

# Corso di Basi di Dati

## Esercitazioni in classe

### Programmazione in SQL base (Cap. 4)

#### Capitolo 4. Esercizio 2

Definire un attributo che permetta di rappresentare:

- stringhe di lunghezza massima pari a 256 caratteri,
- su cui non sono ammessi valori nulli
- e con valore di default "sconosciuto".

1

#### Capitolo 4. Esercizio 3

Dare le definizioni SQL delle tre tabelle

FONDISTA(Nome, Nazione, Età)  
GAREGGIA(NomeFondista, NomeGara, Piazzamento)  
GARA(Nome, Luogo, Nazione, Lunghezza)

rappresentando in particolare i vincoli di foreign key della tabella GAREGGIA.

2

#### Capitolo 4. Esercizio 4

Dare le definizioni SQL delle tabelle

AUTORE (Nome, Cognome, DataNascita, Nazionalità)  
LIBRO (TitoloLibro, NomeAutore, CognomeAutore, Lingua)

Per il vincolo foreign key specificare una politica di **cascade** sulla cancellazione e di set null sulle modifiche.

3

#### Capitolo 4. Esercizio 5

Dato lo schema dell'esercizio precedente, spiegare cosa può capitare con l'esecuzione dei seguenti comandi di aggiornamento:

```
delete from AUTORE where Cognome = 'Rossi'
```

```
update LIBRO set NomeAutore= 'Umberto'  
where CognomeAutore = 'Eco'
```

```
insert into AUTORE (Nome, Cognome)  
values ('Antonio', 'Bianchi')
```

```
update AUTORE set Nome = 'Italo'  
where Cognome = 'Calvino'
```

5

4

#### Capitolo 4. Esercizio 5

Dato lo schema dell'esercizio precedente, spiegare cosa può capitare con l'esecuzione dei seguenti comandi di aggiornamento:

```
delete from AUTORE where Cognome = 'Rossi'
```

- Il comando cancella dalla tabella AUTORE tutte le tuple con Cognome = 'Rossi'.
- A causa della politica cascade anche le tuple di LIBRO con CognomeAutore = 'Rossi' verranno eliminate.

6

#### Capitolo 4. Esercizio 5

Dato lo schema dell'esercizio precedente, spiegare cosa può capitare con l'esecuzione dei seguenti comandi di aggiornamento:

```
update LIBRO set NomeAutore= 'Umberto'
where CognomeAutore = 'Eco'
```

Il comando non è corretto: Nome e Cognome sono attributi della tabella AUTORE e non della tabella LIBRO

#### Capitolo 4. Esercizio 5

Dato lo schema dell'esercizio precedente, spiegare cosa può capitare con l'esecuzione dei seguenti comandi di aggiornamento:

```
insert into AUTORE (Nome, Cognome)
values ('Antonio', 'Bianchi')
```

- Il comando aggiunge una nuova tupla alla tabella AUTORE.
- Non ha alcun effetto sulla tabella LIBRO

7

#### Capitolo 4. Esercizio 5

Dato lo schema dell'esercizio precedente, spiegare cosa può capitare con l'esecuzione dei seguenti comandi di aggiornamento:

```
update AUTORE set Nome = 'Italo'
where Cognome = 'Calvino'
```

- Le tuple di AUTORE con Cognome = Calvino vengono aggiornate a Nome = Italo.
- A causa della politica set null gli attributi NomeAutore e CognomeAutore delle tuple di Libro con CognomeAutore = Calvino vengono posti a NULL

8

#### Capitolo 4. Esercizio 6

Date le definizioni:

```
create domain Dominio as integer default 10
create table Tabella (Attributo Dominio default 5)
```

Indicare cosa avviene in seguito ai comandi:

1. alter table Tabella alter column Attributo drop default
2. alter domain Dominio drop default
3. drop domain Dominio

9

#### Capitolo 4. Esercizio 7

Con riferimento ad una relazione

PROFESSORI(CF, Nome, Eta, Qualifica),  
scrivere le interrogazioni SQL che calcolano l'età media dei professori di ciascuna qualifica, nei due casi seguenti:

1. se l'eta non è nota si usa per essa il valore nullo
2. se l'eta non è nota si usa per essa il valore 0.

10

#### Capitolo 4. Esercizio 8

Spiegare perchè in SQL è previsto (e necessario) un operatore di **unione** mentre in molte versioni non esistono gli operatori di **intersezione** e **differenza**.

11

12

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

AEROPORTO (Città, Nazione, NumPiste)  
VOLO (IdVolo, GiornoSett, CittàPart, OraPart, CittàArr, OraArr, TipoAereo)  
AEREO (TipoAereo, NumPasseggeri, QtaMerchi)



Scrivere le interrogazioni SQL che permettono di determinare:

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

AEROPORTO (Città, Nazione, NumPiste)  
VOLO (IdVolo, GiornoSett, CittàPart, OraPart, CittàArr, OraArr, TipoAereo)  
AEREO (TipoAereo, NumPasseggeri, QtaMerchi)



Scrivere le interrogazioni SQL che permettono di determinare:

- 1) Le città con un aeroporto di cui non è noto il numero di piste;

13

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

AEROPORTO (Città, Nazione, NumPiste)  
VOLO (IdVolo, GiornoSett, CittàPart, OraPart, CittàArr, OraArr, TipoAereo)  
AEREO (TipoAereo, NumPasseggeri, QtaMerchi)



Scrivere le interrogazioni SQL che permettono di determinare:

- 2) Le nazioni da cui parte e arriva il volo con codice AZ274;

14

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

AEROPORTO (Città, Nazione, NumPiste)  
VOLO (IdVolo, GiornoSett, CittàPart, OraPart, CittàArr, OraArr, TipoAereo)  
AEREO (TipoAereo, NumPasseggeri, QtaMerchi)



Scrivere le interrogazioni SQL che permettono di determinare:

- 3) I tipi di aereo usati nei voli che partono da Torino;

15

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

AEROPORTO (Città, Nazione, NumPiste)  
VOLO (IdVolo, GiornoSett, CittàPart, OraPart, CittàArr, OraArr, TipoAereo)  
AEREO (TipoAereo, NumPasseggeri, QtaMerchi)



Scrivere le interrogazioni SQL che permettono di determinare:

- 4) I tipi di aereo e il corrispondente numero di passeggeri per i tipi di aereo usati nei voli che partono da Torino. Se la descrizione dell'aereo non è disponibile, visualizzare solamente il tipo;

17

16

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

AEROPORTO (Città, Nazione, NumPiste)  
VOLO (IdVolo, GiornoSett, CittàPart, OraPart, CittàArr, OraArr, TipoAereo)  
AEREO (TipoAereo, NumPasseggeri, QtaMerchi)



Scrivere le interrogazioni SQL che permettono di determinare:

- 5) Le città da cui partono voli internazionali;

18

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

```
AEROPORTO (Città, Nazione, NumPiste)
VOLO (IdVolo, GiornoSett, CittàPart, OraPart,
      CittàArr, OraArr, TipoAereo)
AEREO (TipoAereo, NumPasseggeri, QtaMerci)
```



Scrivere le interrogazioni SQL che permettono di determinare:

- 6) Le città da cui partono voli diretti a Bologna, ordinate alfabeticamente;

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

```
AEROPORTO (Città, Nazione, NumPiste)
VOLO (IdVolo, GiornoSett, CittàPart, OraPart,
      CittàArr, OraArr, TipoAereo)
AEREO (TipoAereo, NumPasseggeri, QtaMerci)
```



Scrivere le interrogazioni SQL che permettono di determinare:

- 7) Il numero di voli internazionali che partono il giovedì da Napoli;

19

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

```
AEROPORTO (Città, Nazione, NumPiste)
VOLO (IdVolo, GiornoSett, CittàPart, OraPart,
      CittàArr, OraArr, TipoAereo)
AEREO (TipoAereo, NumPasseggeri, QtaMerci)
```



Scrivere le interrogazioni SQL che permettono di determinare:

- 9) Le città francesi da cui partono più di venti voli alla settimana diretti in Italia

20

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

```
AEROPORTO (Città, Nazione, NumPiste)
VOLO (IdVolo, GiornoSett, CittàPart, OraPart,
      CittàArr, OraArr, TipoAereo)
AEREO (TipoAereo, NumPasseggeri, QtaMerci)
```



Scrivere le interrogazioni SQL che permettono di determinare:

- 10) Gli aeroporti italiani che hanno **solo voli interni**.  
Rappresentare questa interrogazione in tre modi:
- con **operatori insiemistici**;
  - con un interrogazione **nidificata** con l'operatore **not in**;
  - con un interrogazione **nidificata** con l'operatore **not exists**;

21

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

```
AEROPORTO (Città, Nazione, NumPiste)
VOLO (IdVolo, GiornoSett, CittàPart, OraPart,
      CittàArr, OraArr, TipoAereo)
AEREO (TipoAereo, NumPasseggeri, QtaMerci)
```



- 12) Le città che sono servite dall'aereo caratterizzato dal **massimo** numero di passeggeri

22

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

```
AEROPORTO (Città, Nazione, NumPiste)
VOLO (IdVolo, GiornoSett, CittàPart, OraPart,
      CittàArr, OraArr, TipoAereo)
AEREO (TipoAereo, NumPasseggeri, QtaMerci)
```



#### COMPITO PER CASA:

Provare le seguenti query e discuterne il risultato

```
select count(CittàArr)
from AEROPORTO as A1 join VOLO
on A1.Città=CittàPart
   join AEROPORTO as A2
   on CittàArr=A2.Città
where A1.Nazione='Italia' and
      A2.Nazione <> 'Italia'
group by CittàPart
```

```
select count(*), CittàPart
from AEROPORTO as A1 join VOLO
on A1.Città=CittàPart
   join AEROPORTO as A2
   on CittàArr=A2.Città
where A1.Nazione='Italia' and
      A2.Nazione <> 'Italia'
group by CittàPart
```

23

24

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)  
CONTIENE(NroSerieDisco, CodiceReg, NroProg)  
ESECUZIONE(CodiceReg, TitoloCanz, Anno)  
AUTORE(Nome, TitoloCanzone)  
CANTANTE(NomeCantante, CodiceReg)



#### Capitolo 4. Esercizio 14

Dato il seguente schema:

DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)  
CONTIENE(NroSerieDisco, CodiceReg, NroProg)  
ESECUZIONE(CodiceReg, TitoloCanz, Anno)  
AUTORE(Nome, TitoloCanzone)  
CANTANTE(NomeCantante, CodiceReg)



formulare le interrogazioni SQL che permettono di determinare:

- 1) I cantautori (persone che hanno cantato e scritto la stessa canzone) il cui nome inizia per 'd';

25

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)  
CONTIENE(NroSerieDisco, CodiceReg, NroProg)  
ESECUZIONE(CodiceReg, TitoloCanz, Anno)  
AUTORE(Nome, TitoloCanzone)  
CANTANTE(NomeCantante, CodiceReg)



formulare le interrogazioni SQL che permettono di determinare:

- 2) I titoli dei dischi che contengono canzoni di cui **non** si conosce l'anno di registrazione;

26

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)  
CONTIENE(NroSerieDisco, CodiceReg, NroProg)  
ESECUZIONE(CodiceReg, TitoloCanz, Anno)  
AUTORE(Nome, TitoloCanzone)  
CANTANTE(NomeCantante, CodiceReg)



formulare le interrogazioni SQL che permettono di determinare:

- 3) I pezzi del disco con numero di serie 78574, **ordinati** per numero progressivo, con indicazione degli interpreti per i pezzi che hanno associato un cantante;

27

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)  
CONTIENE(NroSerieDisco, CodiceReg, NroProg)  
ESECUZIONE(CodiceReg, TitoloCanz, Anno)  
AUTORE(Nome, TitoloCanzone)  
CANTANTE(NomeCantante, CodiceReg)



formulare le interrogazioni SQL che permettono di determinare:

- 4) Gli autori e i cantanti **puri**, ovvero autori che non hanno mai registrato una canzone e cantanti che non hanno mai scritto una canzone;

28

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)  
CONTIENE(NroSerieDisco, CodiceReg, NroProg)  
ESECUZIONE(CodiceReg, TitoloCanz, Anno)  
AUTORE(Nome, TitoloCanzone)  
CANTANTE(NomeCantante, CodiceReg)



formulare le interrogazioni SQL che permettono di determinare:

- 5) I cantanti del disco che contiene il **maggior numero** di canzoni;

29

30

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)  
CONTIENE(NroSerieDisco, CodiceReg, NroProg)  
ESECUZIONE(CodiceReg, TitoloCanz, Anno)  
AUTORE(Nome, TitoloCanzone)  
CANTANTE(NomeCantante, CodiceReg)



formulare le interrogazioni SQL che permettono di determinare:

- 6) Gli autori solisti di "collezioni di successi" (dischi in cui tutte le canzoni sono di un solo cantante e in cui almeno tre registrazioni sono di anni precedenti la pubblicazione del disco);

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)  
CONTIENE(NroSerieDisco, CodiceReg, NroProg)  
ESECUZIONE(CodiceReg, TitoloCanz, Anno)  
AUTORE(Nome, TitoloCanzone)  
CANTANTE(NomeCantante, CodiceReg)



formulare le interrogazioni SQL che permettono di determinare:

- 7) I cantanti che non hanno **mai** registrato una canzone come solisti;

31

#### Capitolo 4. Esercizio 14

Dato il seguente schema:

DISCO(NroSerie, TitoloAlbum, Anno, Prezzo)  
CONTIENE(NroSerieDisco, CodiceReg, NroProg)  
ESECUZIONE(CodiceReg, TitoloCanz, Anno)  
AUTORE(Nome, TitoloCanzone)  
CANTANTE(NomeCantante, CodiceReg)



formulare le interrogazioni SQL che permettono di determinare:

- 8) I cantanti che non hanno **mai** inciso un disco in cui comparissero come **unici** cantanti;

33

32

#### Capitolo 4. Esercizio 9

Considerare le relazioni

IMPIEGATI (Matricola, Nome, Stipendio)

DIPARTIMENTI (Codice, Direttore)

esiste v.i.r. tra direttore di dipartimenti e matricola di impiegati

e le due interrogazioni seguenti, specificare se e in quali casi esse possono produrre **risultati diversi**:

```
select avg(Stipendio) from Impiegati  
where matricola in (select Direttore from Dipartimenti)
```

```
select avg(Stipendio) from Impiegati I, Dipartimenti D  
where I.matricola = D.Direttore
```

34