

# Purpose and Goals

The database aims to streamline pharmacy operations by providing an integrated system for inventory tracking, supplier management, prescription records, and sales processing.

Automating invoicing and insurance claims will enhance accuracy, improve workflows, and ensure compliance, ultimately delivering better customer service and operational efficiency.

—  
**Provide an integrated system for inventory, supplies, and sales.**

—  
**Automate invoicing, billing, inventory updates and insurance claims.**

—  
**Maintain accurate records for compliance.**

—  
**Improve workflows for better service and efficiency.**

# Design Constraints

The database is designed for a small pharmacy with a single user. It must manage complex relationships, maintain data consistency, and prioritize reliability while being scalable to the pharmacy's needs.

—  
**Built for a single-user,  
small pharmacy.**

—  
**Ensure data consistency  
and reliability.**

—  
**Limited scalability for  
specific needs.**

—  
**Handle interactions  
between prescriptions,  
sales, and suppliers.**



# Database Design

# Assumptions

- There is only one type of prescription, with no variant categories.
- Insurance coverage is binary and uniform across all customers.
- All invoices are considered paid in full upon issuance, with no payment tracking or installments.
- Orders are always fulfilled completely by suppliers.

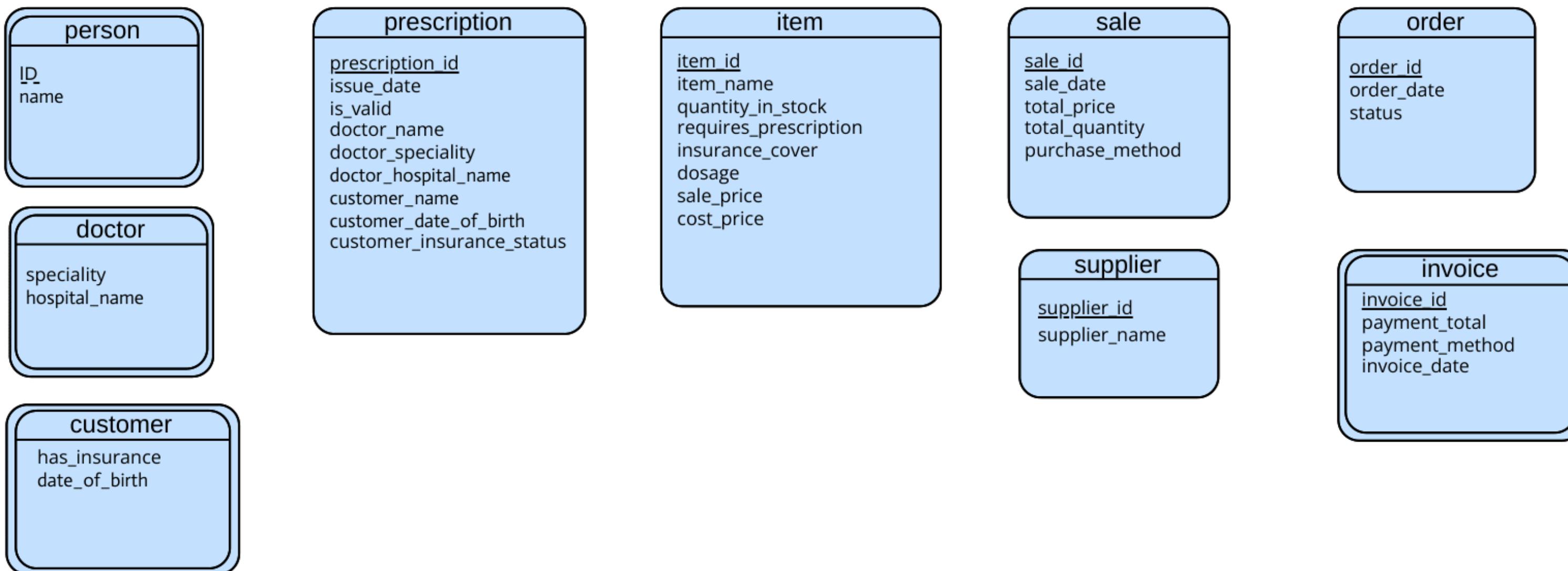
# Business Rules

Prescriptions are valid for four days from the issue date.

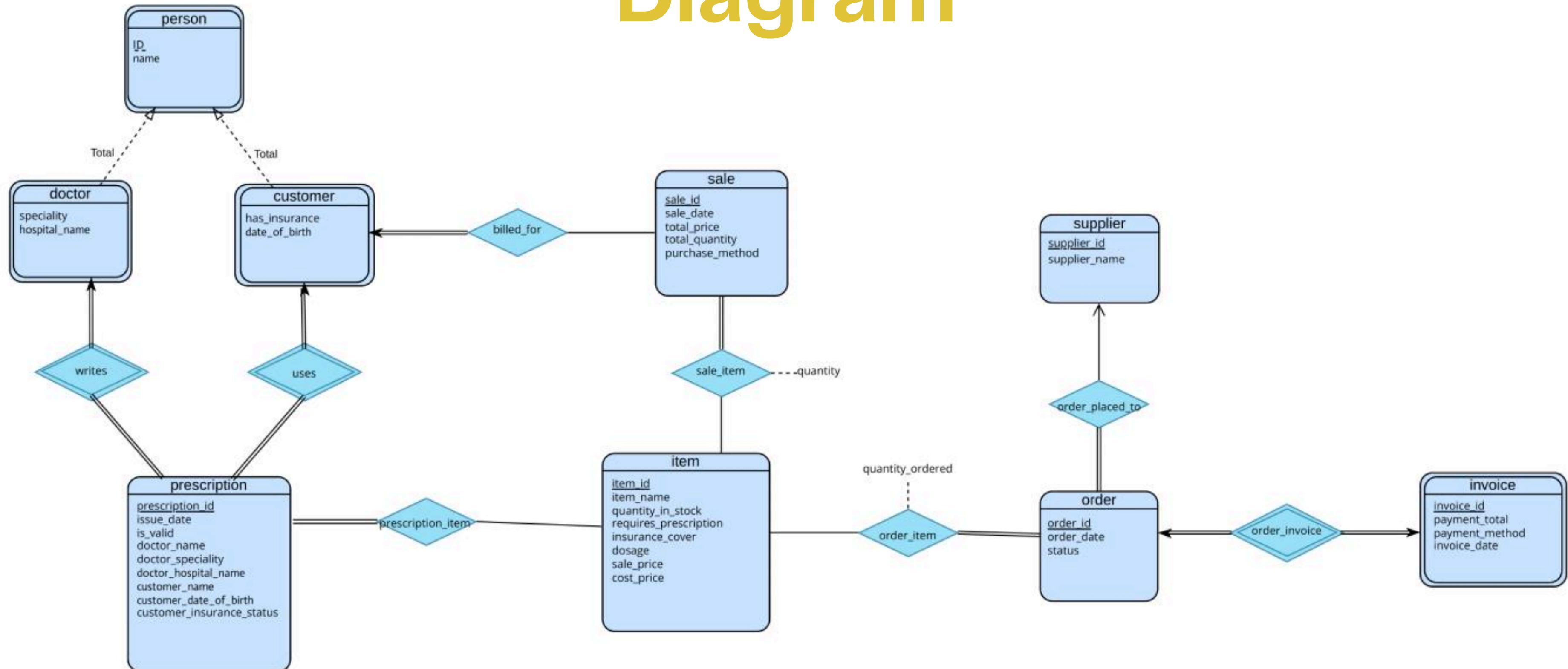
Prescriptions cannot be edited after being used for a sale.

All individuals (doctors, customers) are entered via prescription records.

# Entity Sets



# Entity-Relationship Diagram



# Relational Schema

- person (ID, name)
- doctor (ID, name speciality\_name, hospital\_name)
- customer (ID, name, date\_of\_birth)
- prescription (prescription\_id, issue\_date, doctor\_name, doctor\_speciality, doctor\_hospital\_name, customer\_name, customer\_date\_of\_birth customer\_insurance\_status)
- item (item\_id, item\_name, quantity\_in\_stock, requires\_prescription, insurance\_cover, dosage, sale\_price, cost\_price)
- sale (sale\_id, sale\_date, total\_price, total\_quantity, purchase\_method)
- supplier (supplier\_id, supplier\_name)
- order (order\_id, order\_date, status)
- invoice (invoice\_id, payment\_total, payment\_method, invoice\_date)
- doctor\_prescription (doctor\_id, prescription\_id)
- customer\_prescription (customer\_id, prescription\_id)
- item\_prescription (customer\_id, item\_id)
- customer\_sale (customer\_id, sale\_id)
- sale\_item (sale\_id, item\_id, quantity)
- supplier\_order (supplier\_id, order\_id)
- order\_item (order\_id, item\_id, quantity\_ordered)
- order\_invoice (order\_id, invoice\_id)

# Relational Schema

- person (ID, name)
- doctor (ID, name, speciality\_name, hospital\_name, **prescription\_id**)
- customer (ID, name, date\_of\_birth, **prescription\_id**)
- prescription (prescription\_id, issue\_date, doctor\_name, doctor\_speciality, doctor\_hospital\_name, customer\_name, customer\_date\_of\_birth, customer\_insurance\_status, **d\_ID**, **s\_ID**)
- item (item\_id, item\_name, quantity\_in\_stock, requires\_prescription, insurance\_cover, dosage, sale\_price, cost\_price)
- sale (sale\_id, sale\_date, total\_price, total\_quantity, purchase\_method, **c\_ID**)
- supplier (supplier\_id, supplier\_name)
- order (order\_id, order\_date, status, **supplier\_id**)
- invoice (invoice\_id, payment\_total, payment\_method, invoice\_date, **order\_ID**)
- ~~doctor\_prescription~~ (doctor\_id, prescription\_id)
- ~~customer\_prescription~~ (customer\_id, prescription\_id)
- item\_prescription (prescription\_id, item\_id)
- ~~customer\_sale~~ (customer\_id, sale\_id)
- sale\_item (sale\_id, item\_id, quantity)
- ~~supplier\_order~~ (supplier\_id, order\_id)
- order\_item (order\_id, item\_id, quantity\_ordered)
- ~~order\_invoice~~ (order\_id, invoice\_id)

# Triggers

- CREATE INVOICE AND UPDATE STOCK ON ORDER COMPLETION
- VALIDATIONS FOR ORDERS, SALES, PRESCRIPTIONS
- CREATE DOCTOR AND CUSTOMER FROM PRESCRIPTIONS
- CALCULATIONS FOR TOTALS

Script: **all code** [?](#) Status: **Complete** [?](#)View:  Detail  **Summary** [?](#)Rows **15**[Go](#)[Create App](#)[Edit Script](#)

<b>Number</b> ↑	<b>Elapsed</b>	<b>Statement</b>	<b>Feedback</b>	<b>Rows</b>
1	0.04	CREATE TABLE Person ( ID_ NUMBER PRIMARY KEY, Name_	Table created.	0
2	0.03	CREATE TABLE Doctor ( ID_ NUMBER PRIMARY KEY, Specia	Table created.	0
3	0.02	CREATE TABLE Customer ( ID_ NUMBER PRIMARY KEY, hasI	Table created.	0
4	0.02	CREATE TABLE Item ( itemID NUMBER PRIMARY KEY, itemN	Table created.	0
5	0.03	CREATE TABLE Prescription ( prescription_id NUMBER PRIMA	Table created.	0
6	0.02	CREATE TABLE Prescription_Item ( prescription_id NUMBER,	Table created.	0
7	0.02	CREATE TABLE Sale ( sale_id NUMBER PRIMARY KEY, sale	Table created.	0
8	0.02	CREATE TABLE Sale_Item ( sale_id NUMBER, itemID NUMB	Table created.	0
9	0.01	CREATE TABLE Supplier ( supplier_id NUMBER PRIMARY KEY,	Table created.	0
10	0.03	CREATE TABLE Orders ( order_id NUMBER PRIMARY KEY, o	Table created.	0
11	0.02	CREATE TABLE Order_Item ( order_id NUMBER, itemID NU	Table created.	0
12	0.03	CREATE TABLE Invoice ( invoice_id NUMBER PRIMARY KEY,	Table created.	0
13	0.04	ALTER TABLE Sale_Item ADD CONSTRAINT chk_positive_quantity_s	Table altered.	0
14	0.02	ALTER TABLE Order_Item ADD CONSTRAINT chk_positive_quantity_	Table altered.	0
15	0.03	CREATE OR REPLACE TRIGGER Reduce_Stock_On_Sale AFTER INSERT	Trigger created.	0

[Download](#)row(s) 1 - 15 of 310 [Next ▶](#)

310

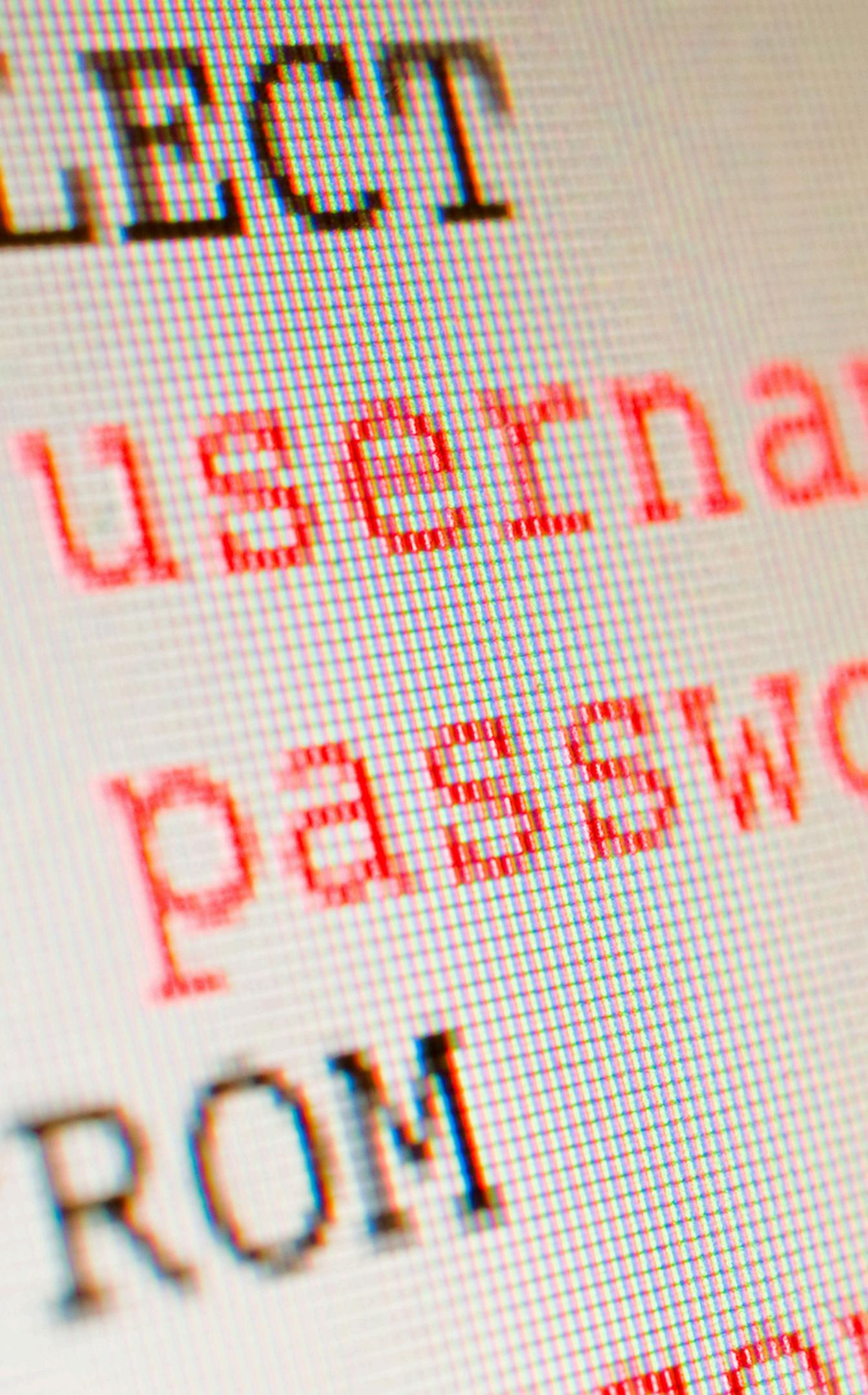
Statements Processed

281

Successful

29

With Errors



# **Implementation**

- Populated Tables**
- Example Queries**



+ Insert Row Columns... Filter... Count Rows Load Data Download Refresh

	ITEMID	ITEMNAME	QUANTITY_INSTOCK	REQUIRE_PRESCRIPTION	INSURANCE_COVER	DOSAGE	SALE_PRICE	COST_PRICE
	1	Paracetamol	542	Y	Y	500mg	5	2.5
	2	Ibuprofen	321	Y	N	200mg	8	4
	3	Cough Syrup	157	N	N	10ml	10	6
	4	Antibiotic A	195	Y	Y	250mg	20	10
	5	Antibiotic B	240	Y	N	500mg	25	12.5
	6	Vitamin C	995	N	N	500mg	3	1.5
	7	Pain Relief Gel	133	N	N	50g	15	8
	8	Allergy Medicine	427	N	Y	10mg	12	6
	9	Antiseptic Cream	203	N	N	25g	7	3.5
	10	Digestive Enzyme	296	N	N	100mg	9	4.5
	11	Multivitamin	494	N	N	1 Tablet	2	1
	12	Blood Pressure Medicine	332	Y	Y	50mg	30	15
	13	Cholesterol Medicine	406	Y	Y	20mg	28	14
	14	Diabetes Medicine	321	Y	Y	500mg	35	17.5
	15	Cold Medicine	611	N	N	1 Tablet	6	3
	16	Energy Drink	198	N	N	250ml	4	2
	17	Oral Rehydration Salt	96	N	N	1 Sachet	2.5	1.25
	18	Anti-inflammatory Ointm...	148	N	N	30g	10	5
	19	Anti-Allergy Drops	76	N	Y	5ml	18	9

[Columns](#) [Data](#) [Indexes](#) [Constraints](#) [Grants](#) [Statistics](#) [Triggers](#) [Dependencies](#) [DDL](#) [Sample Queries](#)[+ Insert Row](#)[Columns...](#)[Filter...](#)[Count Rows](#)[Load Data](#)[Download](#)[Refresh](#)

	ORDER_ID	ORDER_DATE	STATUS	SUPPLIER_ID
	1	12/14/2024	Done	1
	2	12/04/2024	Pending	2
	3	12/09/2024	Done	3
	4	11/29/2024	Pending	4
	5	12/19/2024	Done	5
	6	12/12/2024	Pending	6
	7	12/16/2024	Pending	7
	8	12/06/2024	Pending	8
	9	12/02/2024	Pending	9
	10	11/24/2024	Pending	10
	11	12/17/2024	Pending	11
	12	12/10/2024	Pending	12
	13	12/15/2024	Pending	13
	14	12/05/2024	Pending	14
	15	12/21/2024	Pending	15
	16	12/08/2024	Pending	16
	17	12/20/2024	Pending	17
	18	12/03/2024	Pending	18
	19	12/18/2024	Pending	19



## ORDER\_ITEM

[Columns](#) [Data](#) [Indexes](#) [Constraints](#) [Grants](#) [Statistics](#) [Triggers](#) [Dependencies](#) [DDL](#) [Sample Queries](#)[+ Insert Row](#)[Columns...](#)[Filter...](#)[Count Rows](#)[Load Data](#)[Download](#)[Refresh](#)

	ORDER_ID	ITEMID	QUANTITY
	1	1	50
	1	2	30
	1	3	20
	2	4	25
	2	5	15
	2	6	10
	3	7	20
	3	8	30
	3	9	25
	4	10	40
	4	11	50
	4	12	35
	5	13	20
	5	14	25
	5	15	30
	6	16	10
	6	17	15
	6	18	5
	7	19	20



## PRESCRIPTION\_ITEM

[Columns](#) [Data](#) [Indexes](#) [Constraints](#) [Grants](#) [Statistics](#) [Triggers](#) [Dependencies](#) [DDL](#) [Sample Queries](#)[+ Insert Row](#)[Columns...](#)[Filter...](#)[Count Rows](#)[Load Data](#)[Download](#)[Refresh](#)

	PREScription_ID	ITEMID	QUANTITY
	1	1	10
	1	6	5
	2	2	8
	2	3	4
	2	9	2
	3	4	7
	3	5	3
	5	11	12
	5	12	4
	6	13	5
	6	14	3
	7	15	9
	7	16	2
	8	17	4
	8	18	6
	10	1	5
	10	2	7
	11	3	6
	11	4	2

[Columns](#) [Data](#) [Indexes](#) [Constraints](#) [Grants](#) [Statistics](#) [Triggers](#) [Dependencies](#) [DDL](#) [Sample Queries](#)[+ Insert Row](#) [Columns...](#) [Filter...](#) [Count Rows](#) [Load Data](#) [Download](#) [Refresh](#)

	ID	HASINSURANCE	DATE_OF_BIRTH
	201	Y	03/25/1990
	202	N	07/12/1985
	203	Y	11/05/1975
	205	Y	01/30/1988
	206	Y	10/15/1980
	207	N	04/20/1995
	208	Y	06/11/1978
	210	Y	12/03/1991
	211	N	05/17/1984
	212	Y	02/22/1987
	213	Y	07/07/1993
	214	N	10/14/1979
	215	Y	03/01/1986
	216	Y	08/30/1994
	218	Y	12/15/1990
	219	N	05/20/1981
	220	Y	11/09/1989



	ID	SPECIALITY	HOSPITAL_NAME
	101	Pediatrics	City Hospital
	102	Dermatology	Health Clinic
	103	Cardiology	Heart Center
	105	Neurology	Brain Institute
	106	Oncology	Cancer Care
	107	Gastroenterology	St. Mary Hospital
	108	Endocrinology	Specialist Center
	110	Pulmonology	Breath Clinic
	111	Nephrology	Kidney Care
	112	Urology	Mens Health Center
	113	Hematology	Blood Institute
	114	Ophthalmology	Vision Clinic
	115	Radiology	Imaging Center
	116	Allergy and Immunology	Allergy Center
	118	Family Medicine	Family Health
	119	Emergency Medicine	ER Hospital
	120	Dermatology	Skin Clinic

[Columns](#) [Data](#) [Indexes](#) [Constraints](#) [Grants](#) [Statistics](#) [Triggers](#) [Dependencies](#) [DDL](#) [Sample Queries](#)[+ Insert Row](#)[Columns...](#)[Filter...](#)[Count Rows](#)[Load Data](#)[Download](#)[Refresh](#)

	ID	NAME
	201	Alice Brown
	101	Dr. Smith
	202	Bob Green
	102	Dr. Johnson
	203	Carol White
	103	Dr. Lee
	205	Eve Blue
	105	Dr. Garcia
	206	Frank Yellow
	106	Dr. Walker
	207	Grace Red
	107	Dr. Harris
	208	Henry Orange
	108	Dr. Adams
	210	Jack Purple
	110	Dr. Carter
	211	Kate Gray
	111	Dr. Collins
	212	Leo Brown



+ Insert Row

Columns...

Filter...

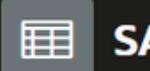
Count Rows

Load Data

Download

Refresh

	SALE_ID	SALE_DATE	TOTAL_PRICE	TOTAL_QUANTITY	PURCHASE_METHOD	CUSTOMER_ID
	1	12/19/2024	130	10	Cash	201
	2	12/18/2024	264	12	Card	202
	3	12/20/2024	53	5	Cash	203
	5	12/21/2024	100	6	Card	205
	6	12/22/2024	19	4	Online	206
	7	12/16/2024	92	7	Cash	207
	8	12/15/2024	220	11	Card	208
	10	12/23/2024	43	3	Cash	210
	11	12/19/2024	130	10	Cash	211
	12	12/18/2024	264	12	Card	212
	13	12/20/2024	53	5	Cash	213
	14	12/17/2024	110	8	Online	214
	15	12/21/2024	100	6	Card	215
	16	12/22/2024	19	4	Online	216
	18	12/15/2024	220	11	Card	218
	19	12/14/2024	176	9	Online	219
	20	12/23/2024	43	3	Cash	220
	21	12/22/2024	92	7	Online	
	22	12/16/2024	30	3	Cash	

[Columns](#) [Data](#) [Indexes](#) [Constraints](#) [Grants](#) [Statistics](#) [Triggers](#) [Dependencies](#) [DDL](#) [Sample Queries](#)[+ Insert Row](#) [Columns...](#) [Filter...](#) [Count Rows](#) [Load Data](#) [Download](#) [Refresh](#)

	SALE_ID	ITEMID	QUANTITY
	1	1	2
	1	12	3
	1	15	5
	2	2	3
	2	13	5
	2	5	4
	3	3	2
	3	6	1
	3	7	2
	5	10	2
	5	14	2
	5	15	2
	6	16	1
	6	17	2
	6	18	1
	7	11	3
	7	19	2
	7	20	2
	8	1	2

[Columns](#) [Data](#) [Indexes](#) [Constraints](#) [Grants](#) [Statistics](#) [Triggers](#) [Dependencies](#) [DDL](#) [Sample Queries](#)[+ Insert Row](#)[Columns...](#)[Filter...](#)[Count Rows](#)[Load Data](#)[Download](#)[Refresh](#)

	INVOICE_ID	PAYMENT_TOTAL	INVOICE_DATE	PAYMENT_METHOD	ORDER_ID
	1	365	12/24/2024	N/A	1
	2	427.5	12/24/2024	N/A	3
	3	807.5	12/24/2024	N/A	5

# Example Queries

- ALL THE PRESCRIPTIONS OF THE CUSTOMERS CONTAINING MEDICINES WRITTEN BY THE DOCTORS
- TRACKING THE MOST SELLING NON-PRESCRIBED ITEMS
- TRACKING REVENUE BY CUSTOMER
- TRACKING MOST ACTIVE SUPPLIERS

# All the Prescriptions of the Customers Containing Medicines Written by the Doctors

```
SELECT p.prescription_id, p.issue_date,
       (SELECT Name_ FROM Person WHERE ID_ = p.Doctor_ID) AS doctor_name,
       d.Speciality AS doctor_speciality,
       (SELECT Name_ FROM Person WHERE ID_ = p.Customer_ID) AS customer_name,
       (SELECT Date_of_Birth FROM Customer WHERE ID_ = p.Customer_ID) AS customer_dob,
       i.itemName AS medicine_name,
       pi.quantity
  FROM Prescription_Item pi
 JOIN Prescription p ON pi.prescription_id = p.prescription_id
 JOIN Doctor d ON p.Doctor_ID = d.ID_
 JOIN Item i ON pi.itemID = i.itemID
 ORDER BY p.issue_date DESC, customer_name;
```

PRESCRIPTION_ID	ISSUE_DATE	DOCTOR_NAME	DOCTOR_SPECIALITY	CUSTOMER_NAME	CUSTOMER_DOB	MEDICINE_NAME	QUANTITY
3	12/24/2024	Dr. Lee	Cardiology	Carol White	11/05/1975	Antibiotic B	3
3	12/24/2024	Dr. Lee	Cardiology	Carol White	11/05/1975	Antibiotic A	7
6	12/24/2024	Dr. Walker	Oncology	Frank Yellow	10/15/1980	Cholesterol Medicine	5
6	12/24/2024	Dr. Walker	Oncology	Frank Yellow	10/15/1980	Diabetes Medicine	3
10	12/24/2024	Dr. Carter	Pulmonology	Jack Purple	12/03/1991	Ibuprofen	7
10	12/24/2024	Dr. Carter	Pulmonology	Jack Purple	12/03/1991	Paracetamol	5
13	12/24/2024	Dr. Foster	Hematology	Mona White	07/07/1993	Pain Relief Gel	10
13	12/24/2024	Dr. Foster	Hematology	Mona White	07/07/1993	Allergy Medicine	3
16	12/24/2024	Dr. James	Allergy and Immunology	Paul Yellow	08/30/1994	Diabetes Medicine	5
16	12/24/2024	Dr. James	Allergy and Immunology	Paul Yellow	08/30/1994	Cholesterol Medicine	4
20	12/24/2024	Dr. Nelson	Dermatology	Tina Purple	11/09/1989	Paracetamol	4
20	12/24/2024	Dr. Nelson	Dermatology	Tina Purple	11/09/1989	Ibuprofen	6
1	12/23/2024	Dr. Smith	Pediatrics	Alice Brown	03/25/1990	Paracetamol	10
1	12/23/2024	Dr. Smith	Pediatrics	Alice Brown	03/25/1990	Vitamin C	5
5	12/23/2024	Dr. Garcia	Neurology	Eve Blue	01/30/1988	Multivitamin	12

# Tracking the most selling Non-Prescribed Items

```
SELECT i.itemName, SUM(si.quantity) AS total_quantity_sold
FROM Sale_Item si
JOIN Item i ON si.itemID = i.itemID
WHERE i.Require_Prescription = 'N'
GROUP BY i.itemName
ORDER BY total_quantity_sold DESC;
```

ITEMNAME	TOTAL QUANTITY SOLD
Cold Medicine	19
Cough Syrup	13
Pain Relief Gel	7
Multivitamin	6
Vitamin C	5
Anti-Allergy Drops	4
Oral Rehydration Salt	4
Pain Relief Spray	4
Digestive Enzyme	4
Allergy Medicine	3
Anti-inflammatory Ointment	2
Antiseptic Cream	2
Energy Drink	2

13 rows returned in 0.04 seconds [Download](#)

# Tracking Revenue by Customer

```
SELECT (SELECT Name_ FROM Person WHERE ID_ = s.Customer_ID_) AS customer_name,  
       SUM(si.quantity * i.Sale_Price) AS total_spent  
  FROM Sale_Item si  
 JOIN Sale s ON si.sale_id = s.sale_id  
 JOIN Item i ON si.itemID = i.itemID  
 GROUP BY s.Customer_ID_  
 ORDER BY total_spent DESC;
```

CUSTOMER_NAME	TOTAL_SPENT
Bob Green	264
Leo Brown	264
Henry Orange	220
Rose Orange	220
Sam Pink	176
-	170
Alice Brown	130
Kate Gray	130
Nate Black	110
Olive Blue	100
Eve Blue	100
Grace Red	92
Mona White	53
Carol White	53
Tina Purple	43

 yigitsariyar23@gmail.com  yigitsariyar  en Copyright © 1999, 2024, Oracle and/or its affiliates. Oracle APEX 24.1.5

# Tracking Most Active Suppliers

```
SELECT s.supplier_name, COUNT(o.order_id) AS total_orders, SUM(oi.quantity) AS total_items_supplied
FROM Order_Item oi
JOIN Orders o ON oi.order_id = o.order_id
JOIN Supplier s ON o.supplier_id = s.supplier_id
GROUP BY s.supplier_name
ORDER BY total_orders DESC, total_items_supplied DESC;
```

SUPPLIER_NAME	TOTAL_ORDERS	TOTAL_ITEMS_SUPPLIED
MediCore Inc.	3	125
PharmaPlus Ltd.	3	100
Health Pharma Co.	3	100
Elite Health Products	3	95
Advanced Health Solutions	3	90
SmartMed Distributors	3	85
Essential Drugs Inc.	3	85
Universal Pharmacy Supplies	3	80
VitalMed Supply Chain	3	75
Rapid Care Suppliers	3	75
Wellness Distributors	3	75
NextGen Medicals	3	65
QuickMed Corp.	3	65
Pure Pharma Co.	3	60
MedLife Group	3	60

**Thanks for your attention**