SQL Training

Lesson-End Project Solution

Payroll Calculation

1. Write a query to create the **employee** and **department** tables

SQL code: Employee table

```
CREATE TABLE lep_7.employee (
   emp_id int NOT NULL,
   f_name varchar(45) NULL,
   l_name varchar(45) NOT NULL,
   job_id varchar(45) NOT NULL,
   salary decimal(8,2) NOT NULL,
   manager_id int NOT NULL,
   dept_id varchar(45) NOT NULL,
   PRIMARY KEY(emp_id));
```

SQL code: Department table

```
CREATE TABLE lep_7.department (
   dept_id int NOT NULL,
   dept_name varchar(45) NOT NULL,
   location varchar(45)  NULL,
   manager_id varchar(45) NULL,
   PRIMARY KEY(dept_id));
```

2. Write a query to insert values into the **employee** and **department** tables

SQL code: Employee table

```
INSERT INSERT INTO lep_7. employee
(emp_id,f_name,l_name,job_id,salary,manager_id,dept_id) VALUES
('103','krishna','gee','HP125','500000','05','44')
```

SQL code: Department table

```
INSERT INTO lep_7. department (dept_id,dept_name,location,manager_id) VALUES
('24','production','india','2');
```

3. Write a query to create a view of the **employee** and **department** tables

SQL code:

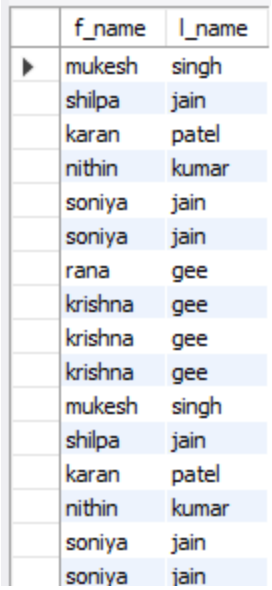
```
CREATE VIEW emp AS SELECT f_name,l_name,salary ,dept_name,location,emp_id
FROM lep_7.employee,lep_7.department WHERE l_name = 'jain';
```

4. Write a query to display the first and last names of every employee in the **employee** table whose salary is greater than the average salary of the employees listed in the **SQL basics** table

SQL code:

```
SELECT e.f_name,e.l_name
FROM lep_7.employee e,sqlbasics.emp s
WHERE e.salary > s.salary;
```

Output:



f_name	l_name
mukesh	singh
shilpa	jain
karan	patel
nithin	kumar
soniya	jain
soniya	jain
rana	gee
krishna	gee
krishna	gee
krishna	gee
mukesh	singh
shilpa	jain
karan	patel
nithin	kumar
soniya	jain
soniya	jain

5. Write a query to change the delimiter to //

   SQL code:

```
delimiter //
```

6. Write a query to create a stored procedure in the **employe**e table for every employee whose salary is greater than or equal to 250,000

   SQL code:

```
use lep_7;
SELECT * from employee;
delimiter &&
CREATE PROCEDURE top_salarys()
BEGIN
SELECT job_id,f_name,salary
FROM employee where salary>=250000;
END &&
delimiter ;;
```

7. Write a query to execute the stored procedure

   SQL code:

   call top_salarys();

   Output:

	job_id	f_name	salary
▶	HP125	krishna	500000.00
	HP122	rana	250000.00
	HP121	soniya	400000.00
	HP120	nithin	300000.00
	HP126	karan	300001.00
	HP127	shilpa	300001.00
	HP128	mukesh	300001.00

8. Write a query to create and execute a stored procedure with one parameter using the **order by** function in descending order of the salary earned

   SQL code:

```
delimiter //
CREATE PROCEDURE sort_salarys(IN var INT)
BEGIN
SELECT job_id,f_name,salary
FROM employee ORDER BY salary DESC LIMIT var;
end //
delimiter ;

call sort_salarys(3);
```

   Output:

	job_id	f_name	salary
▶	HP125	krishna	500000.00
	HP121	soniya	400000.00
	HP126	karan	300001.00