1: You are managing a database system for a transportation and supply chain company that handles tracking shipments, managing warehouse data, and processing client orders. However, the system's performance has declined due to the increasing data size, resulting in delays in shipment tracking and processing. Moreover, transaction errors during peak demand have occasionally led to incorrect updates in warehouse and shipment records.

Analyze the given scenario and complete the following tasks:

Warehouses Table

WarehouseID	Location	Capacity
W001	Houston	1200
W002	Atlanta	800
W003	Seattle	600

Shipments Table

ShipmentID	WarehouseID	ClientID	Date	Quantity
SHP001	W001	CLT001	2024-01-10	75
SHP002	W002	CLT002	2024-01-12	40
SHP003	W003	CLT003	2024-01-15	25

Tasks:

1. **Stored Procedure Design:** Write a stored procedure to calculate the total shipments dispatched from each warehouse within a given date range.
2. **Client-Specific Shipments:** Create a stored procedure to retrieve all shipment records for a particular client (e.g., a given ClientID).
3. **Monthly Shipments View:** Develop a view to display all shipments within a specific month, grouped by warehouse location.
4. **Warehouse Comparison Analysis:** Analyze the data to compare total shipment quantities between two selected warehouse locations (e.g., Houston and Atlanta) and identify which warehouse has dispatched a greater quantity.

2: You are responsible for managing and analyzing the database of a university's academic management system. This database stores information about students, courses, and their enrollments. Recently, the system has faced challenges with data inconsistency and redundancy, leading to issues such as incorrect course assignments, duplicate student records, and difficulty maintaining accurate data. Your task is to address these issues and enhance the system's integrity and reliability.

Tasks:

1. **Create Student Table:** Design a table to store student information, ensuring the `EnrollmentYear` field is mandatory. Prevent duplication of student records.
2. **Create Courses Table:** Develop a table for course information, ensuring the `Department` field is mandatory. Maintain referential integrity by linking the `Department` field in this table to the corresponding departments in the `Students` table.
3. **Analyze Tables and Constraints:** Evaluate the tables and apply appropriate constraints to maintain data consistency and integrity.
4. **Unique Constraint on Enrollments:** Add a unique constraint to the `Enrollments` table on the combination of `StudentID` and `CourseID` to ensure no student is enrolled in the same course more than once.

5. **Create Enrollments Table:** Create a table to store enrollment records, ensuring that `StudentID` and `CourseID` together form a unique key. Implement foreign key constraints linking this table to the `Students` and `Courses` tables.
6. **Query Enrollment Details:** Write a query to retrieve complete enrollment details, including student and course information.
7. **List Courses with High Enrollment:** Use a join to identify courses with the highest number of enrolled students.
8. **List Students by Course:** Create a query to list all students enrolled in a specific course (by `CourseID`), displaying the student's name, department, and course details.
9. **Display Students and Courses:** Write a query using an appropriate join to display all students and their respective courses by linking the `Students`, `Courses`, and `Enrollments` tables.
10. **Trigger for Overlapping Enrollments:** Develop a trigger that ensures a student cannot be enrolled in two courses that overlap on the same date.

3: You are responsible for managing the database of a healthcare facility that stores information about patient records, doctor assignments, and appointment scheduling. The hospital management faces challenges in generating accurate reports on patient visits, doctor workloads, and revenue analysis due to inefficient data querying. Additionally, the relationships between the tables are not well-defined, leading to errors in data retrieval and reporting. Your task is to improve the system by designing queries, triggers, and stored procedures to support effective aggregation, grouping, and accurate representation of relationships between patients, doctors, and appointments.

Appointments Table

AppointmentID	PatientID	DoctorID	Date	Fee
APPT001	PAT001	DOC001	2024-02-10	600
APPT002	PAT002	DOC002	2024-02-11	750
APPT003	PAT003	DOC003	2024-02-12	900

Patients Table

PatientID	Name	Age	Gender
PAT001	John	28	M
PAT002	Emma	34	F
PAT003	Michael	40	M

Doctors Table

DoctorID	Name	Speciality
DOC001	Dr. Williams	Pediatrics
DOC002	Dr. Brown	Orthopedics
DOC003	Dr. Taylor	Dermatology

Tasks:

1. **Calculate Total Revenue:** Write a query to calculate the total revenue collected from all appointments by summing the `Fee` column in the `Appointments` table.
2. **Doctor Appointment Count:** Join the `Appointments` table with the `Doctors` table to count the number of appointments handled by each doctor. Include the doctor's name and the total appointment count, grouped by doctor name.

3. **Top Patient by Appointments:** Identify the patient with the highest number of appointments by joining the `Appointments` and `Patients` tables. Display the patient's ID, name, and total number of appointments.
4. **Patient with Most and Least Appointments:** Write a query using `MAX()` and `MIN()` functions to find the patient with the highest and lowest number of appointments. Include the patient's name and appointment count in the result.

Additional Operations:

1. **Doctor Appointment Analysis:** Compare the tables by joining the `Appointments` and `Doctors` tables. Group the results by doctor name and count the number of appointments for each doctor.
2. **Trigger for Validation:** Design a trigger that checks the existence of the associated patient and doctor in their respective tables before allowing a new appointment entry.
3. **Stored Procedure for Doctor Revenue:** Create a stored procedure to calculate the total revenue generated by each doctor within a specific time frame, based on the `Fee` column in the `Appointments` table.

4:You are responsible for managing the inventory and sales database for a retail chain. The company is encountering difficulties in tracking product performance, analyzing customer purchase trends, and maintaining accurate stock levels. Additionally, repetitive tasks such as calculating total revenue and updating stock quantities after sales are time-consuming and prone to errors. Your objective is to optimize database operations, automate key tasks, and provide insightful data analysis to enhance efficiency and accuracy.

Products Table

ProductID	Name	Category	Price	Stock
PRD001	Smartphone	Electronics	900	15
PRD002	Sofa	Furniture	600	20
PRD003	Tablet	Electronics	300	30

Sales Table

SaleID	ProductID	Quantity	SaleDate
SAL001	PRD001	3	2025-01-10
SAL002	PRD002	2	2025-01-11
SAL003	PRD003	1	2025-01-12

Tasks:

1. **Analyze Tables:** Examine the `Products` and `Sales` tables to extract valuable insights, such as sales trends and stock levels.
2. **Perform Calculations:** Calculate metrics like total revenue generated from sales, remaining stock levels, and total units sold for each product.
3. **Add a New Category:** Introduce a new product category, "Gaming Accessories," and ensure it integrates seamlessly with the existing database structure.
4. **Retrieve Sales for Specific Product:** Write a query to retrieve all sales details for a specific product of your choice, including the product name and category.
5. **Identify Top-Selling Product:** Determine the product with the highest total sales (based on the quantity sold).

Additional Operations:

1. **Automate Database Tasks:** Implement mechanisms to automate routine operations, such as stock updates and revenue calculations.
2. **Update Stock Levels:** Simulate a scenario where the stock of all products is reduced by 5 units. Reflect these changes in the database.
3. **Create Procedures:** Develop stored procedures to handle critical tasks, such as updating stock after sales, calculating total sales revenue, and generating product performance reports.

5:You are managing the database system for a multi-purpose organization that handles various operations, including retail inventory, hospital management, academic records, and logistics. The organization is facing several challenges, such as maintaining data consistency, tracking trends, automating manual operations, and generating accurate reports. These issues include inefficient querying, data redundancy, and lack of automation for tasks such as revenue calculation, stock updates, and maintaining relationships between data entities.

You are tasked with designing and optimizing the database to address these challenges, ensuring accurate relationships between tables, efficient data analysis, and automated operations.

Products Table

ProductID	Name	Category	Price	Stock
P001	Laptop	Electronics	1500	20
P002	Sofa	Furniture	800	10
P003	Monitor	Electronics	300	15

Sales Table

SaleID	ProductID	Quantity	SaleDate
S001	P001	3	2025-01-15
S002	P002	1	2025-01-16
S003	P003	2	2025-01-17

Patients Table

PatientID	Name	Age	Gender
PAT001	John	30	M
PAT002	Emily	25	F
PAT003	Michael	40	M

Doctors Table

DoctorID	Name	Speciality
DOC001	Dr. Smith	Cardiology
DOC002	Dr. Brown	Neurology
DOC003	Dr. Taylor	Orthopedics

Appointments Table

AppointmentID	PatientID	DoctorID	Date	Fee
A001	PAT001	DOC001	2025-01-18	500
A002	PAT002	DOC002	2025-01-19	700
A003	PAT003	DOC003	2025-01-20	800

Tasks:

1. **Data Analysis:**
    - Analyze the tables to identify trends, such as the product with the highest sales or the doctor with the most appointments.
    - Identify the patient with the highest number of visits.
2. **Automation:**
    - Implement a trigger to ensure that stock levels are automatically reduced after a sale and validate that a valid `ProductID` exists before inserting a sales record.
    - Create a trigger to validate the existence of `PatientID` and `DoctorID` in their respective tables before adding a new appointment.
3. **Stored Procedures:**
    - Write a stored procedure to calculate total revenue collected by all appointments for a given date range.
    - Develop a procedure to calculate the total revenue generated from sales by category (e.g., Electronics, Furniture).
4. **Joins and Aggregation:**
    - Formulate a query to join the `Products` and `Sales` tables to calculate total revenue per product and total units sold.
    - Write a query to join the `Doctors` and `Appointments` tables to count the total appointments handled by each doctor.
    - Create a query to display the details of all patients who have been treated by a specific doctor.
5. **Constraints:**
    - Add a unique constraint to ensure that no patient can have duplicate appointments on the same date.
    - Add a unique constraint to the `Sales` table to ensure that no duplicate sales entries are created for the same product on the same date.
6. **Category Management:**
    - Add a new product category (e.g., "Gaming Accessories") and integrate it into the database schema.
    - Display all products in the new category with their stock levels and total sales.