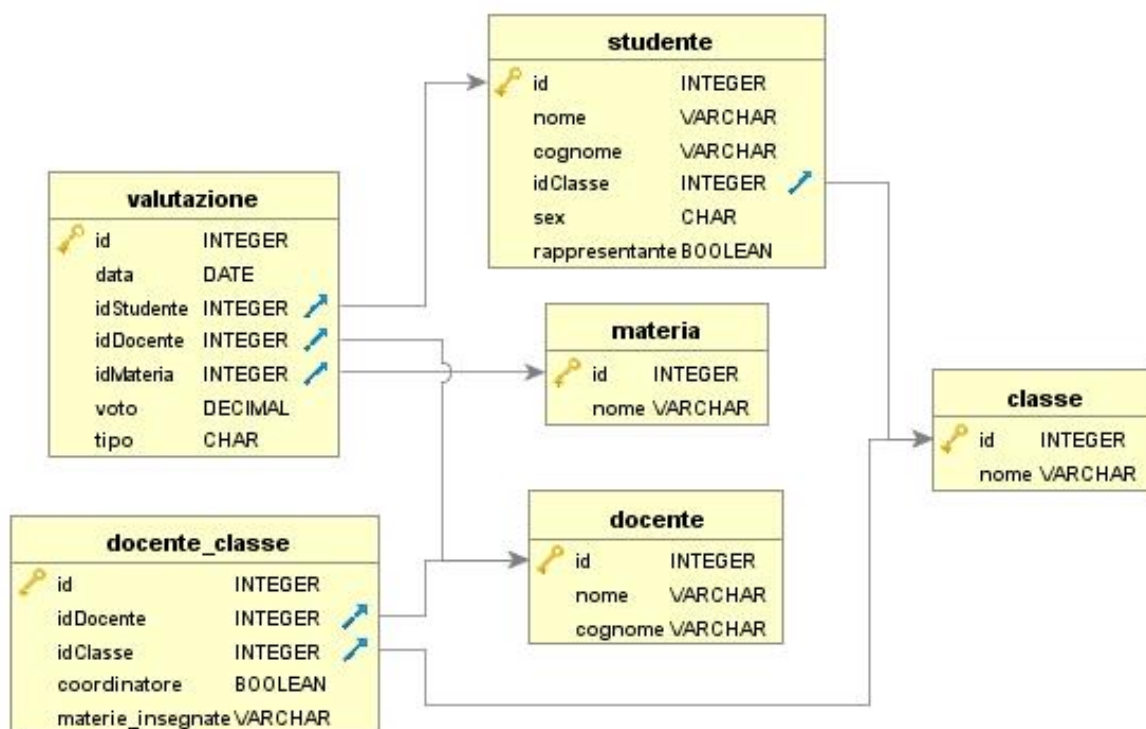


GUIDA SQL

Con questo esercizio vi spiegheremo come creare le query di un database partendo dallo schema tabellare.

I comandi base per creare una query sono: ***select***, ***from*** e ***where***.

- Nel comando ***select*** vanno trascritti tutti i campi che vogliamo che siano visti nella query separandoli con una virgola.
- In ***from*** vengono elencate le tabelle in cui si trovano le informazioni riguardanti la query.
- Nel comando ***where*** vengono scritte le condizioni per creare la query.



1. Voglio gli studenti rappresentanti di classe.

La query riguarda solamente la tabella “*studente*” perciò, nel comando “*from*”, viene scritta solamente quella. In “*select*” vengono richiesti gli attributi “*nome*” e “*cognome*” della tabella *studente*. In “*where*” viene scritta la condizione secondo la quale nella query, appaiono soltanto i rappresentanti di classe.

```
SELECT    s.nome, s.cognome
FROM      studente s
WHERE     s.rappresentante='true'
```

2. Voglio le studentesse della classe 3^aRB.

Nei comandi “*select*” e “*from*” il procedimento è il medesimo del punto precedente; mentre in “*where*” viene introdotta la foreign key ovvero l'attributo che collega una tabella ad un'altra. Per fare ciò, basta eguagliare l'id della classe e l'attributo “*idClasse*” della tabella “*studente*”.

```
SELECT    s.nome, s.cognome, c.nome
FROM      studente s, classe c
WHERE     s.sex='F' and s.idClasse=c.id and c.id='2'
```

3. Voglio i docenti della 3^aRB.

In questo caso non è presente una foreign key che collega due tabelle quindi si ricorre a un'ulteriore tabella per collegarle. La tabella risultante è “*docente_classe*”: si eguagliano l'id della tabella “*classe*” con l'attributo “*idClasse*” della tabella risultante e l'id della tabella “*docente*” con l'attributo “*idDocente*” della tabella risultante.

```
SELECT    d.nome, d.cognome, c.nome
FROM      docente d, docente_classe dc, classe c
WHERE     dc.idClasse=c.id and dc.idDocente=d.id and c.id='2'
```

4. Voglio gli studenti in classe con “Elena Grandi”.

Qui la tabella “*docente*” occorre scriverla due volte: una per indicare gli studenti in generale e una per Elena Grandi. In “*from*” viene indicata in due modi diversi la stessa tabella. In “*where*” viene imposto che l'*idClasse* sia uguale per entrambe le tabelle e che l'id della tabella riguardante gli studenti in generale sia diverso dall'id della tabella di Elena Grandi. Per indicare il diverso si usano i simboli “*!=*”

```
SELECT    s2.nome, s2.cognome
FROM      studente s1, studente s2
WHERE     s1.nome='Elena' and s1.cognome='Grandi' and s2.idClasse=s1.idClasse and
s1.id!=s2.id
```

5. Voglio i docenti che insegnano in una classe con “Vincenzo Galella”.

Adesso vi faremo vedere il result della query prima senza, poi con la clausola “*distinct*”. La clausola *distinct* permette di eliminare le ripetizioni dei dati uguali in una query.

```
SELECT    d2.nome, d2.cognome
FROM      docente d1, docente_classe dc1, docente d2, docente_classe dc2
WHERE     d1.cognome='Galella' and d1.nome='Vincenzo' and dc1.idDocente=d1.id
and d2.id=dc2.idDocente and dc1.idClasse=dc2.idClasse and d1.id!=d2.id
```

	nome	cognome
1	Lucia	Guidotti
2	Maurizio	Giannotti
3	Alessandra	Mazzei
4	Lucia	Guidotti
5	Alessandra	Mazzei
6	Annamaria	Bocci

```
SELECT    distinct d2.nome, d2.cognome
FROM      docente d1, docente_classe dc1, docente d2, docente_classe dc2
WHERE     d1.cognome='Galella' and d1.nome='Vincenzo' and dc1.idDocente=d1.id and
d2.id=dc2.idDocente and dc1.idClasse=dc2.idClasse and d1.id!=d2.id
```

	nome	cognome
1	Lucia	Guidotti
2	Maurizio	Giannotti
3	Alessandra	Mazzei
4	Annamaria	Bocci

6. Voglio la media voti e il numero di voti di “Elena Grandi”.

In questo esercizio vengono usate alcune delle cinque “funzioni aggregate”:

<i>min</i>	<i>max</i>	<i>sum</i>	<i>avg</i>	<i>count</i>
minimo	massimo	somma	media aritmetica	contatore

Inoltre viene usata la funzione “as” che serve per rinominare in uscita il nome di una colonna.

```
SELECT s.nome, s.cognome, avg(v.voto) as media, count(v.voto) as numero voti
FROM studente s, valutazione v
WHERE v.idStudente=s.id and s.nome='Elena' and s.cognome='Grandi'
```

7. Voglio i tre alunni con il voto più alto.

Infine si utilizzano i comandi “order by (desc)” e “limit”. Il comando “order by” serve per ordinare il result; l'ordinamento può essere decrescente (desc) o ascendente (asc). Il comando “limit” non è possibile usarlo senza la clausola “order by” e serve per limitare il numero massimo di righe.

```
SELECT s.nome, s.cognome, v.voto
FROM studente s, valutazione v
WHERE v.idStudente=s.id
ORDER BY v.voto desc
LIMIT 3
```

Appendice

Tabella classe:

	id	nome
1	1	3 RA
2	2	3 RB
3	3	3 AA
4	4	3 AB
5	5	3 SA
6	6	4 RA
7	7	4 RB
8	8	4 AA
9	9	4 AB
10	10	4 SA
11	11	5 RA
12	12	5 RB
13	13	5AA
14	14	5 AB
15	15	5 SA
16	16	1 A
17	17	1 B

Tabella docente:

	id	nome	coognome
1	1	Vincenzo	Gallella
2	2	Maurizio	Giannotti
3	3	Alessandra	Mazzei
4	4	Lucia	Guidotti
5	5	Annamaria	Bocci

Tabella materia:

	id	nome
1	1	Italiano
2	2	Matematica
3	3	Storia
4	4	Informatica
5	5	Inglese
6	6	Francese
7	7	Tedesco
8	8	Spagnolo
9	9	Ed Fisica
10	10	Economia Az
11	11	Economia Politica
12	12	Diritto
13	13	Geografia
14	14	Fisica
15	15	Chimica

Tabella docente_classe:

	id	idDocente	idClasse	coordinatore	materie_insegnate
1	1	1	1	false	Informatica
2	2	1	2	true	Informatica
3	3	1	6	false	Informatica
4	4	1	7	false	Informatica
5	5	1	8	false	Informatica
6	6	2	2	false	Educazione fisica
7	7	3	2	false	Italiano e Storia
8	8	3	7	true	Italiano e Storia
9	9	3	12	false	Italiano e Storia
10	10	4	1	false	Francese
11	11	4	6	false	Francese
12	12	4	11	true	Francese
13	13	5	9	false	Economia Aziendale
14	14	5	13	false	Economia Aziendale
15	15	5	4	false	Economia Aziendale
16	16	1	4	false	Informatica

Tabella studente:

	id	nome	cognome	idClasse	sex	rappresentante
1	1	Giulia EBE	Pardi	2	F	false
2	2	Elena	Grandi	1	F	false
3	3	Lucia	Pera	1	F	false
4	4	Giovanni	Sguazzi	5	M	false
5	5	Alessia	Di Cesare	1	F	false
6	6	Bianca	Cantoneru	2	F	false
7	7	Sonia	Cantoneru	1	F	false
8	8	Bianca	Alistar	6	F	true
9	9	Lorenzo	Cusato	6	M	false
10	10	Federica	Romani	7	F	false
11	11	Marco	Migliavacca	8	M	false
12	12	Beatrice	Lena	2	F	true
13	13	Alessandra	Unti	4	F	false
14	14	Angela	Buiuc	4	F	true

Tabella valutazione:

	id	data	idStudente	idDocente	idMateria	voto	tipo
1	1	10/3	3	3	3	6	p
2	2	10/3	5	1	4	5.5	p
3	3	10/3	2	2	9	5.5	p
4	4	10/3	5	1	4	6	p
5	5	10/3	2	4	6	4	p
6	6	10/3	2	1	4	3	p
7	7	10/3	3	1	4	4.75	p
8	8	10/3	3	3	1	6.5	p
9	9	10/3	2	4	6	6	p
10	10	18/2	14	1	4	3	p
11	19	11/2	2	5	3	5	p