

31. ALL e ANY

database

ALL e ANY

Abbiamo visto come la chiusura del linguaggio SQL ci consenta di sfruttare il risultato di una subquery come argomento della clausola **WHERE** realizzando query nidificate.

Comunque, quando nella clausola `WHERE` abbiamo da un lato il valore di un attributo e dall'altro il risultato di una subquery, essendo quest'ultimo un insieme di righe, non possono essere direttamente confrontati tra loro.

Piuttosto possiamo verificare la presenza del valore dell'attributo nell'insieme di valori prodotti dalla subquery mediante l'operatore `IN` (o `NOT IN`, per verificarne l'assenza).

ALL e ANY

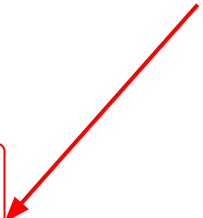
Ad es. la query seguente, che estrae i dipendenti che lavorano in dipartimenti ubicati nella città di Catanzaro:

```
SELECT *  
FROM Personale AS P, Dipartimenti AS D  
WHERE P.CodDip = D.CodDip  
      AND D.Citta = 'Catanzaro';
```

può essere riscritta come segue:

```
SELECT *  
FROM Personale AS P, Dipartimenti AS D  
WHERE P.CodDip = D.CodDip  
      AND P.CodDip IN (SELECT CodDip  
                        FROM Dipartimenti  
                        WHERE Citta = 'Catanzaro');
```

Restituisce un insieme dei codici dei dipartimenti
ubicati nella città di Catanzaro



ALL e ANY

Oltre agli operatori `IN` e `NOT IN`, SQL consente di estendere, con le parole chiave **ALL** o **ANY** i normali operatori di confronto (`=`, `<`, `>`, `<>`, `>=`, `<=`), che permettono di verificare rispettivamente se una certa relazione vale per tutti i valori, o per almeno un valore dell'insieme prodotto dalla subquery nidificata.

```
SELECT *  
FROM Personale AS P, Dipartimenti AS D  
WHERE P.CodDip = D.CodDip  
      AND P.CodDip = ANY (SELECT CodDip  
                          FROM Dipartimenti  
                          WHERE Citta = 'Catanzaro');
```

La parola chiave **ANY**, ad es. specifica che la riga soddisfa la condizione se risulta vero il confronto (in funzione dell'operatore di confronto usato) tra il valore dell'attributo per la riga e almeno uno (*any* = **uno qualunque**) degli elementi restituiti dall'interrogazione.

ALL e ANY

Riprendiamo l'esempio già visto in precedenza.

Dato che ci sono due dipartimenti ubicati nella città di Catanzaro (Amministrazione e Logistica) nei quali lavorano in tutto otto dipendenti, la query vista prima fornisce come risultato proprio questi dipendenti.

+ Opzioni

← T →			Matricola	CodDip	Cognome	Nome	DataNasc	Qualifica	Stipendio	Ufficio	Citta	
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	AB001	D1	Rossi	Mario	1989-02-01	Tecnico	25000.00	10	Catanzaro
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	AB002	D1	Bianchi	Carlo	1992-12-12	Programmatore	32400.00	20	Cosenza
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	ZZ001	D2	Palmi	Pedi	1993-04-21	Analista	NULL	14	Catanzaro
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	LM003	D2	Segni	Carlo	1998-04-14	Programmatore	28800.00	7	Napoli
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	CD003	D2	Rossi	Carlo	2001-04-01	Impiegato	22000.00	14	Cosenza
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	CD002	D2	Verdi	Franco	1996-09-02	Tecnico	26000.00	16	Catanzaro
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	CD001	D2	Verdi	Giovanni	2000-11-07	Analista	28000.00	20	Reggio Calabria
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	EF001	D3	Gialli	Lorenzo	1982-10-05	Impiegato	23000.00	7	Reggio Calabria
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	EF003	D3	Franco	Marco	1999-07-03	Programmatore	30000.00	20	Catanzaro
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	EF002	D3	Rosati	Paola	1995-08-13	Impiegato	24000.00	75	Crotone
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	NP002	D4	Voci	Claudio	1999-03-21	Impiegato	18500.00	20	Messina
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	NP001	D4	Gessi	Giovanni	2003-04-13	Tecnico	25000.00	10	Catanzaro
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	LM002	D4	Vacchi	Marco	1996-04-03	Programmatore	35000.00	16	Cosenza
<input type="checkbox"/>	Modifica	Copia	Elimina	LM001	D4	Ponsi	Gianni	2002-04-09	Analista	38000.00	7	Cosenza
<input type="checkbox"/>	Modifica	Copia	Elimina	GH001	D4	Tedesco	Matteo	2001-06-13	Programmatore	32000.00	12	Cosenza
<input type="checkbox"/>	Modifica	Copia	Elimina	ZZ002	D4	Ponsi	Mario	1994-05-11	Programmatore	38000.00	7	Reggio Calabria

				CodDip	Nome	Indirizzo	Città
<input type="checkbox"/>	Modifica	Copia	Elimina	D1	Amministrazione	Via Roma, 34	Catanzaro
<input type="checkbox"/>	Modifica	Copia	Elimina	D2	Ricerca	Via Napoli, 53	Cosenza
<input type="checkbox"/>	Modifica	Copia	Elimina	D3	Produzione	Via Palermo, 22	Reggio Calabria
<input type="checkbox"/>	Modifica	Copia	Elimina	D4	Logistica	Via Crotone, 12	Catanzaro

ALL e ANY

Analizziamo la query:

```
SELECT *  
FROM Personale AS P, Dipartimenti AS D  
WHERE P.CodDip = D.CodDip  
      AND P.CodDip = ANY (SELECT CodDip  
                          FROM Dipartimenti  
                          WHERE Citta = 'Catanzaro');
```

Come detto prima, la subquery (interna) restituisce come risultato una tabella con i soli codici dei dipartimenti ubicati nella città di Catanzaro (D1 e D4).

+ Opzioni

					CodDip
<input type="checkbox"/>		Modifica		Copia	Elimina D1
<input type="checkbox"/>		Modifica		Copia	Elimina D4

La query esterna seleziona dalla tabella risultante dal join tra *Personale* e *Dipartimenti*, le sole righe in cui il codice del dipartimento risulta **uguale ad uno qualunque** dei codici contenuti nella tabella restituita dalla subquery.

In pratica confronta il campo *CodDip* di ogni riga con D4 e D1, selezionando solo le righe che presentano tali valori.

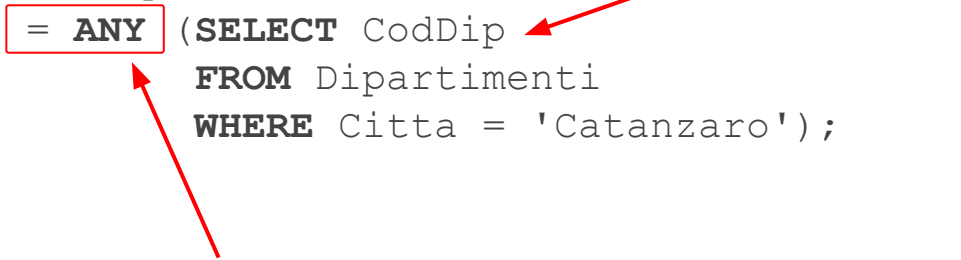
+ Opzioni

Matricola	CodDip	Cognome	Nome	DataNasc	Qualifica	Stipendio	Ufficio	Citta	CodDip	Nome
AB001	D1	Rossi	Mario	1989-02-01	Tecnico	25000.00	10	Catanzaro	D1	Ammin
AB002	D1	Bianchi	Carlo	1992-12-12	Programmatore	32400.00	20	Cosenza	D1	Ammin
GH001	D4	Tedesco	Matteo	2001-06-13	Programmatore	32000.00	12	Cosenza	D4	Logistic
LM001	D4	Ponsi	Gianni	2002-04-09	Analista	38000.00	7	Cosenza	D4	Logistic
LM002	D4	Vacchi	Marco	1996-04-03	Programmatore	35000.00	16	Cosenza	D4	Logistic
NP001	D4	Gessi	Giovanni	2003-04-13	Tecnico	25000.00	10	Catanzaro	D4	Logistic
NP002	D4	Voci	Claudio	1999-03-21	Impiegato	18500.00	20	Messina	D4	Logistic
ZZ002	D4	Ponsi	Mario	1994-05-11	Programmatore	38000.00	7	Reggio Calabria	D4	Logistic

ALL e ANY

Analizziamo la query:

```
SELECT *  
FROM Personale AS P, Dipartimenti AS D  
WHERE P.CodDip = D.CodDip  
      AND P.CodDip = ANY (SELECT CodDip  
                          FROM Dipartimenti  
                          WHERE Citta = 'Catanzaro');
```



Naturalmente è necessario indicare un solo campo nella clausola **SELECT** della subquery.

NB: “= **ANY**” è equivalente all’uso dell’operatore **IN**, ma più potente in quanto, invece di verificare semplicemente l’uguaglianza, è possibile sostituire l’operatore “=” con qualunque altro operatore di confronto (>, <, <>, >=, <=).

ALL e ANY

Vogliamo ora conoscere il nome di tutti i dipartimenti in cui esiste almeno un dipendente con stipendio maggiore di € 35.000,00:

Possiamo provare con la seguente query :

```
SELECT Nome
FROM Dipartimenti
WHERE CodDip = ANY (SELECT CodDip
                     FROM Personale
                     WHERE Stipendio >= 35000)
```

La query esterna seleziona dalla tabella *Dipartimenti* le righe in cui il codice del dipartimento sia **uguale ad uno qualunque** dei codici contenuti nella tabella restituita dalla subquery.

La subquery (interna), partendo dalla tabella *Personale*, restituisce come risultato una tabella con i codici dei dipartimenti nei quali lavorano i dipendenti con stipendio superiore a 32.000 euro.

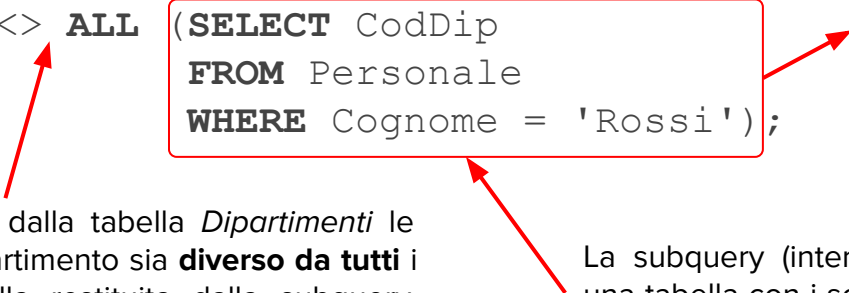
+ Opzioni	
←T→	CodDip
<input type="checkbox"/>	<input type="checkbox"/> Modifica <input type="checkbox"/> Copia <input type="checkbox"/> Elimina D1
<input type="checkbox"/>	<input type="checkbox"/> Modifica <input type="checkbox"/> Copia <input type="checkbox"/> Elimina D4
+ Opzioni	
<input type="checkbox"/>	←T→
<input type="checkbox"/>	<input type="checkbox"/> Nome
<input type="checkbox"/>	<input type="checkbox"/> Modifica <input type="checkbox"/> Copia <input type="checkbox"/> Elimina Amministrazione
<input type="checkbox"/>	<input type="checkbox"/> Modifica <input type="checkbox"/> Copia <input type="checkbox"/> Elimina Logistica

ALL e ANY

SQL mette anche a disposizione l'operatore **ALL** per specificare che la riga soddisfa la condizione indicata dall'operatore di confronto utilizzato, solo se tutti gli elementi restituiti dalla subquery nidificata (*all* = **tutti**) rendono vero il confronto.

Vogliamo estrarre i dipartimenti in cui non lavorano persone con cognome Rossi:

```
SELECT Nome
FROM Dipartimenti
WHERE CodDip <> ALL (SELECT CodDip
                     FROM Personale
                     WHERE Cognome = 'Rossi');
```



La query esterna seleziona dalla tabella *Dipartimenti* le righe in cui il codice del dipartimento sia **diverso da tutti** i codici contenuti nella tabella restituita dalla subquery, quindi ne proietta solo il nome.

+ Opzioni

←T→

CodDip

☐

Modifica

Copia

Elimina

D1

+ Opzioni

←T→

Nome

☐

Modifica

Copia

Elimina

Produzione

☐

Modifica

Copia

Elimina

Logistica

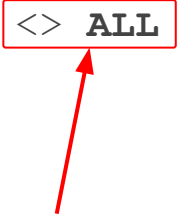
La subquery (interna) restituisce come risultato una tabella con i soli codici dei dipartimenti in cui lavorano persone di cognome 'Rossi' (D1 e D2).

ALL e ANY

SQL mette anche a disposizione l'operatore **ALL** per specificare che la riga soddisfa la condizione indicata dall'operatore di confronto utilizzato, solo se tutti gli elementi restituiti dalla subquery nidificata (*all* = **tutti**) rendono vero il confronto.

Vogliamo estrarre i dipartimenti in cui non lavorano persone di cognome Rossi:

```
SELECT Nome
FROM Dipartimenti
WHERE CodDip <> ALL (SELECT CodDip
                     FROM Personale
                     WHERE Cognome = 'Rossi');
```



NB: Anche in questo caso, possiamo notare come “<> **ALL**” sia equivalente all’uso dell’operatore **NOT IN**.

ALL e ANY

Se vogliamo conoscere matricola, nominativo e data di nascita del dipendente più anziano, possiamo ragionare come segue:

```
SELECT Matricola, Nome, Cognome, DataNasc
FROM Personale
WHERE DataNasc <= ALL (SELECT DataNasc
                        FROM Personale);
```

La query esterna seleziona dalla tabella *Personale* le righe in cui la data di nascita sia **minore o uguale rispetto a tutte** le date di nascita contenute nella tabella restituita dalla subquery, quindi ne proietta solo matricola, nome e cognome.

La subquery (interna) restituisce come risultato una tabella con le date di nascita dei dipendenti.

				DataNasc
←T→				
<input type="checkbox"/>	Modifica	Copia	Elimina	1989-02-01
<input type="checkbox"/>	Modifica	Copia	Elimina	1992-12-12
<input type="checkbox"/>	Modifica	Copia	Elimina	2000-11-07
<input type="checkbox"/>	Modifica	Copia	Elimina	1996-09-02
<input type="checkbox"/>	Modifica	Copia	Elimina	2001-04-01
<input type="checkbox"/>	Modifica	Copia	Elimina	1982-10-05
<input type="checkbox"/>	Modifica	Copia	Elimina	1995-08-13
<input type="checkbox"/>	Modifica	Copia	Elimina	1999-07-03

				Matricola	Nome	Cognome	DataNasc
←T→							
<input type="checkbox"/>	Modifica	Copia	Elimina	EF001	Lorenzo	Gialli	1982-10-05
<input type="checkbox"/>	Modifica	Copia	Elimina				
<input type="checkbox"/>	Modifica	Copia	Elimina				1998-04-14
<input type="checkbox"/>	Modifica	Copia	Elimina				2003-04-13
<input type="checkbox"/>	Modifica	Copia	Elimina				1999-03-21
<input type="checkbox"/>	Modifica	Copia	Elimina				1993-04-21
<input type="checkbox"/>	Modifica	Copia	Elimina				1994-05-11









ALL e ANY

Mentre matricola, nominativo e stipendio dei dipendenti che guadagnano di più è dato dalla query:

```
SELECT Matricola, Nome, Cognome
FROM Personale
WHERE Stipendio >= ALL (SELECT Stipendio
                        FROM Personale)
```

NB: Dal momento che NULL non è un numero, non può essere messo a confronto con gli altri stipendi (tranne che con la clausola ORDER BY, perchè viene cmq considerato più piccolo dello stipendio più piccolo) per cui, prima di provare questa query sul nostro DB, è necessario sostituire gli stipendi NULL con "0.00".

+ Opzioni

 			▼	Matricola	Nome	Cognome
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	QR001	Alfredo	Maino
<input type="checkbox"/>	 Modifica	 Copia	 Elimina	ZZ002	Mario	Ponsi

+ Opzioni

← T →						Stipendio	
<input type="checkbox"/>		Modifica		Copia		Elimina	25000.00
<input type="checkbox"/>		Modifica		Copia		Elimina	32400.00
<input type="checkbox"/>		Modifica		Copia		Elimina	28000.00
<input type="checkbox"/>		Modifica		Copia		Elimina	0.00
<input type="checkbox"/>		Modifica		Copia		Elimina	22000.00
<input type="checkbox"/>		Modifica		Copia		Elimina	0.00
<input type="checkbox"/>		Modifica		Copia		Elimina	24000.00
<input type="checkbox"/>		Modifica		Copia		Elimina	30000.00
<input type="checkbox"/>		Modifica		Copia		Elimina	32000.00
<input type="checkbox"/>		Modifica		Copia		Elimina	38500.00
<input type="checkbox"/>		Modifica		Copia		Elimina	35000.00
<input type="checkbox"/>		Modifica		Copia		Elimina	28800.00
<input type="checkbox"/>		Modifica		Copia		Elimina	25000.00
<input type="checkbox"/>		Modifica		Copia		Elimina	18500.00
<input type="checkbox"/>		Modifica		Copia		Elimina	39000.00
<input type="checkbox"/>		Modifica		Copia		Elimina	32400.00
<input type="checkbox"/>		Modifica		Copia		Elimina	18000.00
<input type="checkbox"/>		Modifica		Copia		Elimina	39000.00

le query nidificate - esercitazione

Riprendendo l'esempio precedente, proviamo le query seguenti usando ANY e ALL:

- Elenco dei dipendenti che lavorano in dipartimenti situati a Cosenza o a Crotone.
- Nome, cognome e città del dipartimento di appartenenza dei dipendenti che lavorano in una città il cui nome comincia con una lettera successiva alla 'C'.
- Nome e codice dei dipartimenti in cui non lavorano persone con cognome 'Rossi'.
- Elenco degli analisti che risiedono in città nelle quali risiede anche almeno un programmatore.
- Elenco degli analisti che lavorano in un dipartimento in cui lavora almeno un tecnico.
- Elenco di tutti i dipendenti non programmatori che guadagnano più dei programmatori.

le query nidificate - esercitazione

Riprendendo l'esempio precedente, proviamo le query seguenti usando ANY e ALL:

- Elenco con nome, cognome e data di nascita dei tecnici che sono più giovani di tutti gli altri dipendenti.
- Matricola, nome, cognome e qualifica dei tecnici che lavorano in una città in cui risiede almeno un programmatore.
- Elenco dei tecnici che risiedono in una città in cui lavora almeno un analista.
- Matricola, nome, cognome del dipendente il cui cognome è il primo/ultimo in ordine alfabetico.