

### 3. Database Task: Employee Management System

#### 1. Database Schema

##### Employee Table

The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows the SQL script for creating and populating the Employee table. The bottom pane shows the results of the query, which is the Employee table.

```
SQLQuery4.sql - P... (PARVATHY\HP (56)) * X PARVATHY\SQLSERVER... - dbo.Employees
select * from Employees

--insert into Employees (Employee_ID,Employee_NAME,Department_ID,Hire_DATE,IS_ACTIVE,CREATED_BY,CREATED_ON,UPDATED_BY,UPDATED_ON)
--values(5, 'parvathy', 2, GETDATE(), 1, 1, GETDATE(), 1, GETDATE())

--select * from Departments

--insert into Departments (Department_ID,Department_NAME,IS_ACTIVE,CREATED_BY,CREATED_ON,UPDATED_BY,UPDATED_ON)
--values(3,'intern', 1,1, GETDATE(),1, GETDATE())

--select * from Salaries

--insert into Salaries(Employee_ID,BaseSalary,Bonus,Deductions,IS_ACTIVE,CREATED_BY,CREATED_ON,UPDATED_BY,UPDATED_ON)
--values(4,45000,2000,3000,1,1,GETDATE(),1,GETDATE())

--select * from SalaryHistory

--INSERT INTO SalaryHistory (History_ID, Employee_ID, Old_BaseSalary, New_BaseSalary, Old_Bonus, New_Bonus, Old_Deduction, New_Deduction ,IS_ACTIVE,CREATED_BY,CREATED_ON,UPDATED_BY,UPDATED_ON)
```

Employee_ID	Employee_NAME	Department_ID	Hire_DATE	IS_ACTIVE	CREATED_BY	CREATED_ON	UPDATED_BY	UPDATED_ON
1	Arjun	1	2025-02-10	1	1	2025-02-10 14:48:17.597	1	2025-02-10 14:48:17.597
2	anu	1	2025-02-10	1	1	2025-02-10 14:50:20.493	1	2025-02-10 14:50:20.493
3	anu	1	2025-02-10	3	1	2025-02-10 14:52:16.110	1	2025-02-10 14:52:16.110
4	anupa	3	2025-02-10	3	1	2025-02-10 14:52:50.657	1	2025-02-10 14:52:50.657
5	parvathy	2	2025-02-10	1	1	2025-02-10 15:00:06.863	1	2025-02-10 15:00:06.863
6	neethu	2	2025-10-01	1	1	2025-02-10 16:54:41.587	1	2025-02-10 16:54:41.587
7	Ryan	2	2025-10-01	1	1	2025-02-10 17:04:17.300	1	2025-02-10 17:04:17.300
8	karthika	1	2024-11-01	1	1	2025-02-10 17:38:53.380	NULL	NULL

##### Department Table

The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows the SQL script for creating and populating the Department table. The bottom pane shows the results of the query, which is the Department table.

```
SQLQuery4.sql - P... (PARVATHY\HP (56)) * X PARVATHY\SQLSERVER... - dbo.Employees
--select * from Employees

--insert into Employees (Employee_ID,Employee_NAME,Department_ID,Hire_DATE,IS_ACTIVE,CREATED_BY,CREATED_ON,UPDATED_BY,UPDATED_ON)
--values(5, 'parvathy', 2, GETDATE(), 1, 1, GETDATE(), 1, GETDATE())

select * from Departments

--insert into Departments (Department_ID,Department_NAME,IS_ACTIVE,CREATED_BY,CREATED_ON,UPDATED_BY,UPDATED_ON)
--values(3,'intern', 1,1, GETDATE(),1, GETDATE())

--select * from Salaries

--insert into Salaries(Employee_ID,BaseSalary,Bonus,Deductions,IS_ACTIVE,CREATED_BY,CREATED_ON,UPDATED_BY,UPDATED_ON)
--values(4,45000,2000,3000,1,1,GETDATE(),1,GETDATE())

--select * from SalaryHistory

--INSERT INTO SalaryHistory (History_ID, Employee_ID, Old_BaseSalary, New_BaseSalary, Old_Bonus, New_Bonus, Old_Deduction, New_Deduction ,IS_ACTIVE,CREATED_BY,CREATED_ON,UPDATED_BY,UPDATED_ON)
```

Department_ID	Department_NAME	IS_ACTIVE	CREATED_BY	CREATED_ON	UPDATED_BY	UPDATED_ON
1	manager	1	1	2025-02-10 14:48:10.807	1	2025-02-10 14:48:10.807
2	developer	1	1	2025-02-10 14:50:15.897	1	2025-02-10 14:50:15.897
3	intern	1	1	2025-02-10 14:51:56.597	1	2025-02-10 14:51:56.597

##### Salary Table

```
--VALUES(3, INTERM , 1,1, GETDATE(),1, GETDATE())

select * from Salaries

--insert into Salaries(Employee_ID,BaseSalary,Bonus,Deductions,IS_ACTIVE,CREATED_BY,CREATED_ON,UPDATED_BY,UPDATED_ON)
--values(4,45000,2000,3000,1,1,GETDATE(),1,GETDATE())

--select * from SalaryHistory

--INSERT INTO SalaryHistory (History_ID, Employee_ID, Old_BaseSalary, New_BaseSalary, Old_Bonus, New_Bonus, Old_Deduction, New_Deduction, IS_ACTIVE, CREATED_BY, CREATED_ON, UPDATED_BY, UPDATED_ON)
VALUES (1, 4, 15000, 20000, 2000, 2500, 1000, 500, 1, 1, GETDATE(), 1, GETDATE());
```

Employee_ID	BaseSalary	Bonus	Deductions	IS_ACTIVE	CREATED_BY	CREATED_ON	UPDATED_BY	UPDATED_ON
1	15000	2000	1000	1	1	2025-02-10 14:57:44.620	1	2025-02-10 14:57:44.620
4	6000	2500	500	1	1	2025-02-10 14:58:33.997	1	2025-02-10 14:58:33.997

## SalaryHistory Table

```
select * from SalaryHistory

--INSERT INTO SalaryHistory (History_ID, Employee_ID, Old_BaseSalary, New_BaseSalary, Old_Bonus, New_Bonus, Old_Deduction, New_Deduction ,IS_ACTIVE,CREATED_BY,CREATED_ON,UPDATED_BY,UPDATED_ON)
VALUES (1, 4, 15000, 20000, 2000, 2500, 1000, 500, 1, 1, GETDATE(), 1, GETDATE());
```

History_ID	Employee_ID	Old_BaseSalary	Old_Bonus	Old_Deduction	New_BaseSalary	New_Bonus	New_Deduction	Changed_Date	IS_ACTIVE	CREATED_BY	CREATED_ON	UPDATED_BY	UPDATED_ON
1	4	15000	2000	1000	20000	2500	500	2025-02-10	1	1	2025-02-10 21:31:33.457	1	2025-02-10 21:31:33.457
2	4	45000	2000	3000	5222222	2500	255	2025-02-11	1	1	2025-02-11 10:39:33.837	1	2025-02-11 10:39:33.837
3	4	5222222	2500	255	60000	2500	500	2025-02-11	1	1	2025-02-11 10:42:50.783	1	2025-02-11 10:42:50.783
4	4	60000	2500	500	6000	2500	500	2025-02-11	1	1	2025-02-11 10:46:27.150	1	2025-02-11 10:46:27.150

## SQL Queries:

- List all employees along with their department names

```
SQLQuery4.sql - P:\PARVATHY\HP (56)\* - X PARVATHY\SQL\EXPL... dpo.employees

-- VALUES (1, 4, 15000, 20000, 2000, 2500, 1000, 500,1,1,GETDATE(), 1, GETDATE(),GETDATE());

--SQL QUERIES

select E.Employee_ID ,E.Employee_NAME ,D.Department_NAME from Employees E
join Departments D on E.Department_ID = D.Department_ID

--select E.Employee_ID,E.Employee_NAME,S.BaseSalary,S.Bonus,S.Deductions ,(S.BaseSalary+S.Bonus-S.Deductions) as Netsalary from Employees E
--join Salaries S on E.Employee_ID=S.Employee_ID

--select TOP 1 D.Department_ID, D.Department_NAME, max(S.BaseSalary+S.Bonus-S.Deductions) as AvgSalary from Employees E
--join Salaries S on E.Employee_ID = S.Employee_ID
--join Departments D on E.Department_ID = D.Department_ID
--group by D.Department_ID,D.Department_NAME
--order by AvgSalary desc
```

Employee_ID	Employee_NAME	Department_NAME
1	Ajun	manager
2	anu	manager
3	anu	manager
4	anupa	intern
5	parvathy	developer
6	neethu	developer
7	Ryan	developer
8	karihika	manager

- Calculate the net salary for each employee using: Net Salary = BaseSalary + Bonus - Deductions.

```
--join Departments D on E.Department_ID = D.Department_ID

select E.Employee_ID,E.Employee_NAME,S.BaseSalary,S.Bonus,S.Deductions ,(S.BaseSalary+S.Bonus-S.Deductions) as Netsalary from Employees E
join Salaries S on E.Employee_ID=S.Employee_ID

--select TOP 1 D.Department_ID, D.Department_NAME, max(S.BaseSalary+S.Bonus-S.Deductions) as Avgsalary from Employees E
--join Salaries S on E.Employee_ID = S.Employee_ID
--join Departments D on E.Department_ID = D.Department_ID
--group by D.Department_ID,D.Department_NAME
--order by Avgsalary desc
```

Employee_ID	Employee_NAME	BaseSalary	Bonus	Deductions	Netsalary
1	Arun	15000	2000	1000	16000
4	anupa	6000	2500	500	8000

- Identify the department with the highest average salary.

```
select TOP 1 D.Department_ID, D.Department_NAME, max(S.BaseSalary+S.Bonus-S.Deductions) as Avgsalary from Employees E
join Salaries S on E.Employee_ID = S.Employee_ID
join Departments D on E.Department_ID = D.Department_ID
group by D.Department_ID,D.Department_NAME
order by Avgsalary desc
```

Department_ID	Department_NAME	Avgsalary
1	manager	16000

## Views:

**EmployeeSalaryView:** A view that combines Employees, Departments, and Salaries to provide a detailed report of employee salaries with department names and net salaries.

PARVATHY\SQLXPRTerm - dbo.View\_1

Column	Alias	Table	Output	Sort Type	Sort Order	Filter	Or...	Or...	Or...
Employee_ID		E	<input checked="" type="checkbox"/>						
Employee_NAME		E	<input checked="" type="checkbox"/>						
Department_NAME		D	<input checked="" type="checkbox"/>						
BaseSalary		S	<input checked="" type="checkbox"/>						
Bonus		S	<input checked="" type="checkbox"/>						
Deductions		S	<input checked="" type="checkbox"/>						

```
SELECT E.Employee_ID, E.Employee_NAME, D.Department_NAME, S.BaseSalary, S.Bonus, S.Deductions, S.BaseSalary + S.Bonus - S.Deductions AS NETSALARY
FROM dbo.Employees AS E INNER JOIN
      dbo.Departments AS D ON E.Department_ID = D.Department_ID INNER JOIN
      dbo.Salaries AS S ON E.Employee_ID = S.Employee_ID
```

Employee_ID	Employee_NAME	Department_NAME	BaseSalary	Bonus	Deductions	NETSALARY
1	Arun	manager	15000	2000	1000	16000
4	anupa	intern	6000	2500	500	8000

of 2