LIMKOKWING
UNIVERSITY
OF CREATIVE TECHNOLOGY

Faculty of Information & Communication Technology

Software Engineering with Multimedia

Semester: 3

Database Systems Final Assignment

Lecturer: Leng Rathana

Prepare by

Name	: Chy Sovan	Name	: Doung Kaneka	Name	: Lee Sothirakyoth
ID	: 601005827	ID	: 601005826	ID	: 601005764

19th October 2023

Payroll Database system - Table of contents

I. Introduction

1. Introduction to Business Entity

When setting up a payroll system, it is crucial to understand the concept of a business entity and its implications. A business entity refers to the legal structure or form that the organization takes, which significantly impacts the design, compliance, and tax obligations of the payroll system. The choice of business entity is a fundamental decision that affects not only the overall structure and operations of the organization but also the payroll processes. Different business entity types, such as sole proprietorships, partnerships, limited liability companies (LLCs), corporations, and nonprofit organizations, have unique characteristics, legal requirements, and tax obligations. These factors directly influence the design and implementation of the payroll system, including employee classification, taxation, liability considerations, and reporting requirements. Understanding the implications of the chosen business entity on the payroll system is essential to ensure accurate and compliant payroll processes, ultimately contributing to the overall success and financial stability of the organization.

2. Overview of Database System

A database system plays a crucial role in a payroll system by efficiently storing and managing employee-related information and financial data. It serves as a central repository for all payroll-related data, including employee records, salary details, tax deductions, and payment history. The database system organizes data into structured tables, allowing for easy retrieval and manipulation of information and it ensures data integrity through mechanisms such as data validation, constraints, and referential integrity. By leveraging a database system, a payroll system can automate various tasks, such as calculating salaries, generating pay stubs, and producing tax reports. Moreover, it provides a secure and scalable platform for handling large volumes of data, ensuring that sensitive information is protected. With its ability to streamline payroll processes and enhance data accuracy, a well-designed database system is essential for a robust and efficient payroll system.

Here are some details about the database system in a payroll system:

Data Organization and Accessibility: The database system organizes payroll-related data into structured tables with each table representing a specific aspect of payroll management such as employee information, time and attendance records, deductions, and benefits. This organization enables efficient storage and retrieval of data to make it easier to access and update employee records as needed.

Data Integration and Consolidation: The payroll system often needs to integrate data from various sources, such as HR systems, time-tracking software, and tax authorities. A database system allows for seamless integration and consolidation of data from multiple sources, ensuring that all necessary information is available in one centralized location.

Data Security and Privacy: Payroll data contains sensitive information, including employee Social Security numbers, salary details, and tax information. A robust database system incorporates security measures such as access controls and encryption to protect the data. It helps ensure compliance with data protection regulations such as the General Data Protection Regulation (GDPR) or the Health Insurance Portability and Accountability Act (HIPAA).

Data Accuracy and Consistency: Accuracy is crucial in payroll processing. A database system enforces data validation rules and constraints, preventing the entry of erroneous or inconsistent data. It helps maintain data integrity and ensures that calculations, such as salary computations and tax deductions, are accurate and reliable.

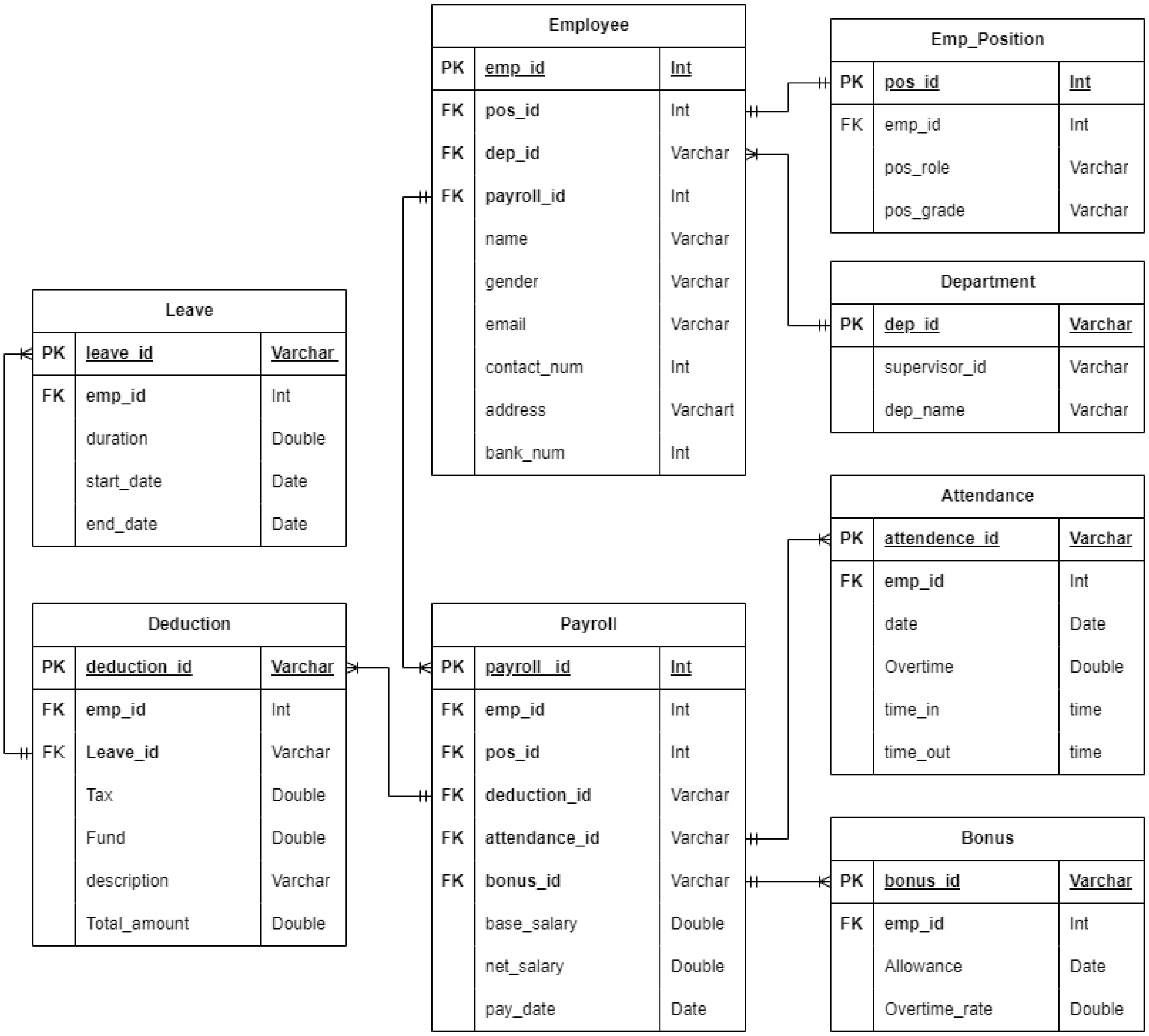
Scalability and Performance: Payroll systems often deal with large volumes of data, especially in organizations with numerous employees. the database system provides scalability, allowing the system to handle increased data volume efficiently. Additionally, it optimizes query performance, ensuring that payroll processing tasks are completed quickly and efficiently.

Reporting and Analysis: The database system enables the generation of comprehensive reports and analyses based on payroll data. It allows for custom queries and data manipulation, making it easier to generate reports such as employee earnings summaries, tax reports, and labor cost analysis. These reports provide valuable insights for decision-making and compliance purposes.

As we know the database system forms the backbone of the payroll system that provides a secure and organized platform for managing employee data, processing payroll transactions, and generating reports. It streamlines payroll processes, enhances data accuracy, and ensures compliance with data security and privacy regulations, ultimately contributing to the smooth functioning of an organization's payroll operations.

II. Database Design

1. An Entity Relationship Diagram (ERD)



Employee		
PK	emp_id	Int
FK	pos_id	Int
FK	dep_id	Varchar
FK	payroll_id	Int
	name	Varchar
	gender	Varchar
	email	Varchar
	contact_num	Int
	address	Varchart
	bank_num	Int

Emp_Position		
PK	pos_id	Int
FK	emp_id	Int
	pos_role	Varchar
	pos_grade	Varchar

Department		
PK	dep_id	Varchar
	supervisor_id	Varchar
	dep_name	Varchar

Leave		
PK	leave_id	Varchar
FK	emp_id	Int
	duration	Double
	start_date	Date
	end_date	Date

Deduction		
PK	deduction_id	Varchar
FK	emp_id	Int
FK	Leave_id	Varchar
	Tax	Double
	Fund	Double
	description	Varchar
	Total_amount	Double

Payroll		
PK	payroll_id	Int
FK	emp_id	Int
FK	pos_id	Int
FK	deduction_id	Varchar
FK	attendance_id	Varchar
FK	bonus_id	Varchar
	base_salary	Double
	net_salary	Double
	pay_date	Date

Attendance		
PK	attendence_id	Varchar
FK	emp_id	Int
	date	Date
	Overtime	Double
	time_in	time
	time_out	time

Bonus		
PK	bonus_id	Varchar
FK	emp_id	Int
	Allowance	Date
	Overtime_rate	Double

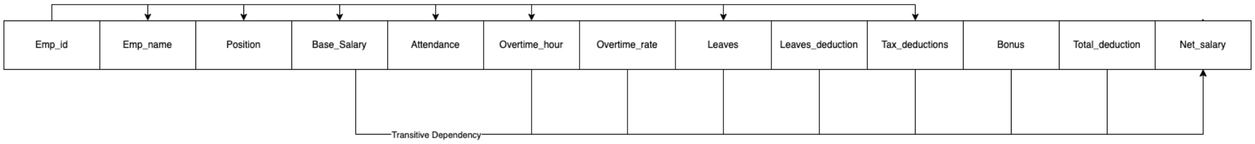
2. Normalization Table

Normalization Table-db assignment

A. 1NF

1NF												
Emp_id	Emp_name	Position	Base_salary	attendance	Overtime_hour	Overtime_rate	Leaves	Leave_deduction	Tax_deduction	Bonus	Total_deduction	Net_salary
001	Adam	Manager	$1,200.00	23	0	$16.00	2	$32.00	$6.00	$350.00	$38.00	$1,512.00
002	Alvin	Intern	$100.00	25	0	$0.00	0	$0.00	$0.50	$250.00	$0.50	$349.50
003	Ban	Director	$1,900.00	23	1.5	$20.00	2	$40.00	$9.50	$250.00	$49.50	$2,130.50
004	Barn	Medior	$600.00	23	2	$9.00	2	$18.00	$3.00	$250.00	$21.00	$847.00
005	Bob	Part-time	$300.00	23	2	$4.00	2	$8.00	$1.50	$250.00	$9.50	$548.50
006	Bam	Junior	$400.00	23	1	$7.00	2	$14.00	$2.00	$250.00	$16.00	$641.00
007	Gojo	Senior	$900.00	25	1	$7.00	0	$0.00	$4.50	$250.00	$4.50	$1,152.50
008	Gin	Director	$1,900.00	22	0	$20.00	3	$60.00	$9.50	$250.00	$69.50	$2,080.50
009	Potter	Medior	$600.00	23.5	4	$9.00	1.5	$13.50	$3.00	$250.00	$16.50	$869.50
010	Satoru	Junior	$400.00	25	5	$7.00	0	$0.00	$2.00	$250.00	$2.00	$683.00

1. Eliminated Repeating Group.
2. Choose a Primary Key: **Emp_id**
3. Identify all the dependencies:



B. 2NF

Since the table doesn't have a composite primary key, the condition for the second normalization is fulfilled

C. 3NF

Identify the transitive dependencies

Table: Employee_Table

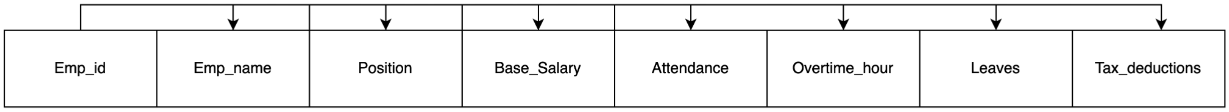
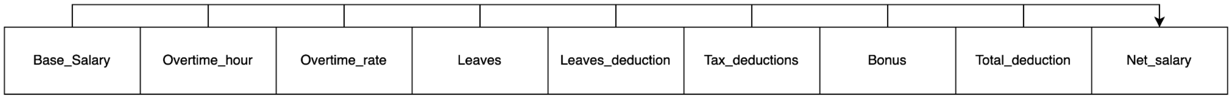


Table: Payroll_Table (Transitive Table)



3NF															
Employee_Table									Payroll_Table						
Emp_id	Emp_name	Position	Base_salary	Attendance	Overtime_hour	Leave	Tax_deduction		Base_Salary	Overtime_Hour	Overtime_Rate	Leave	Leave_Deduction	Tax_deduction	Bonus

3. Data Dictionary

Table	Null	Key	Attribute	length	Data_Type
Employee	Not	**PK**	emp_id		Int
	Not	**FK**	pos_id		Int
	Not	**FK**	dep_id	10	Varchar
	Not	**FK**	payroll_id		Int
	Not		emp_name	25	Varchar
	Not		gender	1	Varchar
			email	25	Varchar
	Not		contact_num	10	Int
			address	50	Varchar
			bank_num		Int
Payroll	Not	**PK**	payroll_id		Int
	Not	**FK**	emp_id		Int
		FK	pos_id		Int
		FK	deduction_id	25	Varchar
		FK	attendance_id	10	Varchar
		FK	bonus_id	10	Varchar
			base_salary		Double
			net_salary		Double
			pay_date		Date
Deduction	Not	**PK**	deduction_id	25	Varchar
	Not	**FK**	emp_id		Int
		FK	leave_id	25	Varchar
			tax		Double
			fund		Double
			description	50	Varchar
			total_amount		Double
Leave		**PK**	leave_id		Varchar
		FK	emp_id		Int
			duration		Double

			start_date		Date
			end_date		Date
Emp_Position	Not	**PK**	pos_id		Int
	Not	**FK**	emp_id		Int
			pos_role	15	Varchar
			pos_grade	15	Varchar
Department	Not	**PK**	dep_id	10	Varchar
			supervisor_id	10	Varchar
	Not		dep_name	25	Varchar
Attendance	Not	**PK**	attendance_id	10	Varchar
	Not	**FK**	emp_id		Int
			date		Date
			overtime		Double
			time_in		Time
			time_out		Time
Bonus	Not	**PK**	bonus_id		Varchar
	Not	**FK**	emp_id		Int
			allowance		Int
			overtime_rate		Double

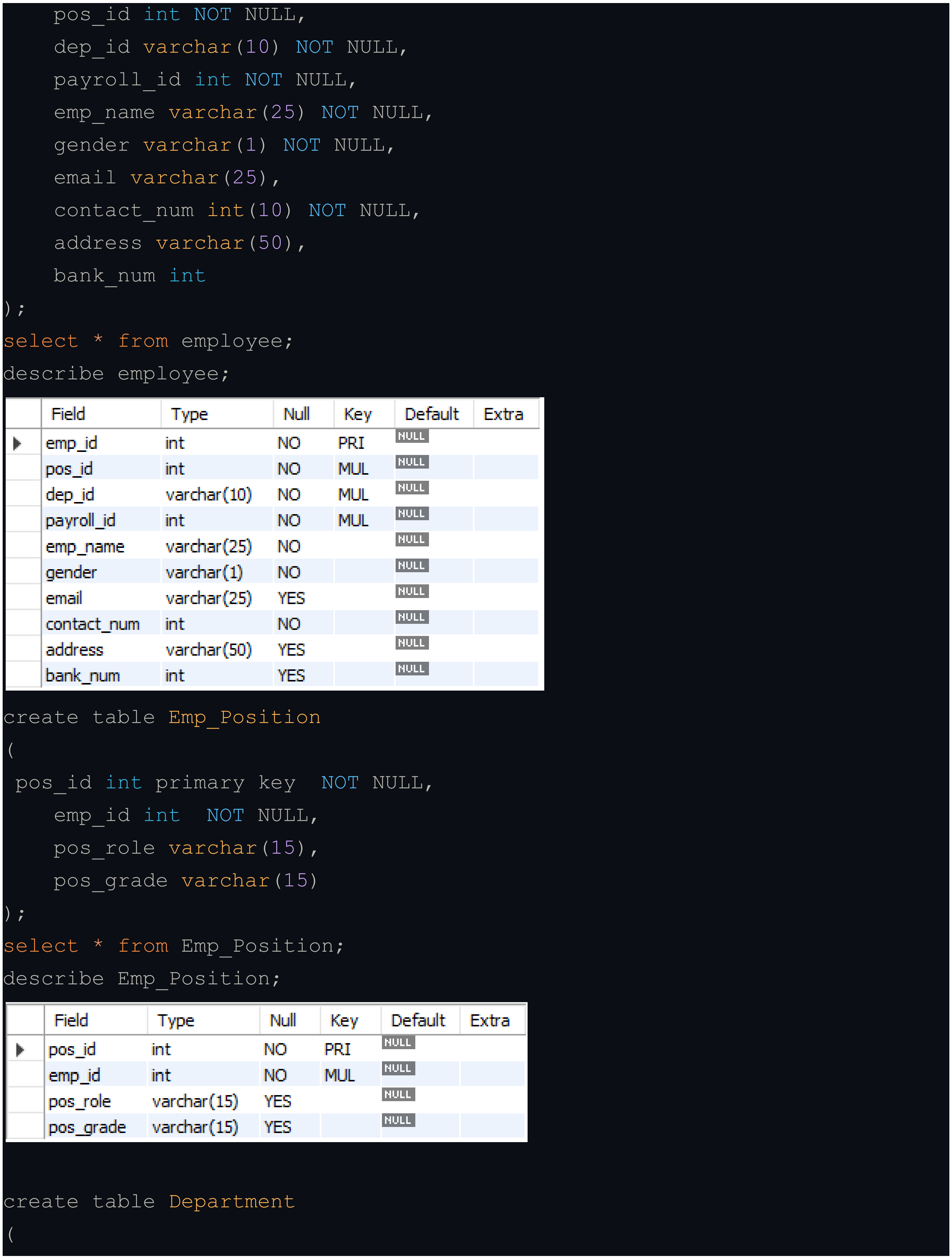
III. List all CREATE,and INSERT commands used to develop the database.

- Create database
- Show tables;

```
create database Payroll_System;
use Payroll_System;
show tables;
```

- Create Table

```
create table Employee
(
    emp_id int primary key NOT NULL,
```



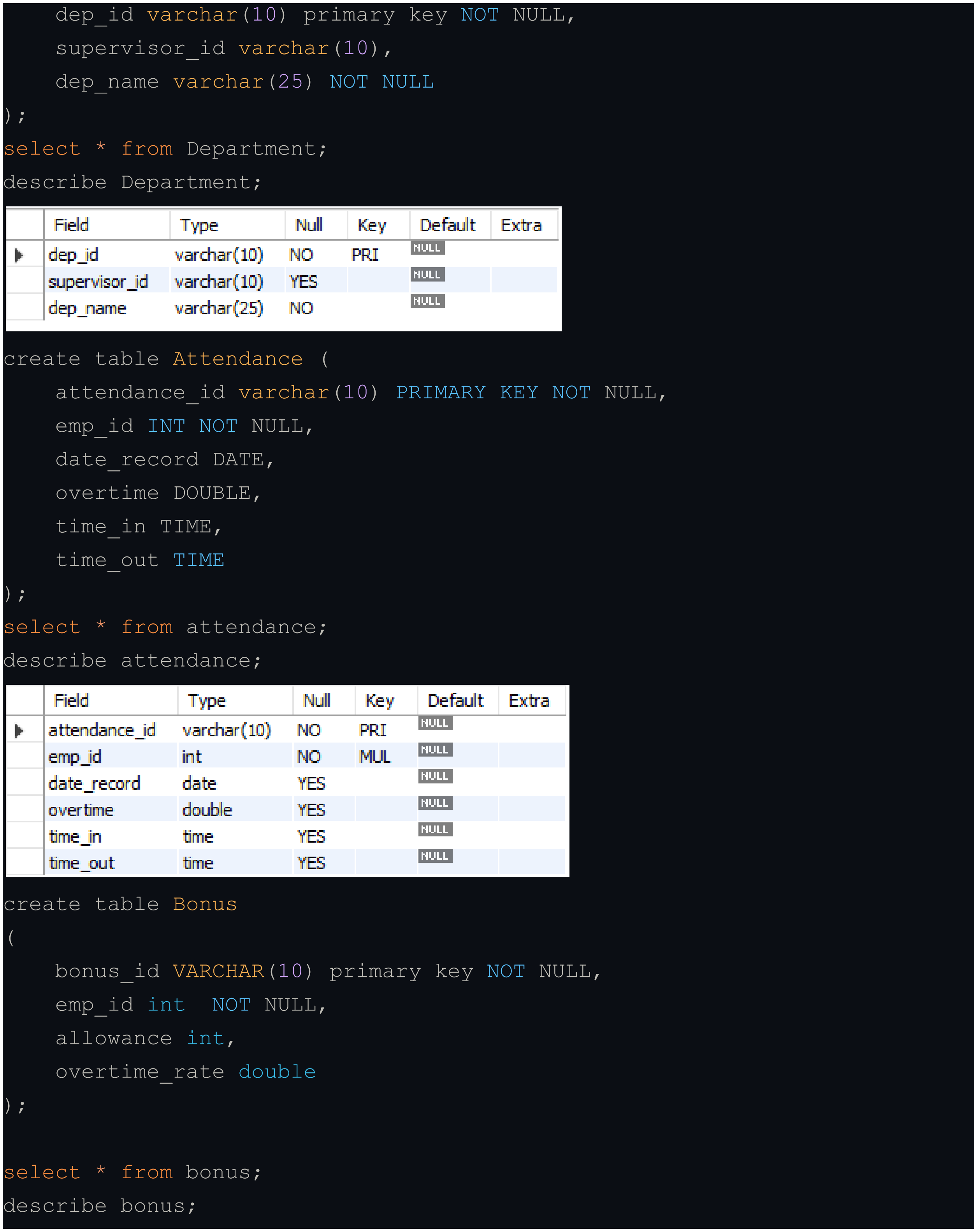
```
    pos_id int NOT NULL,
    dep_id varchar(10) NOT NULL,
    payroll_id int NOT NULL,
    emp_name varchar(25) NOT NULL,
    gender varchar(1) NOT NULL,
    email varchar(25),
    contact_num int(10) NOT NULL,
    address varchar(50),
    bank_num int
);
select * from employee;
describe employee;
```

Field	Type	Null	Key	Default	Extra
emp_id	int	NO	PRI	NULL	
pos_id	int	NO	MUL	NULL	
dep_id	varchar(10)	NO	MUL	NULL	
payroll_id	int	NO	MUL	NULL	
emp_name	varchar(25)	NO		NULL	
gender	varchar(1)	NO		NULL	
email	varchar(25)	YES		NULL	
contact_num	int	NO		NULL	
address	varchar(50)	YES		NULL	
bank_num	int	YES		NULL	

```
create table Emp_Position
(
 pos_id int primary key  NOT NULL,
    emp_id int  NOT NULL,
    pos_role varchar(15),
    pos_grade varchar(15)
);
select * from Emp_Position;
describe Emp_Position;
```

Field	Type	Null	Key	Default	Extra
pos_id	int	NO	PRI	NULL	
emp_id	int	NO	MUL	NULL	
pos_role	varchar(15)	YES		NULL	
pos_grade	varchar(15)	YES		NULL	

```
create table Department
(
```



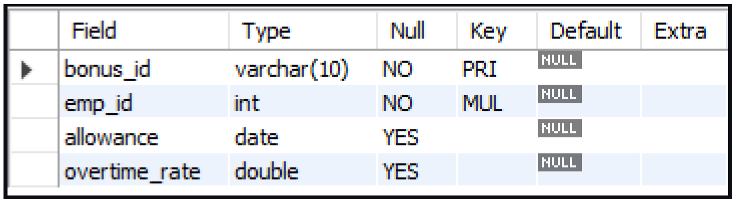
```
    dep_id varchar(10) primary key NOT NULL,
    supervisor_id varchar(10),
    dep_name varchar(25) NOT NULL
);
select * from Department;
describe Department;
```

Field	Type	Null	Key	Default	Extra
dep_id	varchar(10)	NO	PRI	NULL	
supervisor_id	varchar(10)	YES		NULL	
dep_name	varchar(25)	NO		NULL	

```
create table Attendance (
    attendance_id varchar(10) PRIMARY KEY NOT NULL,
    emp_id INT NOT NULL,
    date_record DATE,
    overtime DOUBLE,
    time_in TIME,
    time_out TIME
);
select * from attendance;
describe attendance;
```

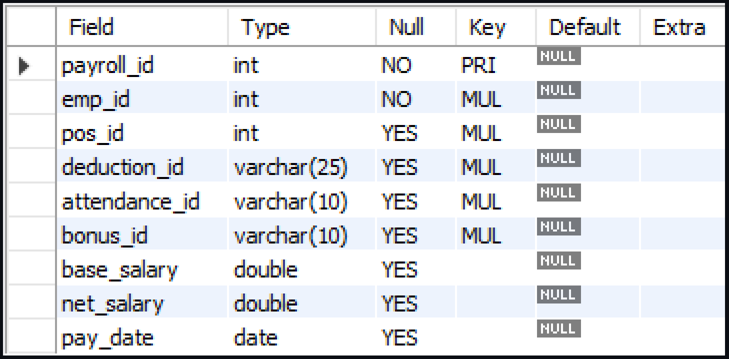
Field	Type	Null	Key	Default	Extra
attendance_id	varchar(10)	NO	PRI	NULL	
emp_id	int	NO	MUL	NULL	
date_record	date	YES		NULL	
overtime	double	YES		NULL	
time_in	time	YES		NULL	
time_out	time	YES		NULL	

```
create table Bonus
(
    bonus_id VARCHAR(10) primary key NOT NULL,
    emp_id int  NOT NULL,
    allowance int,
    overtime_rate double
);

select * from bonus;
describe bonus;
```



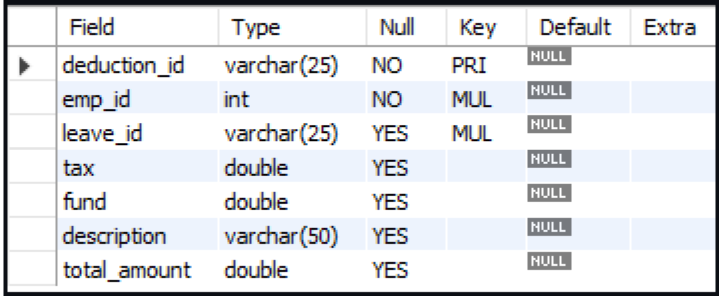
	Field	Type	Null	Key	Default	Extra
▶	bonus_id	varchar(10)	NO	PRI	NULL	
	emp_id	int	NO	MUL	NULL	
	allowance	date	YES		NULL	
	overtime_rate	double	YES		NULL	

```
create table Payroll
(
    payroll_id int primary key  NOT NULL,
    emp_id int  NOT NULL,
    pos_id int,
    deduction_id varchar(25),
    attendance_id varchar(10),
    bonus_id varchar(10),
    base_salary double,
    net_salary double,
    pay_date date
);
select * from payroll;
describe payroll;
```



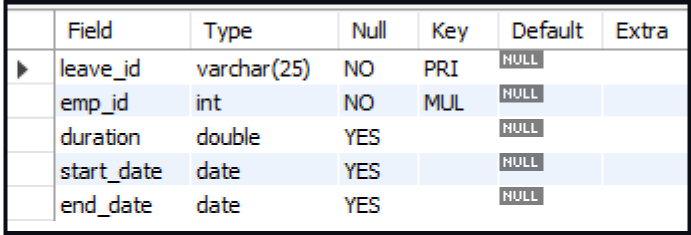
	Field	Type	Null	Key	Default	Extra
▶	payroll_id	int	NO	PRI	NULL	
	emp_id	int	NO	MUL	NULL	
	pos_id	int	YES	MUL	NULL	
	deduction_id	varchar(25)	YES	MUL	NULL	
	attendance_id	varchar(10)	YES	MUL	NULL	
	bonus_id	varchar(10)	YES	MUL	NULL	
	base_salary	double	YES		NULL	
	net_salary	double	YES		NULL	
	pay_date	date	YES		NULL	

```
create table Deduction
(
 deduction_id varchar(25) primary key  NOT NULL,
    emp_id int NOT NULL,
    leave_id varchar(25),
    tax double,
    fund double,
    description varchar(50),
    total_amount double
);
select * from Deduction;
describe Deduction;
```



Field	Type	Null	Key	Default	Extra
deduction_id	varchar(25)	NO	PRI	NULL	
emp_id	int	NO	MUL	NULL	
leave_id	varchar(25)	YES	MUL	NULL	
tax	double	YES		NULL	
fund	double	YES		NULL	
description	varchar(50)	YES		NULL	
total_amount	double	YES		NULL	

```
create table Emp_Leave
(
 leave_id varchar(25) primary key NOT NULL,
    emp_id int  NOT NULL,
    duration double,
    start_date date,
    end_date date
);
select * from Emp_Leave;
describe Emp_Leave;
```



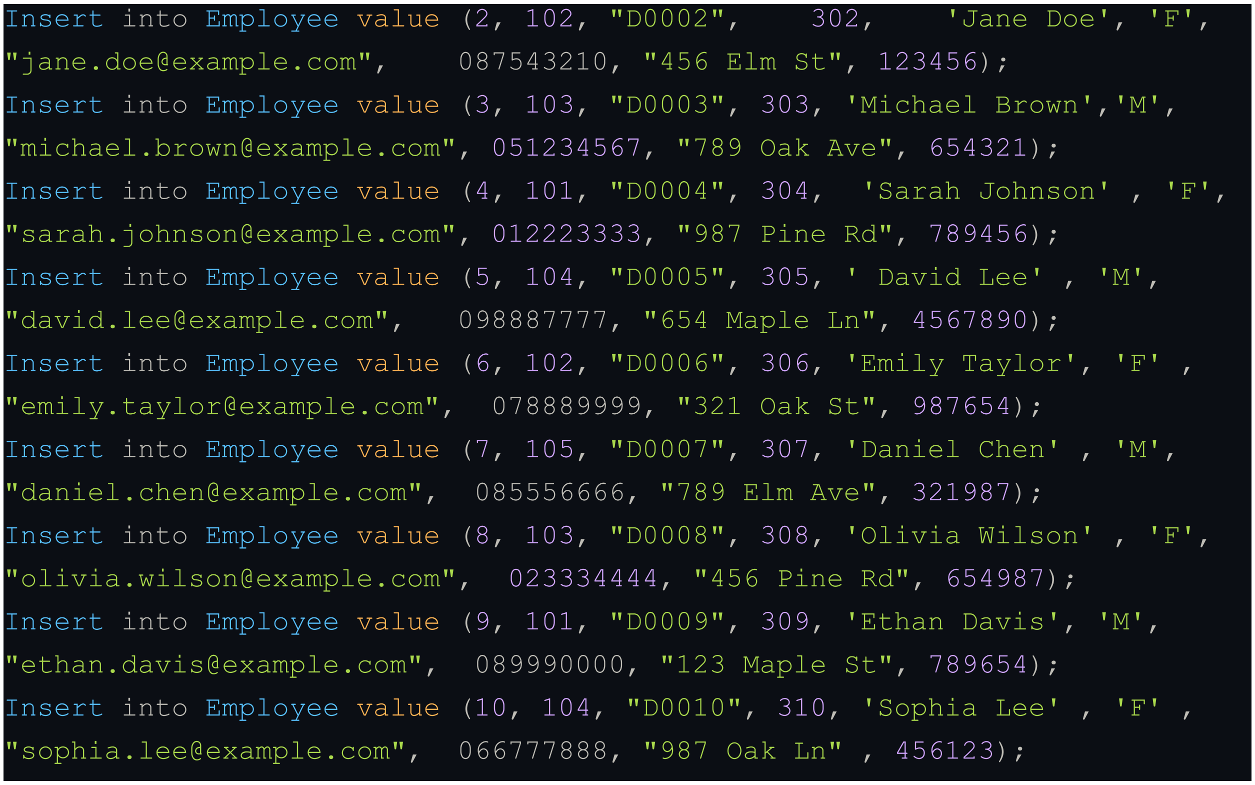
Field	Type	Null	Key	Default	Extra
leave_id	varchar(25)	NO	PRI	NULL	
emp_id	int	NO	MUL	NULL	
duration	double	YES		NULL	
start_date	date	YES		NULL	
end_date	date	YES		NULL	

```
#adding Foreign key
#Employee
alter table Employee add constraint fk_emp_pos foreign key (pos_id)
references Emp_Position(pos_id);
alter table Employee add constraint fk_emp_dep foreign key (dep_id)
references Department(dep_id);
alter table Employee add constraint fk_emp_payroll foreign key
(payroll_id) references Payroll(payroll_id);


#Emp_position
```

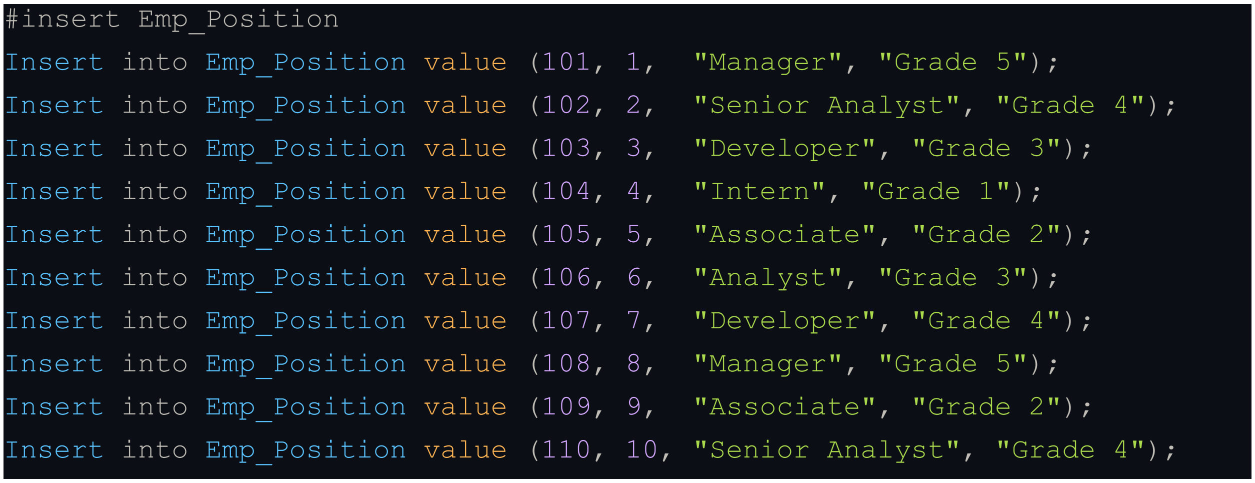
```
alter table Emp_Position add constraint fk_EmPos_emp foreign key (emp_id)
references Employee(emp_id);

#bonus
alter table Bonus add constraint fk_bonus_emp foreign key (emp_id)
references Employee (emp_id);

#payroll
alter table Payroll add constraint fk_payroll_emp foreign key (emp_id)
references Employee (emp_id);
alter table Payroll add constraint fk_payroll_pos foreign key (pos_id)
references Emp_Position (pos_id);
alter table Payroll add constraint fk_payroll_deduct foreign key
(deduction_id) references Deduction (deduction_id);
alter table Payroll add constraint fk_payroll_attend foreign key
(attendance_id) references Attendance (attendance_id);
alter table Payroll add constraint fk_payroll_bonus foreign key (bonus_id)
references Bonus (bonus_id);

#attendance
alter table Attendance add constraint fk_attend_emp foreign key (emp_id)
references Employee (emp_id);

#Deduction
alter table Deduction add constraint fk_deduct_emp foreign key (emp_id)
references Employee (emp_id);
alter table Deduction add constraint fk_deduct_EmLeave foreign key
(leave_id) references Emp_Leave (leave_id);

#Emp_Leave
alter table Emp_Leave add constraint fk_EmLeave_emp foreign key (emp_id)
references Employee (emp_id);
```

- Select to show table
- Insert data

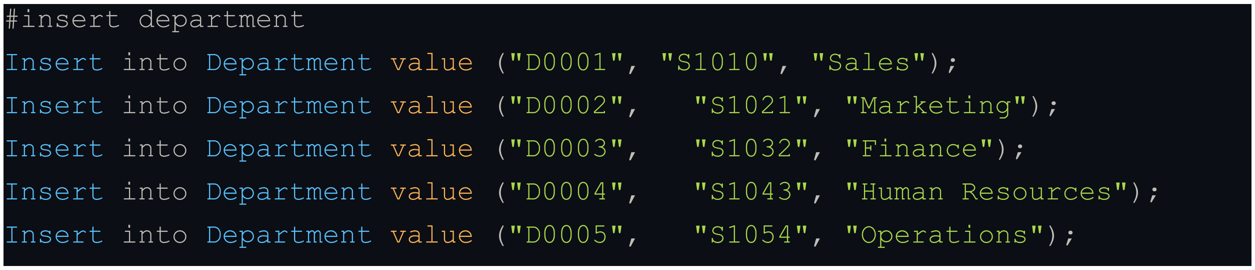
```
#insert Employee
Insert into Employee value (1, 101, "D0001",    301,    'Jame Dove', 'M',
"jame.dove@example.com",  087665210, "486 Elm St", 103456);
```



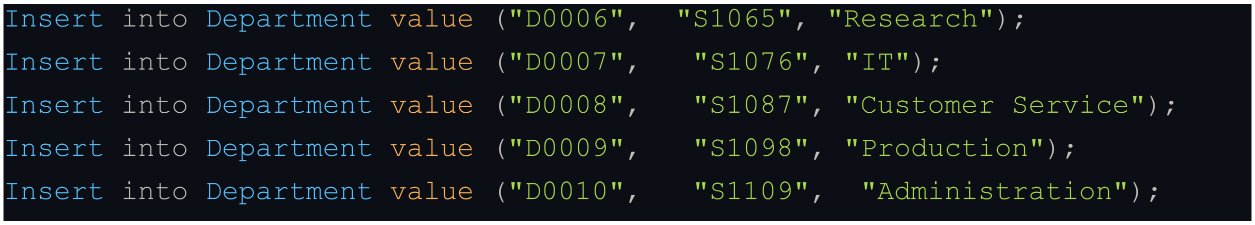
```
Insert into Employee value (2, 102, "D0002",    302,    'Jane Doe', 'F',
"jane.doe@example.com",    087543210, "456 Elm St", 123456);
Insert into Employee value (3, 103, "D0003", 303, 'Michael Brown','M',
"michael.brown@example.com", 051234567, "789 Oak Ave", 654321);
Insert into Employee value (4, 101, "D0004", 304,  'Sarah Johnson' , 'F',
"sarah.johnson@example.com", 012223333, "987 Pine Rd", 789456);
Insert into Employee value (5, 104, "D0005", 305, ' David Lee' , 'M',
"david.lee@example.com",   098887777, "654 Maple Ln", 4567890);
Insert into Employee value (6, 102, "D0006", 306, 'Emily Taylor', 'F' ,
"emily.taylor@example.com",  078889999, "321 Oak St", 987654);
Insert into Employee value (7, 105, "D0007", 307, 'Daniel Chen' , 'M',
"daniel.chen@example.com",  085556666, "789 Elm Ave", 321987);
Insert into Employee value (8, 103, "D0008", 308, 'Olivia Wilson' , 'F',
"olivia.wilson@example.com",  023334444, "456 Pine Rd", 654987);
Insert into Employee value (9, 101, "D0009", 309, 'Ethan Davis', 'M',
"ethan.davis@example.com",  089990000, "123 Maple St", 789654);
Insert into Employee value (10, 104, "D0010", 310, 'Sophia Lee' , 'F' ,
"sophia.lee@example.com",  066777888, "987 Oak Ln" , 456123);
```



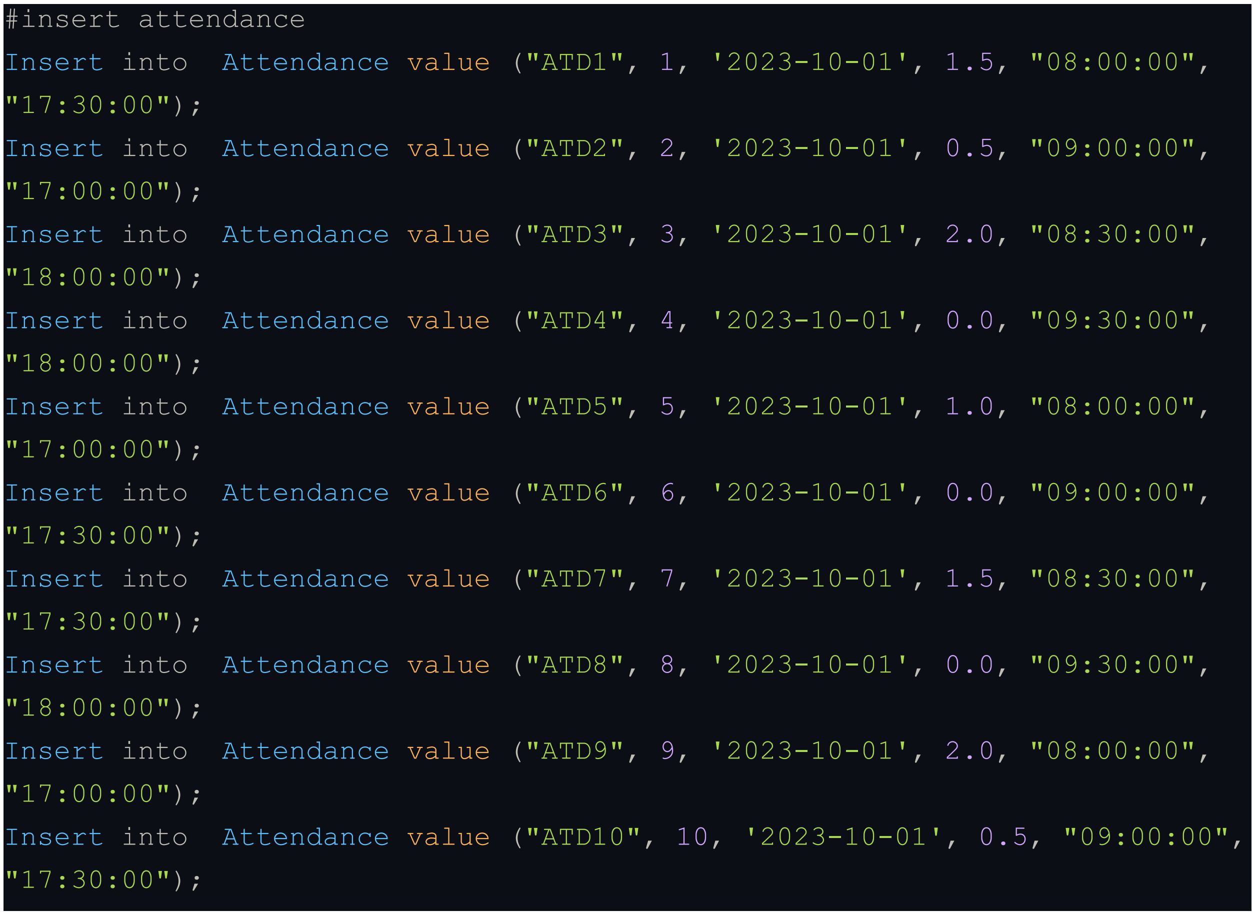
```
#insert Emp_Position
Insert into Emp_Position value (101, 1,  "Manager", "Grade 5");
Insert into Emp_Position value (102, 2,  "Senior Analyst", "Grade 4");
Insert into Emp_Position value (103, 3,  "Developer", "Grade 3");
Insert into Emp_Position value (104, 4,  "Intern", "Grade 1");
Insert into Emp_Position value (105, 5,  "Associate", "Grade 2");
Insert into Emp_Position value (106, 6,  "Analyst", "Grade 3");
Insert into Emp_Position value (107, 7,  "Developer", "Grade 4");
Insert into Emp_Position value (108, 8,  "Manager", "Grade 5");
Insert into Emp_Position value (109, 9,  "Associate", "Grade 2");
Insert into Emp_Position value (110, 10, "Senior Analyst", "Grade 4");
```



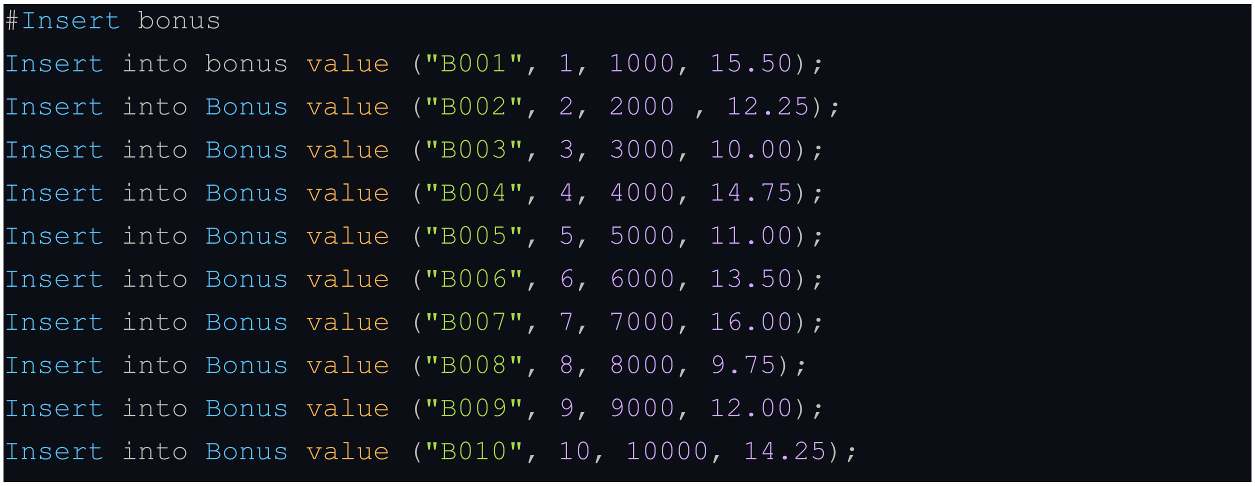
```
#insert department
Insert into Department value ("D0001", "S1010", "Sales");
Insert into Department value ("D0002",   "S1021", "Marketing");
Insert into Department value ("D0003",   "S1032", "Finance");
Insert into Department value ("D0004",   "S1043", "Human Resources");
Insert into Department value ("D0005",   "S1054", "Operations");
```



```
Insert into Department value ("D0006",  "S1065", "Research");
Insert into Department value ("D0007",   "S1076", "IT");
Insert into Department value ("D0008",   "S1087", "Customer Service");
Insert into Department value ("D0009",   "S1098", "Production");
Insert into Department value ("D0010",   "S1109",  "Administration");
```



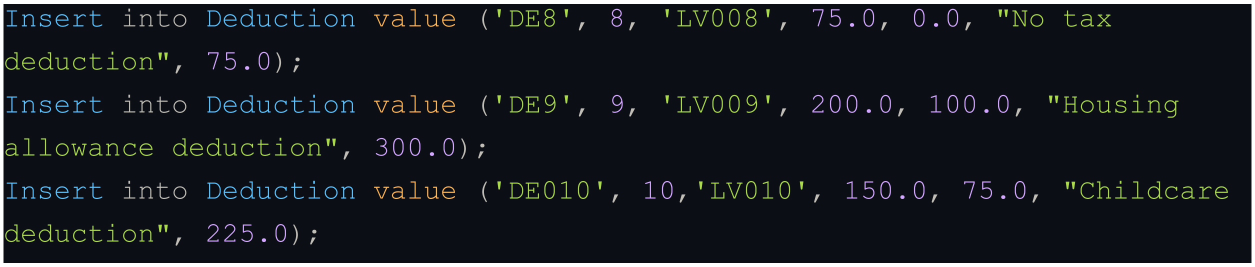
```
#insert attendance
Insert into  Attendance value ("ATD1", 1, '2023-10-01', 1.5, "08:00:00",
"17:30:00");
Insert into  Attendance value ("ATD2", 2, '2023-10-01', 0.5, "09:00:00",
"17:00:00");
Insert into  Attendance value ("ATD3", 3, '2023-10-01', 2.0, "08:30:00",
"18:00:00");
Insert into  Attendance value ("ATD4", 4, '2023-10-01', 0.0, "09:30:00",
"18:00:00");
Insert into  Attendance value ("ATD5", 5, '2023-10-01', 1.0, "08:00:00",
"17:00:00");
Insert into  Attendance value ("ATD6", 6, '2023-10-01', 0.0, "09:00:00",
"17:30:00");
Insert into  Attendance value ("ATD7", 7, '2023-10-01', 1.5, "08:30:00",
"17:30:00");
Insert into  Attendance value ("ATD8", 8, '2023-10-01', 0.0, "09:30:00",
"18:00:00");
Insert into  Attendance value ("ATD9", 9, '2023-10-01', 2.0, "08:00:00",
"17:00:00");
Insert into  Attendance value ("ATD10", 10, '2023-10-01', 0.5, "09:00:00",
"17:30:00");
```



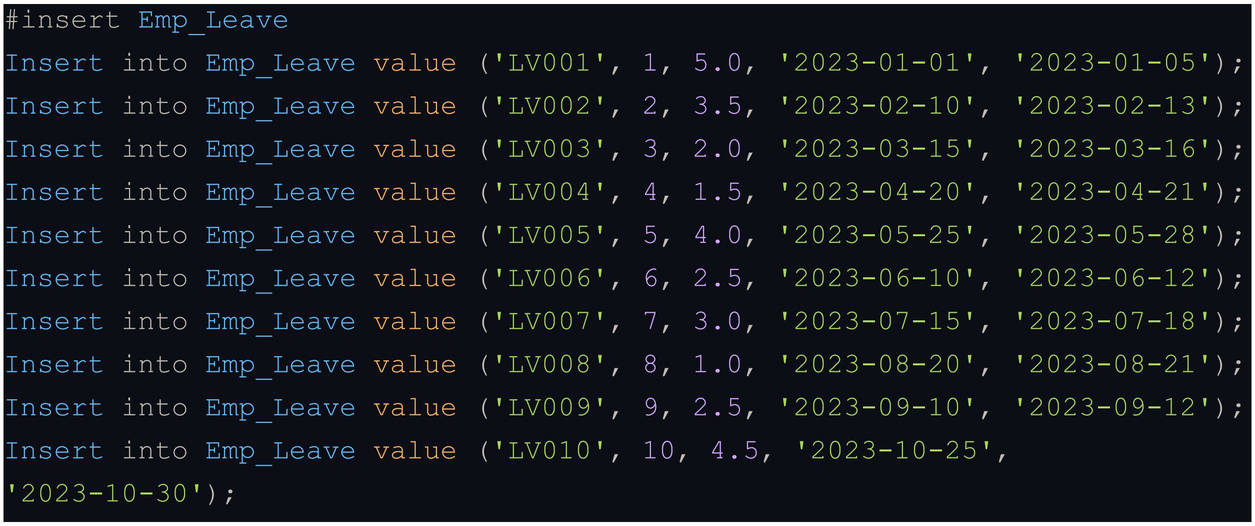
```
#Insert bonus
Insert into bonus value ("B001", 1, 1000, 15.50);
Insert into Bonus value ("B002", 2, 2000 , 12.25);
Insert into Bonus value ("B003", 3, 3000, 10.00);
Insert into Bonus value ("B004", 4, 4000, 14.75);
Insert into Bonus value ("B005", 5, 5000, 11.00);
Insert into Bonus value ("B006", 6, 6000, 13.50);
Insert into Bonus value ("B007", 7, 7000, 16.00);
Insert into Bonus value ("B008", 8, 8000, 9.75);
Insert into Bonus value ("B009", 9, 9000, 12.00);
Insert into Bonus value ("B010", 10, 10000, 14.25);
```

```
#Insert Payroll
Insert into Payroll value (301, 1, 101, 'DE1', 'ATD1', "B001", 5000.00, 4750.00, '2023-10-15');
Insert into Payroll value (302, 2, 102, 'DE2', 'ATD2', "B002", 4000.00, 3850.00, '2023-10-15');
Insert into Payroll value (303, 3, 103, 'DE3', 'ATD3', "B003", 3500.00, 3400.00, '2023-10-15');
Insert into Payroll value (304, 4, 104, 'DE4', 'ATD4', "B004", 3000.00, 2975.00, '2023-10-15');
Insert into Payroll value (305, 5, 105, 'DE5', 'ATD5', "B005", 4500.00, 4375.00, '2023-10-15');
Insert into Payroll value (306, 6, 106, 'DE6', 'ATD6', "B006", 3800.00, 3700.00, '2023-10-15');
Insert into Payroll value (307, 7, 107, 'DE7', 'ATD7', "B007", 5200.00, 5040.00, '2023-10-15');
Insert into Payroll value (308, 8, 108, 'DE8', 'ATD8', "B008", 4200.00, 4080.00, '2023-10-15');
Insert into Payroll value (309, 9, 109, 'DE9', 'ATD9', "B009", 3900.00, 3800.00, '2023-10-15');
Insert into Payroll value (310, 10, 110, 'DE010', 'ATD10',"B010", 4800.00, 4620.00, '2023-10-15');
```

```
#Insert Deduction
Insert into Deduction value ('DE1', 1, 'LV001', 100.0, 50.0, "Medical expense deduction", 150.0);
Insert into Deduction value ('DE2', 2, 'LV002', 75.0, 25.0, "Transportation deduction", 100.0);
Insert into Deduction value ('DE3', 3, 'LV003', 50.0, 0.0, "No deduction", 50.0);
Insert into Deduction value ('DE4', 4, 'LV004', 200.0, 100.0, "Retirement savings deduction", 300.0);
Insert into Deduction value ('DE5', 5, 'LV005', 150.0, 75.0, "Education expense deduction", 225.0);
Insert into Deduction value ('DE6', 6, 'LV006', 50.0, 25.0, "Miscellaneous deduction", 75.0);
Insert into Deduction value ('DE7', 7, 'LV007', 100.0, 50.0, "Health insurance deduction", 150.0);
```



```
Insert into Deduction value ('DE8', 8, 'LV008', 75.0, 0.0, "No tax
deduction", 75.0);
Insert into Deduction value ('DE9', 9, 'LV009', 200.0, 100.0, "Housing
allowance deduction", 300.0);
Insert into Deduction value ('DE010', 10,'LV010', 150.0, 75.0, "Childcare
deduction", 225.0);
```

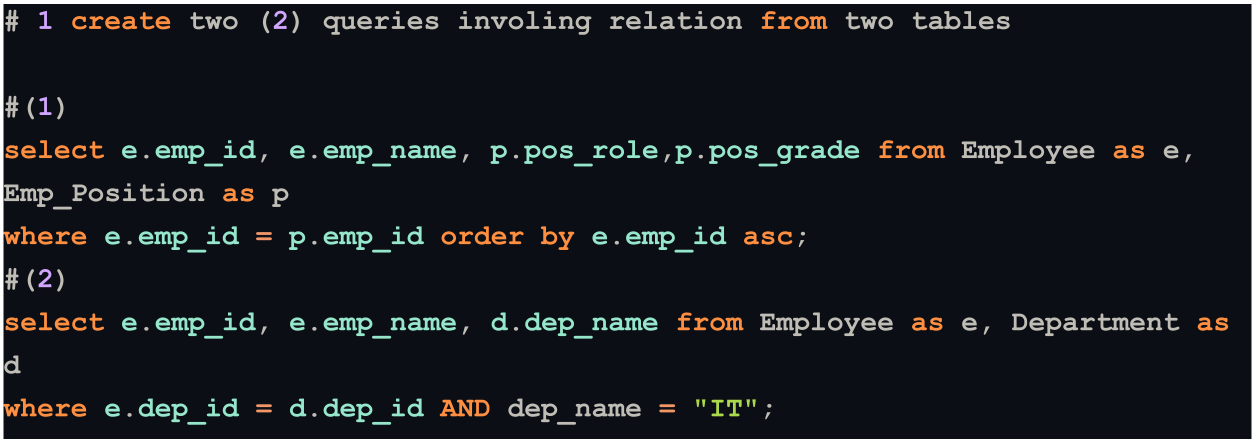


```
#insert Emp_Leave
Insert into Emp_Leave value ('LV001', 1, 5.0, '2023-01-01', '2023-01-05');
Insert into Emp_Leave value ('LV002', 2, 3.5, '2023-02-10', '2023-02-13');
Insert into Emp_Leave value ('LV003', 3, 2.0, '2023-03-15', '2023-03-16');
Insert into Emp_Leave value ('LV004', 4, 1.5, '2023-04-20', '2023-04-21');
Insert into Emp_Leave value ('LV005', 5, 4.0, '2023-05-25', '2023-05-28');
Insert into Emp_Leave value ('LV006', 6, 2.5, '2023-06-10', '2023-06-12');
Insert into Emp_Leave value ('LV007', 7, 3.0, '2023-07-15', '2023-07-18');
Insert into Emp_Leave value ('LV008', 8, 1.0, '2023-08-20', '2023-08-21');
Insert into Emp_Leave value ('LV009', 9, 2.5, '2023-09-10', '2023-09-12');
Insert into Emp_Leave value ('LV010', 10, 4.5, '2023-10-25',
'2023-10-30');
```

IV. Testing & Evaluate

Write ten (10) different queries on your database, using the SELECT, FROM, and WHERE keywords. The ten (10) queries should preferably illustrate six (6) different aspects of database querying such as:

- **TWO (2) queries involving relation from two tables.**



```
# 1 create two (2) queries involing relation from two tables

#(1)
select e.emp_id, e.emp_name, p.pos_role,p.pos_grade from Employee as e,
Emp_Position as p
where e.emp_id = p.emp_id order by e.emp_id asc;
#(2)
select e.emp_id, e.emp_name, d.dep_name from Employee as e, Department as
d
where e.dep_id = d.dep_id AND dep_name = "IT";
```

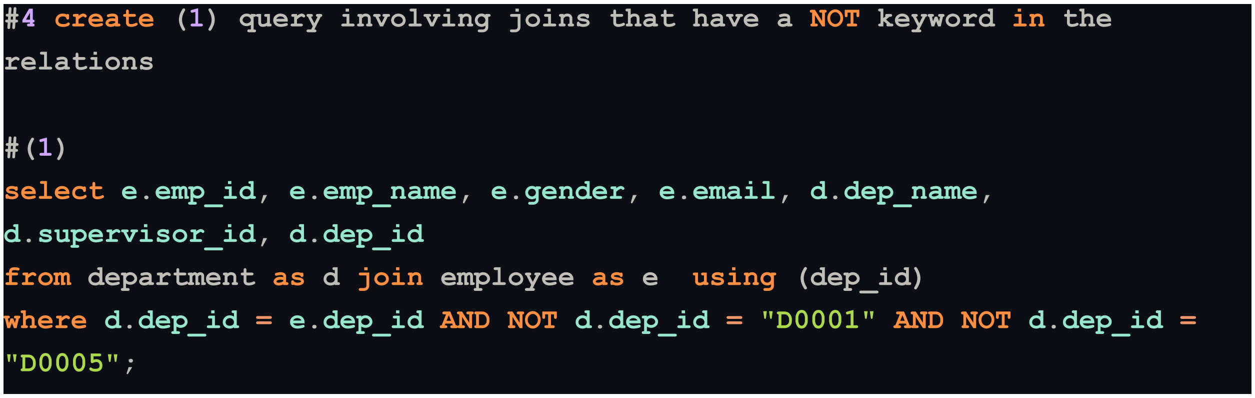
- **Queries involving aggregate functions such as SUM, COUNT, AVG, MAX, MIN.**

```
# 2 create queries involving aggregate functions such as sum, count avg,
max, min

#sum
select sum(net_salary) as The_Total_Sum_Of_Salary from payroll;
#count
select count(base_salary) as The_Amount_Of_Salary from payroll;
#avg
select avg(allowance) as The_Average_Allowance from bonus;
#max
select Max(allowance) as Largest_Allowance from bonus where emp_id <= 5 ;
#min
select emp_id, base_salary, net_salary as Minimum_Salary from payroll
group by emp_id, base_salary, net_salary having net_salary = (select
min(net_salary) from payroll);
```

- **TWO (2) queries involving complicated selects and JOIN from three or more tables.**

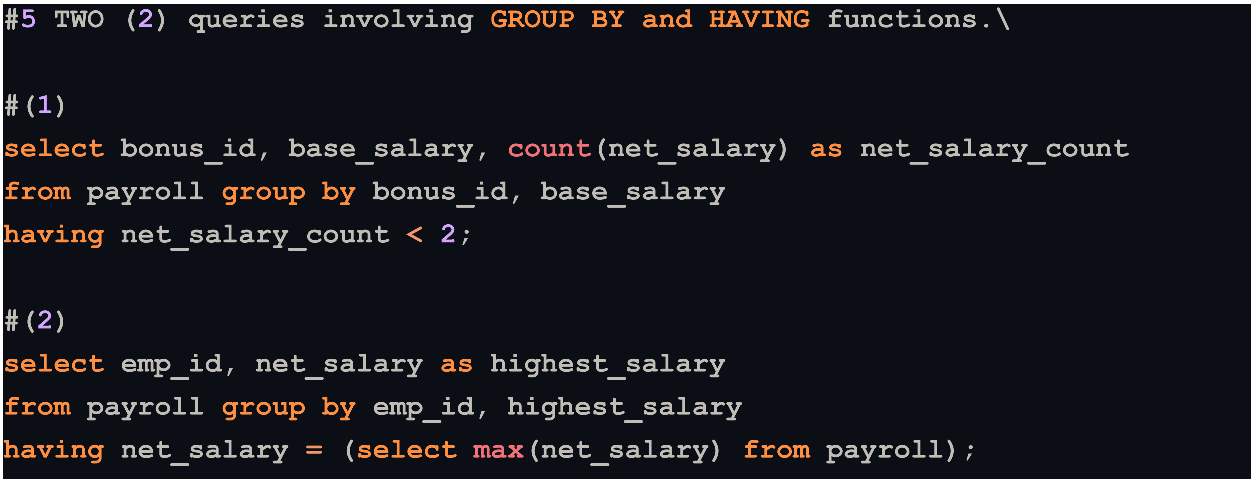
```
#3 create (2) queries involving complicated selects and JOIN from three or
more tables

#(1)
select e.emp_id, emp_name, pos_role, b.allowance, p.net_salary
from employee  e join emp_position  ep on e.emp_id = ep.emp_id
join bonus b on e.emp_id = b.emp_id
join payroll p on b.bonus_id = p.bonus_id;

#(2)
select e.emp_id, emp_name, dep_name, p.base_salary, description
from department d join employee e using (dep_id)
join payroll p using (emp_id)
join deduction de using (deduction_id);
```

- **ONE (1) query involving joins that have a NOT keyword in the relations.**



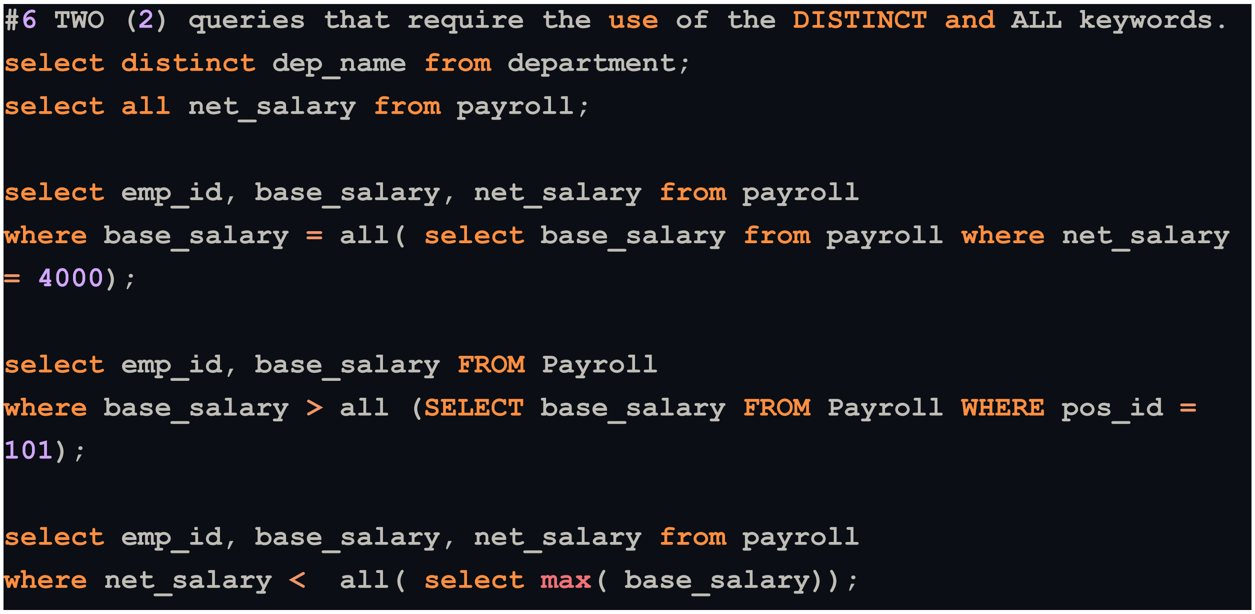
```
#4 create (1) query involving joins that have a NOT keyword in the
relations

#(1)
select e.emp_id, e.emp_name, e.gender, e.email, d.dep_name,
d.supervisor_id, d.dep_id
from department as d join employee as e  using (dep_id)
where d.dep_id = e.dep_id AND NOT d.dep_id = "D0001" AND NOT d.dep_id =
"D0005";
```

- **TWO (2) queries involving GROUP BY and HAVING functions.**



```
#5 TWO (2) queries involving GROUP BY and HAVING functions.\

#(1)
select bonus_id, base_salary, count(net_salary) as net_salary_count
from payroll group by bonus_id, base_salary
having net_salary_count < 2;

#(2)
select emp_id, net_salary as highest_salary
from payroll group by emp_id, highest_salary
having net_salary = (select max(net_salary) from payroll);
```

- **TWO (2) queries that require the use of the DISTINCT and ALL keywords.**



```
#6 TWO (2) queries that require the use of the DISTINCT and ALL keywords.
select distinct dep_name from department;
select all net_salary from payroll;

select emp_id, base_salary, net_salary from payroll
where base_salary = all( select base_salary from payroll where net_salary
= 4000);

select emp_id, base_salary FROM Payroll
where base_salary > all (SELECT base_salary FROM Payroll WHERE pos_id =
101);

select emp_id, base_salary, net_salary from payroll
where net_salary <  all( select max( base_salary));
```

IV. Conclusion

In Conclusion, A payroll database system manages employee pay and benefits data in an organization. Accurate records, automated procedures, increased security, and better reporting are just a few of its numerous advantages. In addition to helping businesses save time and effort, cut down on errors, and adhere to legal requirements, it also helps them streamline payroll procedures. Additionally, it makes handling employee data, including pay, deductions, tax details, and leave balances more effective.

The following are some possible enhancements to the system:

- Connecting it to time and attendance systems will enable automatic tracking of employee work hours, overtime, and attendance.
- Establishing employee self-service portals so that workers can view their payroll data.
- Make it compatible with devices so managers and staff may have better access into the system.
- Including advanced analytics and reporting to offer insights into labor costs, trends, and payroll expenses.
- It should be updated often to maintain correct tax reporting and computations and to comply with the most recent compliance standards.
- integrating it with other financial systems to facilitate smooth data transfer and reconciliation, such as accounting or ERP software.

An essential tool for businesses to efficiently manage their payroll procedures is the payroll database system. They can enhance performance, boost efficiency, and optimize their payroll operations by including new features and keeping them up to date with evolving requirements.

The End