create database EMP\_Project

use EMP\_Project

-- Departments Table

CREATE TABLE Departments (

DepartmentID INT PRIMARY KEY IDENTITY(1,1),

DepartmentName VARCHAR(100) NOT NULL

);

-- Roles Table

CREATE TABLE Roles (

RoleID INT PRIMARY KEY IDENTITY(1,1),

RoleName VARCHAR(100) NOT NULL

);

-- Employees Table

CREATE TABLE Employees (

EmployeeID INT PRIMARY KEY IDENTITY(1,1),

FirstName VARCHAR(50),

LastName VARCHAR(50),

PhoneNumber VARCHAR(20),

DepartmentID INT FOREIGN KEY REFERENCES Departments(DepartmentID),

RoleID INT FOREIGN KEY REFERENCES Roles(RoleID),

JoinDate DATE

);

-- Attendance

CREATE TABLE Attendance (

AttendanceID INT PRIMARY KEY IDENTITY(1,1),

EmployeeID INT FOREIGN KEY REFERENCES Employees(EmployeeID),

AttendanceDate DATE NOT NULL,

CheckInTime DATETIME,

CheckOutTime DATETIME,

Status VARCHAR(20), -- Present, Absent, Late

CreatedAt DATETIME DEFAULT GETDATE()

);

-- Insert Departmentname inorder to get the 200 records

INSERT INTO Departments (DepartmentName)

VALUES ('IT'), ('HR'), ('Finance'), ('Operations'), ('Marketing');

-- Insert Rolename inorder to get the 200 records

INSERT INTO Roles (RoleName)

VALUES ('Developer'), ('Analyst'), ('Manager'), ('Support'), ('Intern');

-- Monthly Attendance Summary per Employee

SELECT

e.EmployeeID,

e.FirstName,

e.LastName,

DATENAME(MONTH, a.AttendanceDate) AS MonthName,

COUNT(CASE WHEN a.Status = 'Present' THEN 1 END) AS DaysPresent,

COUNT(CASE WHEN a.Status = 'Absent' THEN 1 END) AS DaysAbsent,

COUNT(CASE WHEN a.Status = 'Late' THEN 1 END) AS DaysLate

FROM Employees e

JOIN Attendance a ON e.EmployeeID = a.EmployeeID

WHERE YEAR(a.AttendanceDate) = YEAR(GETDATE())

GROUP BY e.EmployeeID, e.FirstName, e.LastName, DATENAME(MONTH, a.AttendanceDate)

ORDER BY e.EmployeeID, MonthName;

-- List Late Arrivals

SELECT

e.EmployeeID,

e.FirstName,

e.LastName,

a.AttendanceDate,

a.CheckInTime

FROM Employees e

JOIN Attendance a ON e.EmployeeID = a.EmployeeID

WHERE a.Status = 'Late'

ORDER BY a.AttendanceDate DESC;

--Trigger for AttendanceStatus

CREATE TRIGGER trg\_AttendanceStatus

ON Attendance

AFTER INSERT

CREATE TRIGGER trg\_AttendanceStatus

ON Attendance

AFTER INSERT

AS

BEGIN

UPDATE A

SET

A.CreatedAt = GETDATE(),

A.Status = CASE

WHEN A.CheckInTime IS NULL THEN 'Absent'

WHEN CAST(A.CheckInTime AS TIME) > '09:30:00' THEN 'Late'

ELSE 'Present'

END

FROM Attendance A

INNER JOIN Inserted I ON A.AttendanceID = I.AttendanceID;

END;

GO

-- Employee Absent

INSERT INTO Attendance (AttendanceID, EmployeeID, AttendanceDate, CheckInTime, CheckOutTime, Status, CreatedAt)

VALUES (201, 5, '2025-09-05', NULL, NULL, 'Absent', '2025-09-05 18:30:00');

SELECT AttendanceID, EmployeeID, AttendanceDate, CheckInTime, CheckOutTime, Status, CreatedAt

FROM Attendance

WHERE AttendanceDate = '2025-09-05';

--Function for total Workhours

CREATE FUNCTION fn\_TotalWorkHours (@EmpID INT, @Month INT, @Year INT)

RETURNS INT

AS

BEGIN

DECLARE @TotalMinutes INT;

SELECT @TotalMinutes = ISNULL(SUM(DATEDIFF(MINUTE, CheckInTime, CheckOutTime)),0)

FROM Attendance

WHERE EmployeeID = @EmpID

AND MONTH(AttendanceDate) = @Month

AND YEAR(AttendanceDate) = @Year

AND Status IN ('Present','Late');

RETURN @TotalMinutes;

END;

--Fetching result

SELECT dbo.fn\_TotalWorkHours(1, MONTH(GETDATE()), YEAR(GETDATE())) AS TotalMinutesWorked;

-- Department-wise average attendance

SELECT

D.DepartmentName,

COUNT(A.AttendanceID) AS TotalMarked,

SUM(CASE WHEN A.Status = 'Present' THEN 1 ELSE 0 END) AS PresentCount,

SUM(CASE WHEN A.Status = 'Late' THEN 1 ELSE 0 END) AS LateCount,

SUM(CASE WHEN A.Status = 'Absent' THEN 1 ELSE 0 END) AS AbsentCount

FROM Departments D

JOIN Employees E ON D.DepartmentID = E.DepartmentID

JOIN Attendance A ON E.EmployeeID = A.EmployeeID

WHERE MONTH(A.AttendanceDate) = MONTH(GETDATE())

AND YEAR(A.AttendanceDate) = YEAR(GETDATE())

GROUP BY D.DepartmentName

HAVING SUM(CASE WHEN A.Status = 'Absent' THEN 1 ELSE 0 END) > 5

ORDER BY LateCount DESC;