

## Module 10

### What to learn

#### CURSOR

- Declare cursor
- Fetch cursor
- Open cursor
- Deallocate cursor

### Practice Exercise

#### Practice 1

### Normalize the Employee and Department Data

You have been provided with the following unnormalized table:

EmployeeID	EmployeeName	Department	DepartmentLocation	Manager	Skills
1	John Doe	HR	New York	Jane Smith	Recruitment,Compliance
2	Alice Brown	IT	San Francisco	Bob Green	Networking,Security,Development
3	Mike Johnson	Finance	Chicago	Anna White	Accounting,Budgeting,Compliance

Normalize this table step-by-step up to 3NF (Third Normal Form). Explain how redundancy is reduced at each step and provide the normalized tables.

#### Practice 2

### Identify Functional Dependencies

Consider the following table:

OrderID	CustomerID	CustomerName	ProductID	ProductName	OrderDate	Amount
101	C001	John Smith	P101	Laptop	2023-10-01	1000
102	C002	Emma Brown	P102	Smartphone	2023-10-02	800
103	C001	John Smith	P103	Tablet	2023-10-03	600

Based on this table, identify all functional dependencies and explain their role in normalization.

#### Practice 3

### Denormalize a Normalized Database

You have been provided with the following normalized tables:

#### Products Table

ProductID	ProductName	Category
1	Laptop	Electronics
2	Phone	Electronics

#### Orders Table

OrderID	CustomerName	ProductID	Quantity
101	Alice	1	2
102	Bob	2	1

Denormalize these tables into a single table. Explain the trade-offs that come with denormalization, including data redundancy and potential anomalies.

#### Practice 4

### Normalization Case Study

Consider the following business need: A library system tracks books, authors, and members borrowing the books.

Books have attributes like BookID, Title, ISBN, AuthorName, and Publisher.

Authors have attributes like AuthorID, AuthorName, and Country.

Members who borrow books have attributes like MemberID, MemberName, Contact.

Design the normalized schema for this library system. Explain which normal form (1NF, 2NF, or 3NF) is achieved at each step and why.

#### Practice 5

### Spot Redundancy in an Unnormalized Table

Given the unnormalized table below:

FlightID	Airline	Source	Destination	Pilot	FlightSchedule	AircraftType
FL001	Airways1	New York	Chicago	John Doe	Monday, Wednesday	Boeing 737
FL002	Airways2	San Francisco	New York	Mike Johnson	Tuesday, Thursday	Airbus A320

Analyze the table and identify redundant data. Normalize the table while ensuring no data is lost.

#### Assignment Exercise

##### Assignment 1

Using cursor implement the following task employee Update the salary of the employee using following condition Salary between 30000 and 40000 – 5000 hike Salary between 40000 and 55000 – 7000 hike Salary between 55000 and 65000 – 9000 hike

##### Assignment 2

### Design a Normalized Database for an E-commerce Application

#### Scenario:

An online e-commerce store wants to manage its data using a normalized SQL database. The store sells multiple products that fall under various categories. Customers can place orders for multiple products in one transaction. You are required to design a database to manage the following functionalities:

Track product information such as ProductID, ProductName, Category, Price, and StockQuantity.

Store customer information including CustomerID, FullName, Email, Address, and ContactNumber.

Keep records of orders, where a customer can order multiple products (OrderID, CustomerID, OrderDate).

Track order details including OrderID, ProductID, Quantity, and TotalCost.

#### Requirements:

Start with an unnormalized table and show the steps to normalize it up to 3NF. Provide SQL scripts to create the normalized tables and add sample data.

Create a view to display all orders along with customer names, product names, and total costs.

Write SQL queries for the following:

List all products that are out of stock.

Find total revenue generated to date.

Identify customers who have placed more than 5 orders.

Find the most ordered product.

Explain the advantages of a normalized database in this scenario and any trade-offs involved.

#### Online Reference

<https://docs.microsoft.com/en-us/sql/t-sql/language-elements/cursors-transa...>

**Introduction to Relational Databases**

**Introduction to Select Statement**

**Filtering Results with WHERE Statements**

**Utilizing Joins**

**Executing Sub queries and Unions**

**Aggregating Data**

**Advanced Data Aggregations**

**Built in Functions**

**Query Optimization**

**Modifying Data**

**Advanced Data Modification**

**Stored Procedure**

**Transaction**

**Error handling**

**Designing Tables**

**triggers**