

Example 1. SQL Basics

Create Table EMP

```
/* Create the first table */
```

```
CREATE TABLE EMP (
    EMP_NO NUMBER(4),
    EMP_NAME VARCHAR2(10) NOT NULL,
    EMP_MGR NUMBER(4),
    EMP_HIREDATE DATE,
    EMP_DEPTNO NUMBER(2)
);
```

Describe Table

```
/* Describe the table structure */
desc emp;
```

Insert data

```
/* Insert the first row on the table EMP */
INSERT INTO EMP
(EMP_NO, EMP_NAME, EMP_MGR, EMP_HIREDATE, EMP_DEPTNO)
VALUES
(7300, 'SMITH', NULL, TO_DATE ('17-DEC-1980', 'DD-MON-YYYY'), 10);
```

Select

```
/* Selecting all the columns */
select * from emp;
```

```
/* Selecting specific columns, also called projection */
select emp_name, emp_deptno from emp;
```

Modify the table structure

```
alter table emp add emp_sal number(7);
/* just to check */
desc emp;
```

Populate the table

```
INSERT INTO EMP (EMP_NO, EMP_NAME, EMP_MGR, EMP_HIREDATE,
EMP_DEPTNO,EMP_SAL)
```

```

VALUES
(7380, 'KING', 7300, TO_DATE ('1-JAN-1982', 'DD-MON-YYYY'), 10, 4500);

INSERT INTO EMP (EMP_NO, EMP_NAME, EMP_MGR, EMP_HIREDATE,
EMP_DEPTNO,EMP_SAL)
VALUES
(7382, 'CLARK', 7300, TO_DATE ('1-JAN-1982', 'DD-MON-YYYY'), 20, 5200);

INSERT INTO EMP (EMP_NO, EMP_NAME, EMP_MGR, EMP_HIREDATE,
EMP_DEPTNO,EMP_SAL)
VALUES
(7383, 'MARTIN', 7382, TO_DATE ('1-FEB-1988', 'DD-MON-YYYY'), 20, 4600);

INSERT INTO EMP (EMP_NO, EMP_NAME, EMP_MGR, EMP_HIREDATE,
EMP_DEPTNO,EMP_SAL)
VALUES
(7385, 'CASEY' , 7382, TO_DATE ('1-FEB-1988', 'DD-MON-YYYY'), 30, 4200);

```

Alias

```
-- note: this is aliasing and not synonym:
select * from emp a where a.emp_name='SMITH';
-- in fact the following will not work
-- select * from a;
```

Example 2. Restricting and Sorting data

WHERE clause

```
select * from EMP WHERE EMP_NAME='SMITH';
```

BETWEEN operator

```
select EMP_NAME || ' earns per month ' || EMP_SAL from EMP
WHERE emp_sal BETWEEN 4500 and 5000;
```

IN operator

```
select EMP_NAME, emp_SAL from EMP WHERE emp_sal IN (3500,4500,5500);
```

LIKE operator

```
select * from EMP WHERE EMP_NAME LIKE 'C%';
```

```
/*with a logical operator*/
select * from EMP WHERE EMP_NAME LIKE 'C%' and emp_deptno=20;
```

ORDER BY clause

```
/*ASC by default*/
select EMP_NAME, EMP_DEPTNO from EMP ORDER BY EMP_DEPTNO;

select EMP_NAME, EMP_DEPTNO from EMP ORDER BY EMP_DEPTNO DESC;
```

Example 3. Constraints

Define Primary Key

```
/* Primary Key = unique + not null */
alter table EMP add constraint EMP_PK primary key (emp_no);
```

Violating PK rule

```
/* try to insert an already existing value */
INSERT INTO EMP (EMP_NO, EMP_NAME, EMP_MGR, EMP_HIREDATE,
EMP_DEPTNO,EMP_SAL)
VALUES
(7380, 'ALLEN', 7382, TO_DATE ('15-JAN-1987', 'DD-MON-YYYY'), 20, 4700);
```

Define Foreign Key (1)

```
/* Note that a FK establishes a relationship */
/* in the same or different table */
alter table EMP add constraint
EMP_MGR_FK foreign key (emp_mgr) references emp(emp_no);
```

Create and Populate DEPT table

```
CREATE TABLE DEPT (
    DEPT_NO NUMBER(4),
    DEPT_NAME VARCHAR2(10),
    DEPT_LOC VARCHAR2(15),
constraint DEPT_PK primary key (DEPT_NO));
```

```
insert into DEPT(DEPT_NO, DEPT_NAME, DEPT_LOC)
values (10,'ACCOUNTING','GENEVA');
```

```
insert into DEPT(DEPT_NO, DEPT_NAME, DEPT_LOC)
values (20,'SALES','ROME');
```

Define Foreign Key (2)

```
/* This will not work because one department is missing in dept table */
alter table EMP add constraint EMP_DEPT_FK
foreign key (emp_deptno)references dept(dept_no);

/* insert into dept the missing department*/
insert into DEPT(DEPT_NO, DEPT_NAME, DEPT_LOC)
values (30,'OPERATIONS','ROME');

/* and now it will work */
```

Example 4. Manipulating data from several tables

Equijoin (1)

```
select e.emp_no, e.emp_name, d.dept_name, d.dept_no
from emp e, dept d
where e.emp_deptno=d.dept_no
```

Equijoin (2)

```
/* Preparation */
insert into DEPT(DEPT_NO, DEPT_NAME, DEPT_LOC)
values (40,'SUPPORT','MILAN');

/* this works*/
update emp set emp_deptno=null where emp_name='CLARK';

/* the following does not work if the FK2 is
enabled and 90 is not a valid dept_no */
-- update emp set emp_deptno=90 where emp_name='CLARK';

/* but if I disable the FK is ok */
-- alter table emp disable constraint EMP_DEPT_FK;
-- update emp set emp_deptno=90 where emp_name='CASEY';

/* equijoin that will show that after the change some users disappeared */
select e.emp_no, e.emp_name, d.dept_name, d.dept_no
from emp e, dept d
```

where e.emp_deptno=d.dept_no

Self Join

```
select e1.emp_name || ' works for ' || e2.emp_name "Employees and their Managers"
from emp e1, emp e2
where e1.emp_mngr=e2.emp_no
```

Left Outer Join

```
/* returns all employees, including those who are not assigned to any department */
select e.emp_no, e.emp_name, d.dept_name, d.dept_no
from emp e, dept d
where e.emp_deptno=d.dept_no(+);
```

Right Outer Join

```
/* returns all departments, including those without any employees */
select e.emp_no, e.emp_name, d.dept_name, d.dept_no
from emp e, dept d
where e.emp_deptno(+)=d.dept_no ;
```

Full Outer Join

```
/* full outer join */
select e.emp_no, e.emp_name, d.dept_name, d.dept_no
from emp e full outer join dept d
on e.emp_deptno=d.dept_no;
```

Example 5. Aggregating

Group Function

```
SELECT COUNT(*) FROM emp;
```

GROUP BY clause

```
select 'Department num. ' || EMP_DEPTNO || ' counts '|| count(*) || ' people'
from emp
group by emp_deptno
```

```
select 'Department num. ' || EMP_DEPTNO || ' named ' || dept_name ||' counts '|| count(*)
|| ' people'
from emp e, dept d
```

```
where e.emp_deptno=d.dept_no  
group by emp_deptno,dept_name
```

HAVING clause

```
/* join the 2 tables and group by the department name */  
/* we just want to have the departments with more than one employees */  
select d.DEPT_NAME, count(*) per_dept  
from emp e, dept d  
where e.EMP_DEPTNO = d.DEPT_NO  
group by d.DEPT_NAME  
having count(*) > 1
```

SQL functions

```
/* || is the concatenation operator */  
/* using numeric function 'trunc' and date function 'months between' */  
select e.EMP_NAME||' works in '|d.DEPT_NAME dept,  
'He has been working in the company for '|  
trunc(months_between(sysdate,EMP_HIREDATE)/12) || ' years' working  
from emp e, dept d  
where e.EMP_DEPTNO = d.DEPT_NO
```