LAB CYCLE-1

I. DDL Commands:

1. Create the database named *HR* and the above tables in the *HR* database; include the Primary Key Constraint, Referential Integrity Constraints as shown in the schema diagram.

create table regions(region_id number(5) primary key,region_name varchar(25));

Table created.

create table countries(country_id char(5) primary key,country_name varchar(40),region_id number(5),foreign key(region_id) references regions(region_id));

Table created.

create table locations(location_id number(5) primary key,street_address varchar(25),postal_code varchar(12),city varchar(30),state_province varchar(12),country_id char(5),
            foreign key(country_id) references countries(country_id));

Table created.

create table departments(department_id number(3) primary key,department_name varchar(30),manager_id number(5),location_id number(3),foreign key(location_id) references locations(location_id));

Table created.

create table job_grades(grade_level varchar(2) primary key,lowest_sal number(5),highest_sal number(5));

Table created.

create table jobs(job_id varchar(10) primary key,job_title varchar(35),min_salary number(10),max_salary number(10));

Table created.

create table employees(employee_id number(3) primary key,first_name varchar(20),last_name varchar(25),email varchar(25),phone_number varchar(20),hire_date date,job_id varchar(10),salary number(10), commission_pct number(5),manager_id number(3),department_id number(3),foreign key(job_id) references jobs(job_id),foreign key(department_id) references departments(department_id));

Table created.

create table job_history(employee_id number(3),start_date date,end_date date,job_id varchar(10),department_id number(3),primary key(employee_id,start_date),foreign key(employee_id) references employees(employee_id), foreign key(job_id) references jobs(job_id),foreign key(department_id) references departments(department_id));

Table created.

2. Add a field *Circle* to the table *regions* with the default value set to 'East' and display its modified schema.

alter table regions add circle varchar(3);

Table altered.

alter table regions modify circle varchar(4) default 'East';

Table altered.

desc regions;

TABLE REGIONS

Column	Null?	Type
REGION_ID	NOT NULL	NUMBER(5,0)
REGION_NAME	-	VARCHAR2(25)
CIRCLE	-	VARCHAR2(4)

alter table locations modify state_province varchar(20);

Table altered.

3. Set a constraint to the SALARY field in the *employees* table that accepts only amount greater than zero.

alter table employees modify salary number(10) check(salary>0);

Table altered.

4. Modify the table *countries* by changing the width of the field COUNTRY_NAME to 30.

alter table countries modify country_name varchar(30);

Table altered.

5. Make the PHONE_NUMBER field in *employees* unique and check the EMAIL is a valid gmail account.

alter table employees modify phone_number varchar(20) unique;

Table altered.

alter table employees modify email varchar(25) check(email like '_%@%.%');

Table altered.

6. Drop the column *Circle* from the *regions* table.

alter table regions drop column circle;

Table altered.

7. Create a view for employees who get commission by hiding salary information. Display the contents of created view.

create view view_name as select
employee_id,first_name,last_name,email,phone_number,hire_date,job_id,commission_pct,manager_id,department_id from employees where commission_pct!=0;

```
View created.
```

II. DML Commands:

8. Insert details in the above tables using the values given below.

insert into regions values(1,'zone1');
```
1 row(s) inserted.
```

insert into regions values(2,'zone2');
```
1 row(s) inserted.
```

insert into regions values(3,'zone3');
```
1 row(s) inserted.
```

insert into regions values(4,'zone4');
```
1 row(s) inserted.
```

select *from regions;

REGION_ID	REGION_NAME
1	zone1
2	zone2
3	zone3
4	zone4

insert into countries values('AU','AUSTRALIA',3);
```
1 row(s) inserted.
```

insert into countries values('CA','CANADA',2);
```
1 row(s) inserted.
```

insert into countries values('CN','CHINA',3);
```
1 row(s) inserted.
```

insert into countries values('IN','INDIA',1);
```
1 row(s) inserted.
```

insert into countries values('US','AMERICA',2);

    1 row(s) inserted.

insert into countries values('JP','JAPAN',4)

    1 row(s) inserted.

insert into countries values('IT','ITALY',1);

    1 row(s) inserted.

insert into countries values('MX','MEXICO',2);

    1 row(s) inserted.

insert into countries values('UK','UNITED KINGDOM',1);

    1 row(s) inserted.

select *from countries;

COUNTRY_ID	COUNTRY_NAME	REGION_ID
AU	AUSTRALIA	3
CA	CANADA	2
CN	CHINA	3
IN	INDIA	1
US	AMERICA	2
JP	JAPAN	4
IT	ITALY	1
MX	MEXICO	2
UK	UNITED KINGDOM	1

insert into locations values(1000,'147 BOX WOOD ST.','15643','ROMA','NIL','IT');

    1 row(s) inserted.

insert into locations values(1100,'234 CHURCH RD.','23416','TOKYO','TOKPREFECT','JP');

    1 row(s) inserted.

insert into locations values(1200,'12/98 VICTORIA RD.','1657','TORONTO','ONTARIO','CA');

    1 row(s) inserted.

insert into locations values(1300,'820 ARTHUR RD.','NIL','LONDON','NIL','UK');

> 1 row(s) inserted.

insert into locations values(1400,'2004 CAROL RD','7645','SEETTLE','WASHINGTON','US');

> 1 row(s) inserted.

insert into locations values(1500,'12 CHESTER RD.','8732','WHITEHORSE','YUKON','CA');

> 1 row(s) inserted.

insert into locations values(1600,'MARIANO 9991','11932','MEXICO CITY','DISTRITO FED','MX');

> 1 row(s) inserted.

insert into locations values(1700,'CROWN HILL','678','BROWNSWICK','NEW JERSEY','US');

> 1 row(s) inserted.

insert into locations values(1800,'123 MARINA RD.','123456','MUMBAI','MAHARASHTRA','IN');

> 1 row(s) inserted.

insert into locations values(1900,'29 SEA LANE','3456','SYDNEY','NEW S.WALES','AU');

> 1 row(s) inserted.

select *from locations;

LOCATION_ID	STREET_ADDRESS	POSTAL_CODE	CITY	STATE_PROVINCE	COUNTRY_ID
1000	147 BOX WOOD ST.	15643	ROMA	NIL	IT
1100	234 CHURCH RD.	23416	TOKYO	TOKPREFECT	JP
1200	12/98 VICTORIA RD.	1657	TORONTO	ONTARIO	CA
1300	820 ARTHUR RD.	NIL	LONDON	NIL	UK
1400	2004 CAROL RD	7645	SEETTLE	WASHINGTON	US
1500	12 CHESTER RD.	8732	WHITEHORSE	YUKON	CA
1600	MARIANO 9991	11932	MEXICO CITY	DISTRITO FED	MX
1700	CROWN HILL	678	BROWNSWICK	NEW JERSEY	US
1800	123 MARINA RD.	123456	MUMBAI	MAHARASHTRA	IN
1900	29 SEA LANE	3456	SYDNEY	NEW S.WALES	AU

insert into job_grades values('A',1000,2999);

> 1 row(s) inserted.

insert into job_grades values('B',3000,5999);

1 row(s) inserted.

insert into job_grades values('C',6000,9999);
1 row(s) inserted.

insert into job_grades values('D',10000,14999);
1 row(s) inserted.

insert into job_grades values('E',15000,24999);
1 row(s) inserted.

insert into job_grades values('F',25000,40000);
1 row(s) inserted.

select *from job_grades;

GRADE_LEVEL	LOWEST_SAL	HIGHEST_SAL
A	1000	2999
B	3000	5999
C	6000	9999
D	10000	14999
E	15000	24999
F	25000	40000

insert into jobs values('AD_PRES','PRESIDENT',20080,40000);
1 row(s) inserted.

insert into jobs values('SA_REP','SALES REPRESENTATIVES',6000,12008);
1 row(s) inserted.

insert into jobs values('AD_VP','ADMN.VICE PRESIDENT',15000,30000);
1 row(s) inserted.

insert into jobs values('IT_PROG','PROGRAMMER',6000,11000);
1 row(s) inserted.

insert into jobs values('SA_MAN','SALES MANAGER',10000,20080);
1 row(s) inserted.

insert into jobs values('CLERK','CLERK',2500,7000);

> 1 row(s) inserted.

insert into jobs values('HR_REP','HR REPRESENTATIVE',5000,10000);

> 1 row(s) inserted.

insert into jobs values('ACCOUNTANT','ACCOUNTANT',4200,9000);

> 1 row(s) inserted.

insert into jobs values('FIN_MAN','FINANCE MANAGER',8200,16000);

> 1 row(s) inserted.

select *from jobs;

JOB_ID	JOB_TITLE	MIN_SALARY	MAX_SALARY
AD_PRES	PRESIDENT	20080	40000
SA_REP	SALES REPRESENTATIVES	6000	12008
AD_VP	ADMN.VICE PRESIDENT	15000	30000
IT_PROG	PROGRAMMER	6000	11000
SA_MAN	SALES MANAGER	10000	20080
CLERK	CLERK	2500	7000
HR_REP	HR REPRESENTATIVE	5000	10000
ACCOUNTANT	ACCOUNTANT	4200	9000
FIN_MAN	FINANCE MANAGER	8200	16000

alter table departments modify location_id number(5);

> Table altered.

insert into departments values(10,'administration',200,1700);

> 1 row(s) inserted.

insert into departments values(20,'marketing',201,1900);

> 1 row(s) inserted.

insert into departments values(30,'purchasing',114,1700);

> 1 row(s) inserted.

insert into departments values(40,'human resource',203,1000);

```
1 row(s) inserted.
```

insert into departments values(50,'finance',108,1200);
```
1 row(s) inserted.
```

insert into departments values(60,'accounting',205,1800);
```
1 row(s) inserted.
```

insert into departments values(70,'shipping',121,1500);
```
1 row(s) inserted.
```

insert into departments values(80,'sales',145,1900);
```
1 row(s) inserted.
```

insert into departments values(90,'manufacturing',0,1700);
```
1 row(s) inserted.
```

insert into departments values(100,'construction',0,1300);
```
1 row(s) inserted.
```

insert into departments values(110,'it',103,1400);
```
1 row(s) inserted.
```

insert into departments values(120,'recruiting',0,1600);
```
1 row(s) inserted.
```

insert into departments values(130,'payroll',100,1800);
```
1 row(s) inserted.
```

select *from departments;

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	administration	200	1700
20	marketing	201	1900
30	purchasing	114	1700
40	human resource	203	1000
50	finance	108	1200
60	accounting	205	1800
70	shipping	121	1500
80	sales	145	1900
90	manufacturing	0	1700
100	construction	0	1300
110	it	103	1400
120	recruiting	0	1600
130	payroll	100	1800

insert into employees values(100,'STEVEN','KING','SKING@GMAIL.COM',8182345210,'17-JUN-2003','AD_PRES',28000,0,0,90);

```
1 row(s) inserted.
```

insert into employees values(101,'NEENA','KOCHHAR','NKOCHHAR@GMAIL.COM',9897561234,'21-SEP-2005','AD_VP',22000,0,100,130);

```
1 row(s) inserted.
```

insert into employees values(102,'LEX','DE HAAN','LDEHAAN@GMAIL.COM',9554667890,'13-JAN-2001','AD_VP',30000,0,100,130);

```
1 row(s) inserted.
```

insert into employees values(103,'ALEXANDER','HUNOLD','AHUNOLD@GMAIL.COM',8678905431,'03-JAN-2006','IT_PROG',10500,0,103,110);

```
1 row(s) inserted.
```

insert into employees values(104,'DAVID','AUSTIN','DAUSTIN@GMAIL.COM',9495235678,'10-MAY-2007','IT_PTOG',10000,0,103,110);

```
1 row(s) inserted.
```

insert into employees values(105,'DIANA','LORENTZ','DLORENTZ@GMAIL.COM',9876542310,'20-NOV-2009','IT_PROG',8000,0,103,110);

```
1 row(s) inserted.
```

```sql
insert into employees values(106,'NANCY','GREENBERG','NGREEN@GMAIL.COM',8769054324,'30-OCT-2010','IT_PROG',7000,0,103,110);
```
1 row(s) inserted.

```sql
insert into employees values(107,'JOHN','RUSSEL','JRUSSEL@GMAIL.COM',6778653490,'09-OCT-2004','SA_MAN',14000,0.4,201,20);
```
1 row(s) inserted.

```sql
insert into employees values(108,'PETER','TUKER','PTUKER@GMAIL.COM',9067865423,'30-JAN-2005','SA_MAN',10000,0.3,201,20);
```
1 row(s) inserted.

```sql
insert into employees values(109,'NEIL','JOSEPH','NJOSEPH@GMAIL.COM',7689054321,'31-MAR-2006','SA_MAN',12000,0.2,201,20);
```
1 row(s) inserted.

```sql
insert into employees values(110,'JANETTE','KING','JKING@GMAIL.COM',9809897645,'04-MAR-2006','SA_REP',10000,0.35,145,80);
```
1 row(s) inserted.

```sql
insert into employees values(111,'WILLIAM','JOHN','WJOHN@GMAIL.COM',7682315672,'23-NOV-2008','CLERK',6000,0,200,10);
```
1 row(s) inserted.

```sql
insert into employees values(112,'ANITTA','BAKER','ABAKER@GMAIL.COM',2345678910,'15-SEP-2010','HR_REP',9000,0,203,40);
```
1 row(s) inserted.

```sql
insert into employees values(113,'NANDITA','MOHAN','NMOHAN@GMAIL.COM',94445634213,'20-JUL-2014','ACCOUNTANT',8000,0,205,60);
```
1 row(s) inserted.

```sql
insert into employees values(114,'SUSAN','VARKEY','SVARKEY@GMAIL.COM',8976054321,'14-DEC-2004','FIN_MAN',15000,0,108,50);
```
1 row(s) inserted.

```sql
select *from employees;
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
100	STEVEN	KING	SKING@GMAIL.COM	8182345210	17-JUN-03	AD_PRES	28000	0	0	90
101	NEENA	KOCHHAR	NKOCHHAR@GMAIL.COM	9897561234	21-SEP-05	AD_VP	22000	0	100	130
102	LEX	DE HAAN	LDEHAAN@GMAIL.COM	9554667890	13-JAN-01	AD_VP	30000	0	100	130
103	ALEXANDER	HUNOLD	AHUNOLD@GMAIL.COM	8678905431	03-JAN-06	IT_PROG	10500	0	103	110
104	DAVID	AUSTIN	DAUSTIN@GMAIL.COM	9495235678	10-MAY-07	IT_PTOG	10000	0	103	110
105	DIANA	LORENTZ	DLORENTZ@GMAIL.COM	9876542310	20-NOV-09	IT_PROG	8000	0	103	110
106	NANCY	GREENBERG	NGREEN@GMAIL.COM	8769054324	30-OCT-10	IT_PROG	7000	0	103	110
107	JOHN	RUSSEL	JRUSSEL@GMAIL.COM	6778653490	09-OCT-04	SA_MAN	14000	.4	201	20
108	PETER	TUKER	PTUKER@GMAIL.COM	9067865423	30-JAN-05	SA_MAN	10000	.3	201	20
109	NEIL	JOSEPH	NJOSEPH@GMAIL.COM	7689054321	31-MAR-06	SA_MAN	12000	.2	201	20
110	JANETTE	KING	JKING@GMAIL.COM	9809897645	04-MAR-06	SA_REP	10000	.35	145	80
111	WILLIAM	JOHN	WJOHN@GMAIL.COM	7682315672	23-NOV-08	CLERK	6000	0	200	10
112	ANITTA	BAKER	ABAKER@GMAIL.COM	2345678910	15-SEP-10	HR_REP	9000	0	203	40
113	NANDITA	MOHAN	NMOHAN@GMAIL.COM	94445634213	20-JUL-14	ACCOUNTANT	8000	0	205	60
114	SUSAN	VARKEY	SVARKEY@GMAIL.COM	8976054321	14-DEC-04	FIN MAN	15000	0	108	50

insert into job_history values(102,'13-JAN-2001','24-JUL-2006','IT_PROG',110);

    1 row(s) inserted.

insert into job_history values(101,'21-SEP-1997','27-OCT-2001','ACCOUNTANT',60);

    1 row(s) inserted.

insert into job_history values(101,'28-OCT-2001','15-MAR-2005','FIN_MAN',50);

    1 row(s) inserted.

insert into job_history values(112,'17-FEB-2004','19-DEC-2007','HR_REP',40);

    1 row(s) inserted.

insert into job_history values(114,'24-MAR-2006','31-DEC-2007','CLERK',10);

    1 row(s) inserted.

insert into job_history values(112,'01-JAN-2007','31-DEC-2007','AD_VP',130);

    1 row(s) inserted.

insert into job_history values(109,'24-MAR-2006','31-DEC-2006','SA_MAN',20);

    1 row(s) inserted.

insert into job_history values(105,'01-JUL-2002','31-DEC-2005','SA_REP',110);

> 1 row(s) inserted.

insert into job_history values(105,'17-FEB-2006','19-DEC-2007','IT_PROG',110);

> 1 row(s) inserted.

select *from job_history;

EMPLOYEE_ID	START_DATE	END_DATE	JOB_ID	DEPARTMENT_ID
102	13-JAN-01	24-JUL-06	IT_PROG	110
101	21-SEP-97	27-OCT-01	ACCOUNTANT	60
101	28-OCT-01	15-MAR-05	FIN_MAN	50
112	17-FEB-04	19-DEC-07	HR_REP	40
114	24-MAR-06	31-DEC-07	CLERK	10
112	01-JAN-07	31-DEC-07	AD_VP	130
109	24-MAR-06	31-DEC-06	SA_MAN	20
105	01-JUL-02	31-DEC-05	SA_REP	110
105	17-FEB-06	19-DEC-07	IT_PROG	110

9. Update the street_Address of Roma city to *115 Monte Carlo Rd.* in *locations* table.

update locations set street_address='115 MONTE CARLO RD.' where city='ROMA';

> 1 row(s) updated.

select *from locations;

LOCATION_ID	STREET_ADDRESS	POSTAL_CODE	CITY	STATE_PROVINCE	COUNTRY_ID
1000	115 MONTE CARLO RD.	15643	ROMA	-	IT
1100	234 CHURCH RD.	23416	TOKYO	TOKYO PREFECTURE	JP
1200	12/98 VICTORIA RD.	1657	TORONTO	ONTARIO	CA
1300	820 ARTHUR RD.	-	LONDON	-	UK
1400	2004 CAROL RD	7645	SEETTLE	WASHINGTON	US
1500	12 CHESTER RD.	8732	WHITEHORSE	YUKON	CA
1600	MARIANO 9991	11932	MEXICO CITY	DISTRITO FEDERAL	MX
1700	CROWN HILL	678	BROWNSWICK	NEW JERSEY	US
1800	123 MARINA RD.	123456	MUMBAI	MAHARASHTRA	IN
1900	29 SEA LANE	3456	SYDNEY	NEW SOUTH WALES	AU

10. Give a 15% increment in salary for all employees.

update employees set salary=salary+salary*0.15;

15 row(s) updated.

11. Add a new employee in the 'Recruiting' department as clerk and set joining date as today's date.

insert into employees
values(115,'JAIDEN','ALEX','JAI@GMAIL.COM',9496131093,SYSDATE,'CLERK',2500,0,0,120);

1 row(s) inserted.

12. Delete *grade_level* value '*F*' in table job_grades and conform the deletion by verifying the remaining rows of the table.

delete from job_grades where grade_level='F';

1 row(s) deleted.

select * from job_grades;

GRADE_LEVEL	LOWEST_SAL	HIGHEST_SAL
A	1000	2999
B	3000	5999
C	6000	9999
D	10000	14999
E	15000	24999

III. TCL Commands:

13. Undo the previous delete operation.

rollback;

Statement processed.

14. Save all the transactions to the database.

Commit;

> Statement processed.

15. Grant the privilege to read and delete from the *employees* table to the User U1.

GRANT UPDATE, TRIGGER ON TABLE employees TO UI;

> Statement processed.

16. Revoke the delete privilege from U1.

REVOKE DELETE ON employees FROM UI;

> Statement processed.

IV. Accessing database (SELECT, Filtering using WHERE, HAVING, GROUP BY, ORDER BY Clauses, Subquery):

17. Display countries along with their regional code.

SELECT country_name,region_id from countries;

COUNTRY_NAME	REGION_ID
AUSTRALIA	3
CANADA	2
CHINA	3
INDIA	1
AMERICA	2
JAPAN	4
ITALY	1
MEXICO	2
UNITED KINGDOM	1

SELECT a.country_name,b.region_name from countries a, regions b where a.region_id=b.region_id;

COUNTRY_NAME	REGION_NAME
AUSTRALIA	zone3
CANADA	zone2
CHINA	zone3
INDIA	zone1
AMERICA	zone2
JAPAN	zone4
ITALY	zone1
MEXICO	zone2
UNITED KINGDOM	zone1

18. List department ID without any repeats from employees table.

select * from students;

DEPARTMENT_ID
50
110
40
90
10
130
20
60
80

19. List the official address and names of all employees in Marketing Department.

SELECT first_name,last_name,street_address from employees,locations where location_id in (select location_id from departments where department_name='marketing')
and department_id in (select department_id from departments where department_name='marketing');

FIRST_NAME	LAST_NAME	STREET_ADDRESS
JOHN	RUSSEL	29 SEA LANE
PETER	TUKER	29 SEA LANE
NEIL	JOSEPH	29 SEA LANE

20. Display first and last name in a single heading *Name*, salary, department ID for those employees whose department is located in the city Mumbai.

SELECT department_id,CONCAT(first_name,last_name) AS Name,salary from employees where department_id in
(select department_id from departments where location_id in (select location_id from locations where city='MUMBAI'));

DEPARTMENT_ID	NAME	SALARY
130	NEENAKOCHHAR	22000
130	LEXDE HAAN	30000
60	NANDITAMOHAN	8000

21. List the total salary spent for each department.

SELECT DISTINCT department_id,sum(salary) OVER(PARTITION BY department_id) Salary from employees;

DEPARTMENT_ID	SALARY
50	15000
20	36000
60	8000
90	28000
10	6000
130	52000
80	10000
40	9000
110	35500

22. List the names of employees in Marketing having salary greater than maximum salary of employees in IT department.

SELECT * FROM employees WHERE salary IN (SELECT salary FROM employees WHERE department_id IN (SELECT d.department_id FROM departments d WHERE d.department_name = 'marketing')) and salary > (SELECT max(salary) FROM employees WHERE department_id IN SELECT d.department_id FROM departments d WHERE d.department_name = 'it'));

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
107	JOHN	RUSSEL	JRUSSEL@GMAIL.COM	6778653490	09-OCT-04	SA_MAN	14000	0	201	20
109	NEIL	JOSEPH	NJOSEPH@GMAIL.COM	7689054321	31-MAR-06	SA_MAN	12000	0	201	20

23. Display country names in upper case and in descending order along with their codes.

SELECT country_id,upper(country_name) from countries order by country_name desc;

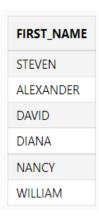| COUNTRY_ID	UPPER(COUNTRY_NAME)
UK	UNITED KINGDOM
MX	MEXICO
JP	JAPAN
IT	ITALY
IN	INDIA
CN	CHINA
CA	CANADA
AU	AUSTRALIA
US	AMERICA

24. Display the full name (first and last), hire date, salary, and department number for those employees whose first name does not containing the letter 'S' and make the result set in ascending order by department number.

SELECT CONCAT(first_name,last_name) AS Name,hire_date,salary,department_id from employees where first_name not LIKE '%S%' order by department_id asc;

NAME	HIRE_DATE	SALARY	DEPARTMENT_ID
WILLIAMJOHN	23-NOV-08	6000	10
JOHNRUSSEL	09-OCT-04	14000	20
NEILJOSEPH	31-MAR-06	12000	20
PETERTUKER	30-JAN-05	10000	20
ANITTABAKER	15-SEP-10	9000	40
NANDITAMOHAN	20-JUL-14	8000	60
JANETTEKING	04-MAR-06	10000	80
NANCYGREENBERG	30-OCT-10	7000	110
DAVIDAUSTIN	10-MAY-07	10000	110
ALEXANDERHUNOLD	03-JAN-06	10500	110
DIANALORENTZ	20-NOV-09	8000	110
NEENAKOCHHAR	21-SEP-05	22000	130
LEXDE HAAN	13-JAN-01	30000	130

25. Identify all employees who work in departments located in the United States.

SELECT first_name from employees where department_id in (select department_id from departments where location_id in (select location_id from locations where country_id in (select country_id from countries where country_name= 'AMERICA')));

FIRST_NAME
STEVEN
ALEXANDER
DAVID
DIANA
NANCY
WILLIAM

26. Display the name (first name and last name) for those employees who gets more salary than the employee whose id is 112.

SELECT CONCAT(first_name,last_name) AS Name,salary from employees where salary > (select salary from employees where employee_id='112');

NAME	SALARY
STEVENKING	28000
NEENAKOCHHAR	22000
LEXDE HAAN	30000
ALEXANDERHUNOLD	10500
DAVIDAUSTIN	10000
JOHNRUSSEL	14000
PETERTUKER	10000
NEILJOSEPH	12000
JANETTEKING	10000
SUSANVARKEY	15000

V. Optimizing databases (Join, Aggregate & Set operations, Other operators like arithmetic, special etc):

27. Use inner join to display the name, department number, and department name for each employee.

SELECT employees.first_name,employees.last_name,departments.department_id,departments.department_name FROM employees INNER JOIN departments ON employees.department_id = departments.department_id;

FIRST_NAME	LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
STEVEN	KING	90	manufacturing
NEENA	KOCHHAR	130	payroll
LEX	DE HAAN	130	payroll
ALEXANDER	HUNOLD	110	it
DAVID	AUSTIN	110	it
DIANA	LORENTZ	110	it
NANCY	GREENBERG	110	it
JOHN	RUSSEL	20	marketing
PETER	TUKER	20	marketing
NEIL	JOSEPH	20	marketing
JANETTE	KING	80	sales
WILLIAM	JOHN	10	administration
ANITTA	BAKER	40	human resource
NANDITA	MOHAN	60	accounting
SUSAN	VARKEY	50	finance

28. List employees who have no job history details using left outer join.

Select employee_name from employees left outer join departments where employee_id is NULL

NULL

29. List department details which have currently no employees using right outer join.

Select department_id,department_id from employees right outer join departments where employee_id is NULL

NULL

30. List all employees who are working in Marketing or Human Resource Department.

SELECT first_name,last_name from employees where department_id in (select department_id from departments where department_name in ('marketing','human resource'));

FIRST_NAME	LAST_NAME
JOHN	RUSSEL
PETER	TUKER
NEIL	JOSEPH
ANITTA	BAKER

31. List the details of employee who draw the highest salary.

SELECT * from employees where salary = (select max(salary) from employees);

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
102	LEX	DE HAAN	LDEHAAN@GMAIL.COM	9554667890	13-JAN-01	AD_VP	30000	0	100	130

32. Find the average salary of employees who joined after 2009.

SELECT avg(salary) FROM employees WHERE hire_date>('01-JAN-09');

AVG(SALARY)
8000

33. Count total number of employees in department with ID 20.

SELECT COUNT(department_id) FROM employees WHERE department_id='20';
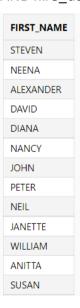
COUNT(DEPARTMENT_ID)
3

34. Find out the person with lowest salary in IT department.

SELECT * from employees where salary = (select min(salary) from employees where department_id in (select department_id from departments where department_name='it'));

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
106	NANCY	GREENBERG	NGREEN@GMAIL.COM	8769054324	30-OCT-10	IT_PROG	7000	0	103	110

35. List all employees who joined in the period 2003-2006 and 2009-2011.

SELECT first_name FROM employees WHERE hire_date BETWEEN ('01-JAN-03') AND ('01-JAN-11') AND hire_date NOT IN ('01-JAN-07','01-JAN-08');

FIRST_NAME
STEVEN
NEENA
ALEXANDER
DAVID
DIANA
NANCY
JOHN
PETER
NEIL
JANETTE
WILLIAM
ANITTA
SUSAN

36. List all employees along with their salary excluding IT programmers.

SELECT * FROM Employees WHERE department_id not in (select department_id from departments WHERE department_name='it');

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
114	SUSAN	VARKEY	SVARKEY@GMAIL.COM	8976054321	14-DEC-04	FIN_MAN	15000	0	108	50
112	ANITTA	BAKER	ABAKER@GMAIL.COM	2345678910	15-SEP-10	HR_REP	9000	0	203	40
100	STEVEN	KING	SKING@GMAIL.COM	8182345210	17-JUN-03	AD_PRES	28000	0	0	90
111	WILLIAM	JOHN	WJOHN@GMAIL.COM	7682315672	23-NOV-08	CLERK	6000	0	200	10
101	NEENA	KOCHHAR	NKOCHHAR@GMAIL.COM	9897561234	21-SEP-05	AD_VP	22000	0	100	130
102	LEX	DE HAAN	LDEHAAN@GMAIL.COM	9554667890	13-JAN-01	AD_VP	30000	0	100	130
107	JOHN	RUSSEL	JRUSSEL@GMAIL.COM	6778653490	09-OCT-04	SA_MAN	14000	0	201	20
108	PETER	TUKER	PTUKER@GMAIL.COM	9067865423	30-JAN-05	SA_MAN	10000	0	201	20
109	NEIL	JOSEPH	NJOSEPH@GMAIL.COM	7689054321	31-MAR-06	SA_MAN	12000	0	201	20
113	NANDITA	MOHAN	NMOHAN@GMAIL.COM	94445634213	20-JUL-14	ACCOUNTANT	8000	0	205	60
110	JANETTE	KING	JKING@GMAIL.COM	9809897645	04-MAR-06	SA_REP	10000	0	145	80

37. Display the name and job_id of employees with their DA amount (DA= 25% of salary).

select first_name,last_name,job_id,salary*.25 as DA from employees;

FIRST_NAME	LAST_NAME	JOB_ID	DA
STEVEN	KING	AD_PRES	7000
NEENA	KOCHHAR	AD_VP	5500
LEX	DE HAAN	AD_VP	7500
ALEXANDER	HUNOLD	IT_PROG	2625
DAVID	AUSTIN	IT_PROG	2500
DIANA	LORENTZ	IT_PROG	2000
NANCY	GREENBERG	IT_PROG	1750
JOHN	RUSSEL	SA_MAN	3500
PETER	TUKER	SA_MAN	2500
NEIL	JOSEPH	SA_MAN	3000
JANETTE	KING	SA_REP	2500
WILLIAM	JOHN	CLERK	1500
ANITTA	BAKER	HR_REP	2250
NANDITA	MOHAN	ACCOUNTANT	2000
SUSAN	VARKEY	FIN_MAN	3750

Download CSV