



UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO DI MATEMATICA

LAUREA TRIENNALE IN INFORMATICA

BASI DI DATI - LABORATORIO 3

SQL Avanzato

Massimiliano de Leoni

deleoni@math.unipd.it

Alessandro Padella

alessandro.padella@phd.unipd.it

Samuel Cognolato

samuel.cognolato@studenti.unipd.it

Indice

1	Creazione tabelle	4
1.1	Creazione tabelle	4
2	Inserimento dei dati	6
3	Query di base	8
4	Esercizi	10
5	Soluzioni	11
5.1	Popolamento database	11
5.2	Interrogazioni	11

1 Creazione tabelle

Questo laboratorio è basato su una versione semplificata di un database contenente dati reali, relativi ad una compagnia logistica tedesca. I dati si riferiscono a viaggi per trasporti di beni nell'anno 2016. Definiamo quindi le seguenti tabelle con i relativi campi dati:

- HUBS:
 - hub di tipo VARCHAR(30) e PRIMARY KEY;
 - country di tipo CHAR(2).
- TRIPS:
 - number di tipo VARCHAR(12) NOT NULL e PRIMARY KEY;
 - meters di tipo REAL.
- LEGS:
 - trip_number di tipo VARCHAR(12) NOT NULL e PRIMARY KEY;
 - origin di tipo VARCHAR(30) NOT NULL e PRIMARY KEY;
 - destination di tipo VARCHAR(30) NOT NULL e PRIMARY KEY;
 - departure_time di tipo TIMESTAMP NOT NULL e PRIMARY KEY;
 - arrival_time di tipo TIMESTAMP NOT NULL e PRIMARY KEY.

Le tabelle devono avere le seguenti chiavi esterne:

- trip_number nella tabella LEGS che riferisce Number della tabella TRIPS;
- destination nella tabella LEGS che riferisce Hub nella tabella HUBS;
- origin nella tabella LEGS che riferisce Hub nella tabella HUBS.

1.1 Creazione tabelle

Il codice per la creazione delle tabelle è disponibile su Moodle ed è il seguente:

```
01 | --Eliminazione di entrambe le tabelle in caso esistano
02 | DROP TABLE IF EXISTS Legs;
03 | DROP TABLE IF EXISTS Hubs;
04 | DROP TABLE IF EXISTS Trips;
05 |
06 |
07 | --Creazione tabella Hubs
08 | CREATE TABLE Hubs (
09 |   Hub VARCHAR(30),
10 |   Country CHAR(2),
11 |   PRIMARY KEY(Hub)
12 | );
13 |
14 | --Creazione tabella Trips
15 | CREATE TABLE Trips (
16 |   Number VARCHAR(12) NOT NULL,
17 |   Meters REAL,
18 |   PRIMARY KEY(Number)
19 | );
```

```
20 |  
21 | --Creazione tabella Legs  
22 | CREATE TABLE Legs (  
23 | Trip_number VARCHAR(12) NOT NULL,  
24 | Origin VARCHAR(30) NOT NULL,  
25 | Destination VARCHAR(30) NOT NULL,  
26 | Departure_time TIMESTAMP NOT NULL,  
27 | Arrival_time TIMESTAMP NOT NULL,  
28 | FOREIGN KEY (Trip_number) REFERENCES Trips (Number) ON DELETE CASCADE,  
29 | FOREIGN KEY (Destination) REFERENCES Hubs (Hub) ON DELETE CASCADE,  
30 | FOREIGN KEY (Origin) REFERENCES Hubs (Hub) ON DELETE CASCADE,  
31 | PRIMARY KEY (Trip_number, Origin, Destination, Departure_time, Arrival_time  
    | )  
32 | );
```

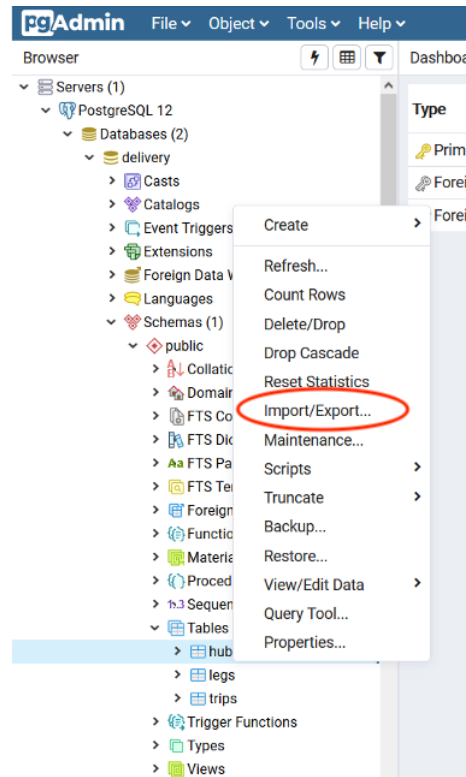


Figura 1: Importazione ed esportazione del contenuto di una tabella.

2 Inserimento dei dati

Il contenuto delle tabelle è memorizzato all'interno di alcuni file CSV: `hubs.csv`, `trips.csv` e `legs.csv`. Per illustrare l'importazione, sarà utilizzata la tabella `HUBS`, ma la stessa procedura deve essere ripetuta per le altre tabelle.

Per iniziare, fare click destro sulla tabella `HUBS` (sul pannello di sinistra) e selezionare `Import/Export...` come mostrato in Figura 1.

Questa azione aprirà la seguente finestra di popup come in Figura 3 all'interno della quale è necessario (1) cambiare da `Export` ad `Import`, (2) selezionare il percorso del file `hubs.csv`, (3) specificare che il file contiene l'header e (4) selezionare la virgola come **delimiter**.

Infine (5) premere il tasto `OK` per iniziare l'importazione. Quando finisce, se questa è andata a buon fine, si vedrà la seguente finestra popup in basso a destra sulla finestra.

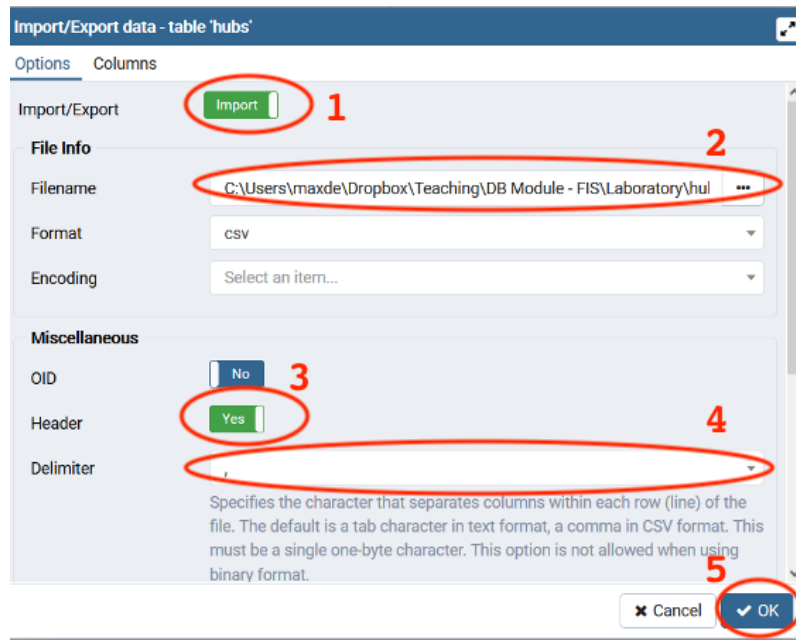


Figura 2: Importazione del contenuto di file CSV.

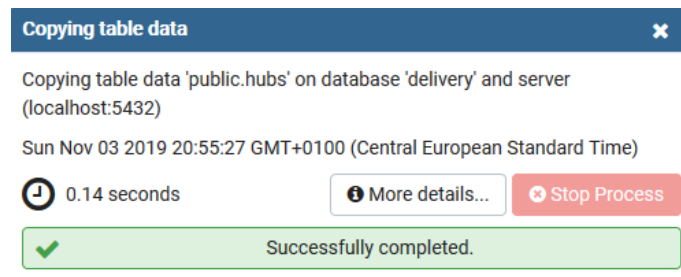


Figura 3: Completamento operazione.

Per controllare il contenuto di una tabella, ad esempio la tabella HUBS appena importata, si può fare click destro su di essa, selezionare **View/Edit Data** ed infine selezionare **All rows**, **First 100 rows** oppure **Last 100 rows**.

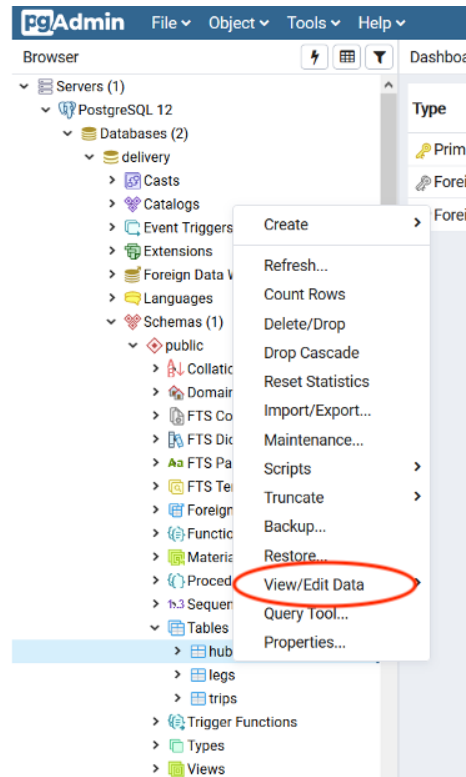


Figura 4: Visualizzazione contenuto della tabella.

La richiesta sarà convertita nella query SQL corrispondente e visualizzata. Occorre ripetere l'importazione anche per le altre due tabelle, riflettendo sul fatto che bisogna eseguire le importazioni nell'ordine corretto. Dopodiché, prima di continuare con l'esercitazione, occorre spendere qualche minuto ad analizzare il contenuto delle tabelle. **NOTA:** l'import dei dati per la tabella LEGS, data la sua dimensione, può richiedere alcuni secondi.

3 Query di base

Con le seguenti query “di base” cercheremo di prendere confidenza con il database.

Ad esempio, potrebbe interessarci estrarre il numero di hubs presenti in ogni nazione.

```
01 | SELECT country, COUNT(*) AS number
02 | FROM HUBS
03 | GROUP BY country
```

Potrebbe essere interessante ordinare i risultati in ordine decrescente rispetto al numero di hub per nazione.

```
01 | SELECT country, COUNT(*) AS number
02 | FROM HUBS
03 | GROUP BY country
04 | ORDER BY number DESC
```


Un'altra informazione utile potrebbe essere sapere qual è la durata media di ogni tratta.

```
01 | SELECT AVG(arrival_time - departure_time)
02 | FROM LEGS
```

Per fare delle analisi più complesse potrebbe interessarci sapere quali sono le tratte che partono dall'Italia.

```
01 | SELECT *
02 | FROM HUBS,LEGS
03 | WHERE hub = origin AND country='IT'
```

Possiamo anche estrarre solo il nome degli hub italiani da cui partono le tratte.

```
01 | SELECT hub
02 | FROM HUBS,LEGS
03 | WHERE hub = origin AND country='IT'
```

Rispetto a questa analisi, una delle domande che potremmo porci è quindi: *Qual è l'hub di partenza più comune tra quelli italiani?*

La risposta a questa domanda può essere ottenuta estraendo tutti gli hub di origine presenti in Italia, contando le loro occorrenze della tabella LEGS, che contiene tutte le tratte, e ordinando in ordine decrescente rispetto al numero di occorrenze ottenuto.

```
01 | SELECT hub,COUNT(*) AS number
02 | FROM HUBS, LEGS
03 | WHERE hub = origin AND country = 'IT'
04 | GROUP BY hub
05 | ORDER BY number DESC
```

4 Esercizi

Definire le query appropriate per rispondere alle seguenti domande:

1. Riflettendo sulla distanza percorsa: quant'è la media di distanza percorsa in metri? Qual è la distanza percorsa massima?
2. Quanti hub sono presenti in Italia?
3. Ordinare i paesi in base al numero di hub.
4. Quali sono i paesi più comuni (ordinare) per gli hub di origine?
5. Quali sono le tratte più frequentate?
6. Riflettendo sulla distanza percorsa: qual è la distanza percorsa media per paese di origine?
7. Prendendo in considerazione gli hub di partenza più comuni (i primi 10), quando sono stati utilizzati la prima e l'ultima volta in uscita?
8. Prendendo in considerazione gli hub di partenza più comuni (almeno 3000 partenze), quando sono stati utilizzati la prima e l'ultima volta?
9. Quali sono gli hub di partenza e destinazione più comuni in tutto il mondo (almeno 5000 voli tra partenze e arrivi)?
10. Quali sono gli hub italiani che sono partenze un numero di volte superiore alla media italiana?
11. Assumi di avere la seguente vista:

```
01 | CREATE VIEW minMaxTimePerTrip (trip_number, min_dep, max_arr) AS
02 | SELECT trip_number, min(departure_time), max(arrival_time)
03 | FROM legs
04 | GROUP BY trip_number;
```

Quali sono gli hub più comuni come prime partenze e ultimi arrivi per i trip? Nota: uno stesso `trip` (`trip_number`) può essere associato a più `legs`! Per questo esercizio risulterà comodo usare una vista.

12. Quali sono i paesi con le più comuni prime partenze e gli ultimi arrivi? Consiglio: una delle viste costruite alla query precedente può essere utile!
13. Qual è la distanza percorsa media per ogni paese di partenza?
14. Qual è la distanza percorsa media per ogni coppia di paesi (partenza, destinazione)?

5 Soluzioni

5.1 Popolamento database

Seguire, per tutte le tabelle, la procedura descritta nella sezione relativa all'inserimento dei dati.

5.2 Interrogazioni

1. Riflettendo sulla distanza percorsa: quant'è la media di distanza percorsa in metri? Qual è la distanza percorsa massima?

```
01 | SELECT AVG(meters), MAX(meters) FROM trips;
```

2. Quanti hub sono presenti in Italia?

```
01 | SELECT COUNT(*)  
02 | FROM hubs  
03 | WHERE country='IT';
```

3. Ordinare i paesi in base al numero di hub?

```
01 | SELECT country, COUNT(*) as number  
02 | FROM hubs  
03 | GROUP BY country  
04 | ORDER BY number DESC;
```

4. Quali sono i paesi più comuni (ordinare) per gli hub di origine?

```
01 | SELECT country, COUNT(*) as number  
02 | FROM legs JOIN hubs on origin=hub  
03 | GROUP BY country  
04 | ORDER BY number DESC;
```

5. Quali sono le tratte più frequentate?

```
01 | SELECT origin, destination, COUNT(*) as number  
02 | FROM legs  
03 | GROUP BY origin, destination  
04 | ORDER BY number DESC;
```

6. Riflettendo sulla distanza percorsa: qual è la distanza percorsa media per paese di origine?

```
01 | SELECT country, AVG(meters) as average  
02 | FROM trips, legs, hubs  
03 | WHERE number=trip_number AND origin=hub  
04 | GROUP BY country  
05 | ORDER BY average DESC;
```

7. Prendendo in considerazione gli hub di partenza più comuni (i primi 10), quando sono stati utilizzati la prima e l'ultima volta in uscita?

```

01 | SELECT origin, MIN(departure_time), MAX(departure_time)
02 | FROM legs JOIN (SELECT origin as common, COUNT(*) as number
03 | FROM legs
04 | GROUP BY origin
05 | ORDER BY number DESC
06 | LIMIT 10) as aux on origin=common
07 | GROUP BY origin;

```

8. Prendendo in considerazione gli hub di partenza più comuni (almeno 3000 partenze), quando sono stati utilizzati la prima e l'ultima volta in uscita?

```

01 | SELECT origin, MIN(departure_time), MAX(departure_time)
02 | FROM legs JOIN (SELECT origin as common, COUNT(*) as number
03 | FROM legs
04 | GROUP BY origin
05 | HAVING count(*) >= 3000) as aux on origin=common
06 | GROUP BY origin;

```

9. Quali sono gli hub di partenza e destinazione più comuni in tutto il mondo (almeno 5000 voli tra partenze e arrivi)?

```

01 | SELECT distinct hub
02 | FROM ((SELECT origin as hub, COUNT(*) as number
03 | FROM legs
04 | GROUP BY origin)
05 | union (SELECT destination as hub, COUNT(*) as number
06 | FROM legs
07 | GROUP BY destination)) as aux
08 | GROUP BY hub
09 | HAVING SUM(number) > 5000;

```

10. Quali sono gli hub italiani che sono partenze un numero di volte superiore alla media italiana?

```

01 | SELECT hub, COUNT(*)
02 | FROM hubs, legs
03 | WHERE hub=origin AND country='IT'
04 | GROUP BY hub
05 | HAVING COUNT(*) >= (SELECT AVG(number)
06 | FROM (SELECT hub, COUNT(*) as number
07 | FROM hubs, legs
08 | WHERE hub=origin AND country='IT'
09 | GROUP BY hub) as aux);

```

11. Quali sono gli hub più comuni come prime partenze e ultimi arrivi per i trip? Nota: uno stesso trip (trip_number) può essere associato a più legs! Per questo esercizio risulterà comodo usare una vista.

```

01 | CREATE VIEW tripFirstDepLastArr (trip_number, first_origin, last_dest) AS
02 | SELECT L1.trip_number, L1.origin, L2.destination
03 | FROM legs L1, legs L2, minTimePerTrip MM

```

```
04 | WHERE L1.trip_number=L2.trip_number AND
05 |       L1.trip_number=MM.trip_number AND
06 |       L1.departure_time=MM.min_dep AND L2.arrival_time=MM.max_arr;
```

```
01 | SELECT first_origin AS origin, last_dest AS destination, COUNT(*) AS
    | occurrence
02 | FROM tripFirstDepLastArr
03 | GROUP BY first_origin, last_dest
04 | ORDER BY occurrence DESC;
```

12. Quali sono i paesi con le più comuni prime partenze e gli ultimi arrivi? Consiglio: una delle viste costruite alla query precedente può essere utile!

```
01 | SELECT H1.country AS departure_country, H2.country AS arrival_country,
    | COUNT(*) AS occurrence
02 | FROM tripFirstDepLastArr T, hubs H1, hubs H2
03 | WHERE H1.hub=T.first_origin AND H2.hub=T.last_dest
04 | GROUP BY H1.country, H2.country
05 | ORDER BY occurrence DESC;
```

13. Qual è la distanza percorsa media per ogni paese di partenza?

```
01 | SELECT country, AVG(meters) AS average
02 | FROM legs, hubs, trips
03 | WHERE hubs.hub=legs.origin AND trips.number=trip_number
04 | GROUP BY country
05 | ORDER BY average DESC;
```

14. Qual è la distanza percorsa media per ogni coppia di paesi (partenza, destinazione)?

```
01 | SELECT h1.country AS country_origin, h2.country AS country_arrival, AVG
    | (meters) AS average
02 | FROM legs, hubs h1, hubs h2, trips
03 | WHERE h1.hub=legs.origin AND h2.hub=legs.destination AND trips.number=
    | trip_number
04 | GROUP BY h1.country, h2.country
05 | ORDER BY average DESC;
```