NOT NULL,
    AccountNumber VARCHAR(20) UNIQUE NOT NULL CHECK (LENGTH(AccountNumber) = 12),
    AccountBalance DECIMAL(10, 2) DEFAULT 0.00 CHECK (AccountBalance >= 0),
    DateCreated DATE NOT NULL,
    FOREIGN KEY (UserID) REFERENCES Client(UserID)
);

INSERT ALL
    INTO Account (AccountID, UserID, AccountNumber, AccountBalance, DateCreated)
    VALUES (1, 1, 'ACC123456789', 500.00, SYSDATE)

    INTO Account (AccountID, UserID, AccountNumber, AccountBalance, DateCreated)
    VALUES (2, 2, 'ACC098765432', 1500.00, SYSDATE)

    INTO Account (AccountID, UserID, AccountNumber, AccountBalance, DateCreated)
    VALUES (3, 3, 'ACC112233445', 2500.00, SYSDATE)
SELECT * FROM DUAL;
```

Task completed in 0.07 seconds

```
https://docs.oracle.com/error-help/db/ora-02290/

More Details :
https://docs.oracle.com/error-help/db/ora-02290/

3 rows inserted.
```