

Esercitazione 2

SQL

c.vallati@iet.unipi.it

DB Riferimento

Consideriamo i seguenti schemi di una base di dati relazionale:

- **DIPARTIMENTI**(CodiceDipartimento, Nome, Indirizzo, Citta, NumeroDipendenti, CostiTotaliAnnuai, FinanziamentoTotaleAnnuo)
- **DOCENTI**(Matricola, Nome, Cognome, Ruolo, DataAssunzione, CodiceDipartimento, OreLezioneAnnue, PubblicazioniAnnue)

Viene data la seguente istanza della base di dati:

CodiceDipartimento	NomeDipartimento	Indirizzo	Citta	NumeroDipendenti	CostiTotaliAnnuai	FinanziamentoTotaleAnnuo
0001	Ingegneria Dell'informazione	Via Caruso 2	Pisa	10	350000	450000
0002	Ingegneria Civile	Via Diotisalvi 2	Pisa	5	250000	250000
0003	Matematica	Largo Pontecorvo 5	Pisa	6	300000	280000
0004	Letteratura e filologia	Piazza Evangelista Torricelli, 2	Pisa	4	220000	230000
0005	Infermieristica	Viale Rinaldo Piaggio, 2	Pontedera	6	300000	290000
0006	Medicina	Via Savi,10	Pisa	6	340000	350000

Matricola	Nome	Cognome	Ruolo	DataAssunzione	CodiceDipartimento	OreLezioneAnnuie	PubblicazioniAnnuie	StipendioAnnuo
000485	Chandler	Bing	Ricercatore	1/1/1999	4	80	NULL	50000
001245	Vittorio	Verdi	Professore Associato	10/4/2009	2	90	4	53000
002473	Francesco	Rossi	Professore Ordinario	10/12/2004	2	110	5	55000
004178	Mario	Rossi	Professore Associato	5/16/2010	1	115	15	50000
004580	Francesco	Verdi	Professore Ordinario	1/30/2005	1	150	20	60000
004595	Dexter	Morgan	Professore Ordinario	6/10/2004	1	110	14	51000
005698	Carlo	Vallati	Ricercatore	1/4/2016	1	60	10	40000
007896	Tyrion	Lannister	Ricercatore	6/30/2015	1	70	17	45000
010102	Joey	Tribbiani	Professore Ordinario	1/2/1999	5	150	10	65000
010244	Jack	Bauer	Ricercatore	6/9/2001	6	75	NULL	55000
010277	Jon	Snow	Professore Ordinario	2/5/1995	6	140	15	69000
014500	Francis	Underwood	Ricercatore	6/25/2014	3	100	0	45000
020150	Robin	Scherbatsky	Ricercatore	12/10/2013	3	110	1	47000
030012	Howard	Wolowitz	Professore Associato	12/30/2001	4	110	3	60000
030405	Ted	Mosby	Professore Associato	5/6/2002	5	120	9	55000
040512	Derek	Shepherd	Ricercatore	5/6/2003	5	40	5	39000
090104	Isobel	Stevens	Professore Associato	2/7/1998	6	125	11	45000

Tipi di dato

Columns in table

Column	Type	Nullable	Index
◇ CodiceDipartimento	varchar(4)	NO	PR
◇ Nome	varchar(45)	YES	
◇ Indirizzo	varchar(45)	YES	
◇ Citta	varchar(45)	YES	
◇ NumeroDipendenti	int(11)	YES	
◇ CostiTotaliAnnui	int(11)	YES	
◇ FinanziamentoTotaleAnnuo	int(11)	YES	

Column	Type	Default Value	Nullable
◇ Matricola	varchar(6)		NO
◇ Nome	varchar(45)		YES
◇ Cognome	varchar(45)		YES
◇ Ruolo	varchar(45)		YES
◇ DataAssunzione	date		YES
◇ CodiceDipartimento	varchar(4)		YES
◇ OreLezioneAnnue	int(11)		YES
◇ PubblicazioniAnnue	int(11)		YES
◇ StipendioAnnuo	int(11)		YES

ES 1: WHERE condizioni semplici

1. Indicare il Nome e Cognome dei docenti che hanno uno stipendio annuo maggiore di 50000EU
2. Indicare il Nome e Cognome dei Ricercatori che hanno uno stipendio maggiore di 50000 EU
3. Indicare Il Nome, Cognome e Ruolo dei docenti che hanno uno stipendio compreso tra 30000 EU e 50000 EU
4. Indicare il Nome, Cognome e Mansione dei docenti che non hanno fornito dati riguardo al numero annuale di pubblicazioni

SOL 1: WHERE condizioni semplici

1. SELECT Nome, Cognome FROM Docenti WHERE StipendioAnnuo > 50000;
2. SELECT Nome, Cognome FROM Docenti WHERE StipendioAnnuo > 30000 AND Ruolo = 'Ricercatore';
3. SELECT Nome, Cognome, Ruolo FROM Docenti WHERE StipendioAnnuo BETWEEN 30000 AND 50000;
4. SELECT Nome, Cognome, Ruolo FROM Docenti WHERE PubblicazioniAnnue IS NULL;

ES 2: DISTINCT, ORDER BY, LIKE

1. Indicare le città in cui hanno sede i dipartimenti
2. Indicare l'elenco dei possibili ruoli, rinominare il campo con la dicitura RuoliPossibili
3. Indicare l'elenco dei dipartimenti elencato per costo totale annuo crescente
4. Indicare l'elenco dei dipartimenti elencato per finanziamento annuo decrescente
5. Indicare i dipartimenti che abbiano nell'indirizzo una via o un viale

SOL 2: DISTINCT, ORDER BY, LIKE

1. `SELECT DISTINCT Citta FROM Dipartimenti;`
2. `SELECT DISTINCT Ruolo AS RuoliPossibili FROM Docenti;`
3. `SELECT * FROM dipartimenti ORDER BY CostiTotaliAnnui;`
4. `SELECT * FROM dipartimenti ORDER BY FinanziamentoTotaleAnnuo DESC;`
5. `SELECT * FROM dipartimenti WHERE Indirizzo LIKE '%Via%';`

ES 3:Funzioni di aggregazione

1. Contare il numero totale dei docenti dell'universita
2. Ricavare lo stipendio massimo tra i docenti rinominando il campo trovato in StipendioMassimo (usare il costrutto LIMIT)
3. Ricavare il numero di docenti per dipartimento, mostrare numero docenti e codice dipartimento (usare GROUP BY)
4. Ricavare l'elenco dei codici dei dipartimenti i cui docenti pubblicano un numero totale di pubblicazioni maggiore di dieci

SOL 3:Funzioni di aggregazione

1. `SELECT count(*) FROM docenti;`
2. `SELECT StipendioAnnuo AS StipendioMassimo
FROM docenti ORDER BY StipendioAnnuo DESC
LIMIT 1;`
3. `SELECT CodiceDipartimento,COUNT(*) AS
NumeroDocenti FROM docenti GROUP BY
CodiceDipartimento;`
4. `SELECT CodiceDipartimento,
SUM(PubblicazioniAnnue) AS TotalePubblicazioni
FROM docenti GROUP BY CodiceDipartimento
HAVING SUM(PubblicazioniAnnue) > 10;`

ES 4: Join

1. Estrarre i nomi e cognomi dei docenti e relativo indirizzo lavorativo
2. Estrarre i cognomi docenti affiliati con il dipartimento di ingegneria dell'informazione
3. Estrarre il numero totale di ore di didattica svolte dai docenti di ogni dipartimento
4. Trovare il nome del dipartimento con meno pubblicazioni annue

SOL 4: Join

1. `SELECT doc.Nome, doc.Cognome, dip.Indirizzo FROM docenti AS doc INNER JOIN dipartimenti AS dip ON dip.CodiceDipartimento = doc.CodiceDipartimento;`
2. `SELECT doc.Nome, doc.Cognome FROM docenti AS doc INNER JOIN dipartimenti AS dip ON dip.CodiceDipartimento = doc.CodiceDipartimento WHERE dip.NomeDipartimento LIKE '%informazione';`
OPPURE
`SELECT doc.Nome, doc.Cognome FROM docenti AS doc INNER JOIN dipartimenti AS dip ON dip.CodiceDipartimento = doc.CodiceDipartimento WHERE dip.NomeDipartimento = 'Ingegneria Dell"informazione';`
3. `SELECT dip.Nome, SUM(doc.OreLezioneAnnuie) FROM docenti AS doc INNER JOIN dipartimenti AS dip ON dip.CodiceDipartimento = doc.CodiceDipartimento GROUP BY dip.Nome;`
4. `SELECT dip.Nome, SUM(doc.PubblicazioniAnnuie) FROM docenti AS doc INNER JOIN dipartimenti AS dip ON dip.CodiceDipartimento = doc.CodiceDipartimento GROUP BY dip.NomeDipartimento ORDER BY SUM(doc.PubblicazioniAnnuie) LIMIT 1;`

NOTA: In questi casi dato che il JOIN viene fatto sugli attributi a comune INNER JOIN ... ON ... può essere sostituito con il NATURAL JOIN in cui la condizione di uguaglianza è implicita

Es 5: IN, ANY, ALL

1. Estrarre i nomi e cognomi dei docenti il cui dipartimento NON ha sede a Pisa
2. Estrarre i dipartimenti che hanno almeno un dipendente che prende più di 50000euro annui
3. Estrarre il nome del dipartimento che ha il dipendente che guadagna di più
4. Estrarre i nomi dei dipartimenti i cui dipendenti producono tutti almeno 10 pubblicazioni annue

SOL 5: IN, ANY, ALL

1. SELECT Nome, Cognome FROM docenti WHERE CodiceDipartimento NOT IN (SELECT CodiceDipartimento FROM dipartimenti WHERE Citta = 'Pisa') ;
2. SELECT DISTINCT dip.NomeDipartimento FROM dipartimenti AS dip JOIN docenti AS doc ON dip.CodiceDipartimento = doc.CodiceDipartimento WHERE doc.Matricola = ANY (SELECT Matricola FROM docenti WHERE StipendioAnnuo > 50000) ;
3. SELECT dip.NomeDipartimento FROM dipartimenti AS dip JOIN docenti AS doc ON dip.CodiceDipartimento = doc.CodiceDipartimento WHERE doc.StipendioAnnuo >= ALL (SELECT StipendioAnnuo FROM docenti) ;

SOL 5: IN, ANY, ALL

4.

```
SELECT DISTINCT dip.NomeDipartimento FROM dipartimenti AS dip
JOIN docenti AS doc ON dip.CodiceDipartimento =
doc.CodiceDipartimento WHERE dip.CodiceDipartimento NOT IN
(SELECT CodiceDipartimento FROM docenti WHERE
PubblicazioniAnnue < 10) ;
```

NOTA: in questo caso come altri casi simili conviene ragionare al contrario, cioè individuare prima i dipartimenti che non soddisfano i criteri dati (cioè che hanno almeno un docente con meno di 10 pubblicazioni annue) e poi stampare solo quelli che non fanno parte di questo insieme. La soluzione alternativa (cioè che tenta direttamente di trovare il numero di dipartimenti che soddisfa la soluzione data) è molto complessa come si può vedere:

```
SELECT DISTINCT dip.NomeDipartimento FROM dipartimenti AS dip
JOIN docenti AS doc ON dip.CodiceDipartimento =
doc.CodiceDipartimento WHERE
(SELECT COUNT(*) FROM docenti WHERE CodiceDipartimento =
dip.CodiceDipartimento) = (SELECT COUNT(*) FROM docenti WHERE
(PubblicazioniAnnue >= 10 OR PubblicazioniAnnue IS NULL) AND
CodiceDipartimento = dip.CodiceDipartimento) ;
```