



Cofinanziato  
dall'Unione europea



REGIONE  
PIEMONTE

# FinTech Software Developer

## Basi di dati SQL

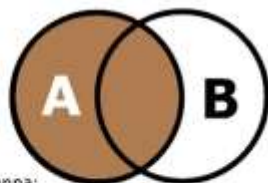
Docente: Roi Davide Simone

Titolo argomento: Sviluppo su Postgresql – Costrutti di querying  
JOIN MULTIPLE, SELECT ANNIDATE, UNION

Roi Davide  
Dispense

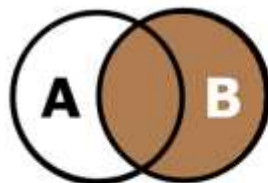


```
SELECT
<list>
FROM
tabella1 AS a
LEFT JOIN
tabella2 AS b ON
a.colonna = b.colonna;
```



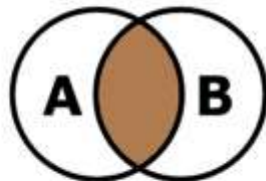
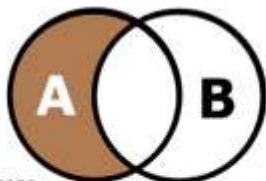
## Join

Roi Davide  
Dispense



```
SELECT
<list>
FROM
tabella1 AS a
RIGHT JOIN
tabella2 AS b ON
a.colonna = b.colonna;
```

```
SELECT
<list>
FROM
tabella1 AS a
LEFT JOIN
tabella2 AS b ON
a.colonna = b.colonna
WHERE
b.colonna IS NULL;
```

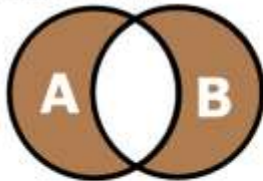
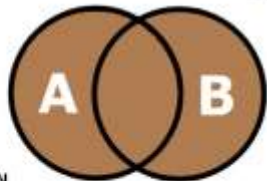


```
SELECT
<list>
FROM
tabella1 AS a
INNER JOIN
tabella2 AS b ON
a.colonna = b.colonna;
```



```
SELECT
<list>
FROM
tabella1 AS a
RIGHT JOIN
tabella2 AS b ON
a.colonna = b.colonna
WHERE
a.colonna IS NULL;
```

```
SELECT
<list>
FROM
tabella1 AS a
FULL JOIN
tabella2 AS b ON
a.colonna = b.colonna;
```



```
SELECT
<list>
FROM
tabella1 AS a
FULL JOIN
tabella2 AS b ON
a.colonna = b.colonna
WHERE
a.colonna IS NULL
OR b.colonna IS NULL;
```

# JOIN Multiple

Come si interpretano le JOIN multiple sui recordset del costrutto FROM

```
SELECT
FROM
  recordset_1 AS rs_1

[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_2 AS rs_2 ON
  rs_1.col = rs_2.col
```

```
[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_3 AS rs_3 ON
  rs_1|2 = rs_3.col
```

```
[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_4 AS rs_4 ON
  rs_1|2|3.col = rs_4.col
```

```
[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_5 AS rs_5 ON
  rs_1|2|3|4.col = rs_5.col
```

```
SELECT
FROM
  recordset_1 AS rs_1

[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_2 AS rs_2 ON
  rs_1.col = rs_2.col

[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_3 AS rs_3 ON
  rs_1|2 = rs_3.col
```

```
[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_4 AS rs_4 ON
  rs_1|2|3.col = rs_4.col
```

```
[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_5 AS rs_5 ON
  rs_1|2|3|4.col = rs_5.col
```

```
SELECT
FROM
  recordset_1 AS rs_1

[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_2 AS rs_2 ON
  rs_1.col = rs_2.col

[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_3 AS rs_3 ON
  rs_1|2 = rs_3.col
```

```
[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_4 AS rs_4 ON
  rs_1|2|3.col = rs_4.col
```

```
[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_5 AS rs_5 ON
  rs_1|2|3|4.col = rs_5.col
```

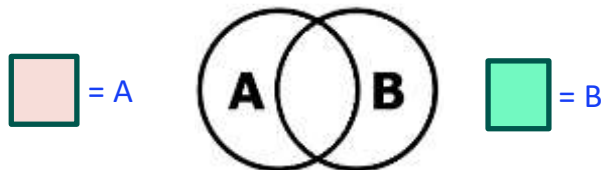
```
SELECT
FROM
  recordset_1 AS rs_1

[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_2 AS rs_2 ON
  rs_1.col = rs_2.col

[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_3 AS rs_3 ON
  rs_1|2 = rs_3.col
```

```
[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_4 AS rs_4 ON
  rs_1|2|3.col = rs_4.col
```

```
[INNER/LEFT/RIGHT/FULL] JOIN
  recordset_5 AS rs_5 ON
  rs_1|2|3|4.col = rs_5.col
```



# JOIN Multiple

Come si interpretano le JOIN multiple sui recordset del costrutto FROM

SELECT

FROM

**recordset\_a AS rs\_a**

[INNER/LEFT/RIGHT/FULL] JOIN

**recordset\_b AS rs\_b** ON

rs\_a.col = rs\_b.col

-- STEP1: Tramite JOIN aggiungo a rs\_a i record di rs\_b  
-- ottenendo un nuovo recordset che è l'unione dei due  
-- (usando come chiave i campi di rs\_a)

[INNER/LEFT/RIGHT/FULL] JOIN

**recordset\_c AS rs\_c** ON

[rs\_a.col|rs\_b.col] = rs\_c.col

-- STEP2: Tramite JOIN aggiungo al recordset ottenuto precedentemente i record di rs\_c  
-- ottenendo un nuovo recordset che è l'unione dei due  
-- (usando come chiave i campi di rs\_a oppure rs\_b **ovvero i campi del recordset precedente**)

[INNER/LEFT/RIGHT/FULL] JOIN

**recordset\_d AS rs\_d** ON

[rs\_a.col|rs\_b.col|rs\_c.col] = rs\_d.col -- (usando come chiave i campi di rs\_a oppure rs\_b oppure rs\_c **ovvero i campi del recordset precedente**)

[INNER/LEFT/RIGHT/FULL] JOIN

ecc...

N.B: Ricordate sempre che i recordset possono essere: TABELLE, VISTE, SELECT ANNIDATE

# JOIN Multiple

Come si interpretano le JOIN multiple sui recordset del costrutto FROM

A.anno_fondazione	A.ragione_sociale	A.id
1800	Banco di Napoli	1
2015	FintechLab	2
1823	Intesa Sanpaolo	3
2011	Moneyfarm	4
2013	N26	5



B.id_azienza	B.nome_invest	B.id
3	Fondo comune	11
4	Obbligazioni	22
5	Azioni	33
6	Cambio valuta	44
7	Derivati	55

STEP1: Una JOIN tra 2 recordset produce come risultato un recordset che contiene l'insieme delle colonne dei due recordset uniti.

Risultato:

A.anno_fondazione	A.ragione_sociale	A.id	B.id_azienza	B.nome_invest	B.id
1800	Banco di Napoli	1	NULL	NULL	NULL
2015	FintechLab	2	NULL	NULL	NULL
1823	Intesa Sanpaolo	3	3	Fondo comune	11
2011	Moneyfarm	4	4	Obbligazioni	22
2013	N26	5	5	Azioni	33

Ad Esempio:  
1 recordset di 3 colonne  
JOIN  
1 recordset di 3 colonne  
=  
1 recordset di 6 colonne

N.B: Ricordate sempre che i recordset possono essere: TABELLE, VISTE, SELECT ANNIDATE

# JOIN Multiple

Come si interpretano le JOIN multiple sui recordset del costrutto FROM

A.anno_fondazione	A.ragione_sociale	A.id	B.id_azien da	B.nome	B.id
1800	Banco di Napoli	1	NULL	NULL	NULL
2015	FintechLab	2	NULL	NULL	NULL
1823	Intesa Sanpaolo	3	3	Fondo comune	11
2011	Moneyfarm	4	4	Obbligazioni	22
2013	N26	5	5	Azioni	33



C.id_invest	C.descr_roi	C.valore	C.i d
11	recensione_1	10	41
22	recensione_2	15.3	42
33	recensione_3	30.2	43
44	recensione_4	12	45

STEP2: Dal risultato precedente faccio un'altra JOIN con un recordset di 4 COLONNE, questo produce come risultato un recordset di 10 colonne. Ecc...

Risultato:

A.anno_fondazione	A.ragione_sociale	A.id	B.id_azienza	B.nome	B.id	C.id_invest	C.descr	C.valore	C.id
1800	Banco di Napoli	1	NULL	NULL	NULL	NULL	NULL	NULL	NULL
2015	FintechLab	2	NULL	NULL	NULL	NULL	NULL	NULL	NULL
1823	Intesa Sanpaolo	3	3	Fondo comune	11	11	recensione_1	10	41
2011	Moneyfarm	4	4	Obbligazioni	22	22	recensione_2	15.3	42
2013	N26	5	5	Azioni	33	33	recensione_3	30.2	43

N.B: Ricordate sempre che i recordset possono essere: TABELLE, VISTE, SELECT ANNIDATE

# SELECT Annidate

Le select annidate sono delle intere QUERY racchiuse tra () e inserite nel costrutto del FROM/WHERE/HAVING.

## SELECT annidate nel costrutto FROM

```
select
  ...
from
  ...
where
  ...
;
```

```
select
  ...
from
  (
    select
      ...
    from
      ...
    where
      ...
  ) as alias_1
where
  ...
;
```

```
select
  ...
from
  (
    select
      ...
    from
      (
        select
          ...
        from
          ...
        ) as alias_2
    where
      ...
  ) as alias_1
where
  ...
;
```

## SELECT annidate nei costrutti WHERE / HAVING

```
select
  ...
from
  t1
where | having
  t1.colonna IN
```

```
(
  select
    t2.colonna
  from
    t2
  where
    ...
)
```

```
select
  ...
from
  t1
where | having
  t1.colonna IN
  (
    select
      t2.colonna
    from
      t2
    group by
      ...
    having
      max(t2.colonna3) IN
      (
        select
          t3.colonna
        from
          t3
        )
    )
;
```

Fine della Presentazione