

```
--Step 1: Create the Database, if empty exists use it as here (was created before)
USE CybersecurityDB;
GO
```

```
--Step 2: Create Tables
-- Create Employees Table
```

```
CREATE TABLE Employees (
    EmployeeID INT PRIMARY KEY IDENTITY(1,1),
    FirstName NVARCHAR(50),
    LastName NVARCHAR(50),
    DepartmentID INT,
    Email NVARCHAR(100),
    IsActive BIT
);
GO
```

```
-- Create Departments Table
```

```
CREATE TABLE Departments (
    DepartmentID INT PRIMARY KEY IDENTITY(1,1),
    DepartmentName NVARCHAR(100)
);
GO
```

```
-- Create Logs Table (Security Logs for Cybersecurity)
```

```
CREATE TABLE Logs (
    LogID INT PRIMARY KEY IDENTITY(1,1),
    EmployeeID INT,
    LogTime DATETIME,
    Activity NVARCHAR(255),
    FOREIGN KEY (EmployeeID) REFERENCES Employees(EmployeeID)
);
GO
```

```
--Step 3: Insert Sample Data
```

```
-- Insert sample departments
```

```
INSERT INTO Departments (DepartmentName)
VALUES ('IT'), ('HR'), ('Finance');
GO
```

```
-- Insert sample employees
```

```
INSERT INTO Employees (FirstName, LastName, DepartmentID, Email, IsActive)
VALUES
('John', 'Doe', 1, 'johndoe@example.com', 1),
('Jane', 'Smith', 2, 'janesmith@example.com', 1),
('Sam', 'Brown', 3, 'sambrown@example.com', 0),
('Alice', 'Davis', 1, 'alicedavis@example.com', 1);
GO
```

```
-- Insert sample logs
```

```
INSERT INTO Logs (EmployeeID, LogTime, Activity)
```

```
VALUES
```

```
(1, '2025-03-10 08:30:00', 'Login from IP 192.168.1.1'),  
(2, '2025-03-10 09:00:00', 'Login from IP 192.168.1.2'),  
(1, '2025-03-11 10:15:00', 'File Accessed - sensitive data'),  
(3, '2025-03-12 12:00:00', 'Login from IP 192.168.1.3');
```

```
GO
```

```
SELECT * FROM Employees;
```

```
SELECT * FROM Departments;
```

```
SELECT * FROM Logs;
```

```
SELECT COUNT(*) FROM Employees;
```

```
SELECT COUNT(*) FROM Departments;
```

```
SELECT COUNT(*) FROM Logs;
```

```
--Step 4: Queries
```

```
--Now, let's create some SQL queries.
```

```
--1. Get all employees with their department names:
```

```
SELECT e.EmployeeID, e.FirstName, e.LastName, d.DepartmentName, e.Email, e.IsActive  
FROM Employees e
```

```
JOIN Departments d ON e.DepartmentID = d.DepartmentID;
```

```
GO
```

```
--2. Get employees with activity logs:
```

```
SELECT e.FirstName, e.LastName, l.Activity, l.LogTime
```

```
FROM Employees e
```

```
JOIN Logs l ON e.EmployeeID = l.EmployeeID
```

```
ORDER BY l.LogTime DESC;
```

```
GO
```

```
--3. Get all active employees:
```

```
SELECT e.FirstName, e.LastName, d.DepartmentName
```

```
FROM Employees e
```

```
JOIN Departments d ON e.DepartmentID = d.DepartmentID
```

```
WHERE e.IsActive = 1;
```

```
GO
```

```
--Step 5: Views
```

```
--Now let's create views for common queries.
```

```
--1. Active Employees View:
```

```
CREATE VIEW ActiveEmployees AS
```

```
SELECT e.EmployeeID, e.FirstName, e.LastName, d.DepartmentName
```

```
FROM Employees e
```

```
JOIN Departments d ON e.DepartmentID = d.DepartmentID
```

```
WHERE e.IsActive = 1;
```

```
GO
```

--2. Recent Logins View:

CREATE VIEW RecentLogins AS

SELECT e.FirstName, e.LastName, l.Activity, l.LogTime

FROM Employees e

JOIN Logs l ON e.EmployeeID = l.EmployeeID

WHERE l.Activity LIKE 'Login%';

GO

--Running above views

SELECT \* FROM ActiveEmployees

SELECT \* FROM RecentLogins

ORDER BY LogTime DESC;

--Step 6: Common Table Expressions (CTEs)

--CTEs are useful for complex queries. Let's create some CTE examples .

--1. CTE to Find Employees with Recent Activity (e.g., last 24 hours):

WITH RecentActivity AS (

SELECT e.EmployeeID, e.FirstName, e.LastName, l.Activity, l.LogTime

FROM Employees e

JOIN Logs l ON e.EmployeeID = l.EmployeeID

WHERE l.LogTime > DATEADD(DAY, -1, GETDATE()))

)

SELECT \* FROM RecentActivity;

GO

--2. CTE to Get Employees with Multiple Logins:

WITH MultipleLogins AS (

SELECT e.EmployeeID, e.FirstName, e.LastName, COUNT(l.LogID) AS LoginCount

FROM Employees e

JOIN Logs l ON e.EmployeeID = l.EmployeeID

WHERE l.Activity LIKE 'Login%'

GROUP BY e.EmployeeID, e.FirstName, e.LastName

HAVING COUNT(l.LogID) > 1

)

SELECT \* FROM MultipleLogins;

GO