

## DEPT

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

## BONUS

ENAME	JOB	SAL	COMM
SMITH	CLERK	800.000	
ALLEN	SALESMAN	1.600.000	300.000
WARD	SALESMAN	1.250.000	500.000
JONES	MANAGER	2.975.000	
MARTIN	SALESMAN	1.250.000	1.400.000

## SALGRADE

GRADE	LOSAL	HISAL
1	700	1200
2	1201	1400
3	1401	2000
4	2001	3000
5	3001	9999

## EMP

EMPNO	ENAME	JOB	MGR	HIREDATE	SALARY	COMM	DEPTNO
7329	SMITH	CLERK	7902	17-DEC-80	800.0	300.00	20
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600.00	300.00	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250.00	500.00	30
7566	JONES	MANAGER	7839	02-APR-81	2975.00		20
7654	MARTIN	SALESMAN	7698	20-SEP-81	1250.00	1400.00	30
7698	BLAKE	MANAGER	7830	01-MAY-81	2850.00		30
7782	CLARK	MANAGER	7839	09-JUNE-81	2450.00		10
7788	SCOTT	ANALYST	7566	09-DEC-82	3000.00		20
7839	KING	PRESIDENT		17-NOV-81	5000.00		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500.00	0.00	30
7876	ADAMS	CLERK	7788	12-JUNE-83	1100.00		20
7900	JAMES	CLERK	7698	03-DEC-81	950.00		30
7902	FORD	ANALYST	7566	03-DEC-81	3000.00		20
7934	MILLER	CLERK	7782	23-JUNE-82	1300.00		10

# SQL

(la sintassi presentata e' quella del DBMS ORACLE; altri DBMS possono presentare lievi differenze)

**linguaggio standard  
per l'interrogazione e la manipolazione  
dei dati  
nei sistemi relazionali**

## ALGEBRA RELAZIONALE      SQL

---

RELAZIONE

TABLE

ATTRIBUTO

COLUMN

TUPLA

ROW

DOMINIO

DATA TYPE

espressionePSJ

comando SELECT

$\pi_x(\sigma_C(R_1 \times R_2 \times \dots \times R_n))$

SELECT X  
FROM  $R_1, R_2, \dots, R_n$   
WHERE C

## COMANDI BASE SQL

SELECT

ricerca

INSERT

DELETE

modifica

UPDATE

CREATE TABLE

creazione

```
SELECT  ENAME, JOB, DEPTNO
FROM    EMP
WHERE   JOB='MANAGER;
```

```
INSERT INTO    EMP
VALUES         (7954,'CARTER','CLERK',
               7698, '7-APR-84, 100,
               NULL, 30);
```

```
UPDATE  EMP
SET     MGR=7566, DEPTNO=20
WHERE   ENAME='CARTER;
```

```
DELETE FROM    EMP
WHERE          ENAME='CARTER  AND
               EMPNO=7954;
```

```
CREATE TABLE  DEPT (DEPTNO NUMBER(2),
                     DNAME CHAR(14), LOC
                     CHAR(13));
```

## TIPO DI DATI

CHAR(a)	$a \leq 240$
NUMBER	40 cifre
NUMBER(a)	$a \leq 105$ (40 cifre significative)
NUMBER(a,d)	d cifre decimali
DATE	(1 gen 4712 a.C.) – (31 dic 4712 d.C.)
LONG	ampiezza $\leq 65535$ (un solo attributo per tabella)

## COMANDO SELECT

SELECT    **clausola\_SELECT**

FROM      **clausola\_FROM**

WHERE    **clausola\_WHERE**



## COMANDO SELECT (OPZIONI sulla clausola\_SELECT)

### Opzione DISTINCT

```
SELECT  JOB  
FROM    EMP;
```

```
SELECT DISTINCT  JOB  
FROM              EMP;
```

### Opzione ALIAS

```
SELECT  DNAME DIPARTIMENTO DEPTNO  
FROM    DEPT;
```

## COMANDO SELECT (clausola WHERE)

### Relazioni di CONFRONTO

```
SELECT  *  
FROM    EMP  
WHERE   JOB='CLERK';
```

### Relazioni di APPARTENENZA INSIEMISTICA

```
SELECT  ENAME, JOB, SAL  
FROM    EMP  
WHERE   SAL BETWEEN 1200 AND 1300;
```

```
SELECT  ENAME, JOB, SAL  
FROM    EMP  
WHERE   JOB IN ('CLERK','ANALYST');
```

### Relazioni di SOMIGLIANZA ORTOGRAFICA

```
SELECT  ENAME, JOB, SAL  
FROM    EMP  
WHERE   ENAME LIKE 'M%'
```

## COMANDO SELECT (Opzione ORDER)

### Ordine ASCENDENTE

```
SELECT      ENAME,JOB,SAL  
FROM        EMP  
WHERE       DEPTNO=30  
ORDER BY    SAL;
```

### Ordine DISCENDENTE

```
SELECT      ENAME,JOB,SAL  
FROM        EMP  
WHERE       DEPTNO=30  
ORDER BY    SAL DESC;
```

### Ordine MULTIPLO

```
SELECT      ENAME,JOB,SAL  
FROM        EMP  
WHERE       DEPTNO=30  
ORDER BY    JOB, SAL DESC;
```

**Valori nulli:** se l'ordinamento viene fatto su una colonna che contiene valori nulli, questi vengono sempre listati per primi

## COMANDO SELECT

(Join)

```
SELECT  ENAME, LOC
FROM    EMP, DEPT
WHERE   ENAME='ALLEN' AND
        EMP.DEPTNO=DEPT.DEPTNO
```

Opzione ALIAS nella clausola FROM

```
SELECT  ENAME, LOC
FROM    EMP E, DEPT D
WHERE   ENAME='ALLEN' AND
        E.DEPTNO=D.DEPTNO
```

```
SELECT  D.*, ENAME, JOB
FROM    EMP E, DEPT D
WHERE   E.DEPTNO=D.DEPTNO AND
        E.DEPTNO IN (30,40)

ORDER BY E.DEPTNO
```

## COMANDO SELECT (Non Equijoin)

```
SELECT  X.ENAME, X.SAL, X.JOB,  
        Y.ENAME, Y.SAL, Y.JOB  
FROM    EMP X, EMP Y  
WHERE   X.SAL > Y.SAL AND Y.NAME  
        ='JONES';
```

```
SELECT  DIP.ENAME, DIP.SAL,  
        DIR.ENAME, DIR.SAL  
FROM    EMP DIP, EMP DIR  
WHERE   DIR.MGR=DIR.EMPNO AND  
        DIP.SAL>DIR.SAL;
```

## DOMANDE SUBORDINATE

```
SELECT SAL, JOB, ENAME, DEPTNO
FROM EMP
WHERE SAL > ANY
      (oppure ALL)
      (SELECT SAL
       FROM EMP
       WHERE DEPTNO=30);
```

```
SELECT JOB, ENAME, EMPNO, EMPTNO
FROM EMP X
WHERE EXISTS
      (SELECT *
       FROM EMP
       WHERE X.EMPNO=MGR);
```

## DOMANDE COORDINATE

```
SELECT  SAL  
FROM    EMP  
WHERE   ENAME='SCOTT'
```

UNION (oppure INTERSECT oppure MINUS)

```
SELECT  SAL  
FROM    EMP  
WHERE   ENAME='WARD';
```

## DATI NUMERICI (Funzioni d'insieme)

AVG  
COUNT  
MAX  
MIN  
SUM

```
SELECT  MAX(SAL) MIN(SAL)  
        MAX (SAL)-MIN(SAL)  
FROM    EMP;
```

output

MAX(SAL)	MIN(SAL)	MAX(SAL)-MIN(SAL)
5000	800	4200



## DATI NUMERICI (Statistiche)

```
SELECT COUNT(DISTINCT JOB)
FROM EMP
WHERE DEPTNO=30;
```

output

COUNT(DISTINCTJOB)
3

```
SELECT COUNT(*)
FROM EMP
WHERE DEPTNO=30;
```

output

COUNT(*)
6

## COMANDO INSERT

```
INSERT INTO EMP  
VALUES (7954, 'CARTER', 'CLERK', 7698, '7-  
APR-84', 1000, NULL, 30);
```

```
INSERT INTO EMP (EMPNO, ENAME, DEPTNO,  
SAL)  
VALUES (7955, 'WILSON', 30, 1500);
```

```
INSERT INTO BONUS (ENAME, JOB, SAL,  
COMM)  
SELECT ENAME, JOB, SAL, COMM  
FROM EMP  
WHERE JOB='MANAGER' OR COMM >  
0.25*SAL;
```

## COMANDO UPDATE

```
UPDATE EMP
SET     JOB='SALSMAN, SAL=1.1*SAL
WHERE  ENAME='WILSON';
```

```
UPDATE EMP
SET     SAL= 1.05*SAL
WHERE  ENAME IN
      (SELECT ENAME
       FROM   BONUS);
```

```
UPDATE EMP
SET     SAL=
      (SELECT      1.1*AVG(SAL)
       FROM        EMP
       WHERE       JOB='SALSMAN')
WHERE  JOB='SALSMAN';
```

## COMANDO DELETE

```
DELETE FROM BONUS  
WHERE ENAME='WARD';
```

```
DELETE FROM BONUS  
WHERE JOB IN  
(SELECT JOB  
FROM BONUS  
WHERE ENAME='WARD');
```

## COMANDO CREATE

```
CREATE TABLE DEPT (DEPTNO NUMBER(2),  
DNAME CHAR(14), LOC  
CHAR(13));
```

```
CREATE TABLE BONUS (ENAME, JOB,  
SAL, COMM)  
AS SELECT ENAME, JOB, SAL, COMM  
FROM EMP  
WHERE JOB = 'MANAGER' OR COMM >  
0.25*SAL;
```

## COMANDO ALTER

Per allargare un campo

```
ALTER TABLE PROJ  
MODIFY          (BUDGET NUMBER (9,2) NOT  
                NULL);
```

Per aggiungere un campo

```
ALTER TABLE EMP  
ADD          (PROJNO NUMBER);
```

## COMANDO DROP

DROP TABLE **BONUS**;

## COMANDO RENAME

RENAME **EMP** TO **IMPIEGATI**;

## VEDUTE (tabelle virtuali)

```
CREATE VIEW EMP10
AS SELECT EMPNO, ENAME, JOB
FROM EMP
WHERE DEPTNO = 10;
```

```
SELECT ENAME, JOB
FROM EMP10
WHERE EMPNO > 7800;
```

```
CREATE VIEW MIO_DIP
AS SELECT *
FROM EMP
WHERE DEPTNO IN
(SELECT DEPTNO
FROM EMP
WHERE ENAME=USER AND
JOB='MANAGER');
```

```
DROP VIEW EMP10;
```



## DIRITTI D'USO

### Comando GRANT

```
GRANT  [modalità d'uso]  
ON     [oggetto]  
TO     [utente];
```

modalità d'uso	oggetto
SELECT	(dati in) tabella o veduta
INSERT veduta	(righe in) tabella o
UPDATE	(dati in) tabella o veduta
DELETE	righe da) tabella o veduta
ALTERA	(attributi in) tabella
INDEX	indice di tabella

## DIRITTI D'USO

```
GRANT    SELECT
ON      DEPT
TO      ADAMS;
```

```
GRANT    ALL
ON      DEPT
TO      ADAMS, JONES;
```

```
GRANT    SELECT
ON      DEPT
TO      PUBLIC;
```

```
GRANT    SELECT
ON      EMP
TO      ADAMS
WITH    GRANT OPTION;
```

```
GRANT    SELECT
ON      MIO_DIP
TO      PUBLIC;
```

su una veduta

## DIRITTI D'USO

### Comando REVOKE

REVOKE [modalità d'uso]  
ON [oggetto]  
TO [utente]

REVOKE ALL  
ON DEPT  
FROM ADAMS;

```
CREATE VIEW V_STUD_INF AS
SELECT *
FROM STUDENTE
WHERE C_LAUREA = 'INFORMATICA';
```

```
CREATE TABLE T_STUD_INF AS
SELECT *
FROM STUDENTE
WHERE C_LAUREA = 'INFORMATICA';
```

```
SELECT *
FROM STUDENTE
```

STUDENTE

NOME	MATR	C_LAUREA
Rossi	01	INFORMATICA
Verdi	02	MATEMATICA

```
SELECT *
FROM V_STUD_INF
```

V\_STUD\_INF

NOME	MATR	C_LAUREA
Rossi	01	INFORMATICA

```
SELECT *
FROM T_STUD_INF
```

T\_STUD\_INF

NOME	MATR	C_LAUREA
Rossi	01	INFORMATICA

```
INSERT INTO STUDENTE  
VALUES ('Bianchi', '03', 'INFORMATICA');
```

```
SELECT *  
FROM STUDENTE;
```

STUDENTE

NOME	MATR	C_LAUREA
Rossi	01	INFORMATICA
Verdi	02	MATEMATICA
Bianchi	03	INFORMATICA

```
SELECT *  
FROM V_STUD_INF
```

V\_STUD\_INF

NOME	MATR	C_LAUREA
Rossi	01	INFORMATICA
Bianchi	03	INFORMATICA

```
SELECT *  
FROM T_STUD_INF
```

T\_STUD\_INF

NOME	MATR	C_LAUREA
Rossi	01	INFORMATICA