

**Antonio Giunta**

# **PROGRAMMAZIONE**

**Fondamenti della programmazione**

**Programmazione procedurale in C**

**Programmazione procedurale / Esercizi**

**Programmazione a oggetti in Java**

**Algoritmi**

**Strutture di dati in Java**

**Basi di dati relazionali / Integrazioni**

Proprietà Letteraria Riservata

L'usufrutto di questo libro (sia in formato cartaceo che digitale) è concesso esclusivamente all'utente indicato nell'intestazione di ogni pagina.

Non è concessa la cessione del libro a terzi, neanche in forma gratuita o parziale.

## 63

# BREVE GUIDA A POSTGRESQL PER WINDOWS

Il DBMS PostgreSQL ha un interprete dei comandi detto psql.

Con questo font

indico i messaggi inviati da psql.

**Con questo font**

indico gli inserimenti che l'utente deve fare.

---

### 63.1

## Gestione della sessione

#### 63.1.1 Entrata in psql

Selezioniamo il comando

Tasto Windows / PostgreSQL 16 / SQL Shell (psql)

Premiamo Invio dopo ognuna di queste stampe

Server [localhost]:

Database [postgres]:

Port [5432]:

Username [postgres]:

Inserisci la password per l'utente postgres: **xyz**

ATTENZIONE: Il code page della console (850) differisce dal code page di Windows (1252). I caratteri a 8-bit potrebbero non funzionare correttamente. Vedi le pagine di riferimento psql "Note per utenti Windows" per i dettagli.

Digita "help" per avere un aiuto.

postgres=#

A questo punto, siamo entrati nella sessione e possiamo iniziare a lavorare.

#### 63.1.2 Comandi di interazione con psql

Ogni comando inizia con il carattere \ ed è seguito da uno o più caratteri minuscoli.

Non necessita di essere concluso dal carattere ;.

#### 63.1.3 Uscita da psql

Digitiamo \q

---

## 63.2 Gestione di un database

### 63.2.1 Elenco dei database

Possiamo listare ogni database creato tramite il comando: `\l`

NB: `psql` memorizza il nome di un db trasformando ogni carattere maiuscolo di questo nome nel carattere minuscolo corrispondente.

### 63.2.2 Creazione di un database

Creiamo il db di nome `nomedb`: `CREATE DATABASE nomedb;`

Dovrebbe apparire il messaggio: `CREATE DATABASE`

### 63.2.3 Connessione con un database

Ci connettiamo a un database (precedentemente creato) digitando: `\c nomedb`

### 63.2.4 Eliminazione di un database

Per eliminare un database digitiamo: `DROP DATABASE nomedatabase;`

Se l'eliminazione è avvenuta, apparirà il seguente messaggio:

`DROP DATABASE`

NB Ovviamente, non possiamo eliminare il db al quale siamo connessi. Se ci proviamo, apparirà il messaggio:

`ERRORE: non si può eliminare il database aperto attualmente`

---

## 63.3 Gestione delle tabelle

### 63.3.1 Creazione delle tabelle

Creiamo una tabella mediante il comando `CREATE TABLE`.

Quando scriviamo un comando, possiamo dividerlo in più linee. In questo caso, `psql` risponde con il suo prompt finché non digitiamo il punto e virgola (che serve a indicare a `psql` che abbiamo concluso il comando corrente).

Dopo la creazione dovrebbe comparire il seguente messaggio: `CREATE TABLE`.

Per visualizzare le tabelle appartenenti al db al quale siamo connessi, digitiamo: `\dt`

Per visualizzare gli attributi (insieme ad altre sue informazioni, come il nome e il tipo) di una tabella appartenente al db al quale siamo connessi, digitiamo: `\d NOMETABELLA`

### 63.3.2 4.2 Eliminazione delle tabelle

Per eliminare una tabella dal database, digitiamo: `DROP TABLE NomeTabella;`

Se l'eliminazione è avvenuta, apparirà il seguente messaggio: `DROP TABLE`.

## 63.4 Modifica delle tuple

Dopo aver creato una tabella, possiamo inserirvi, cancellare e aggiornare tuple, utilizzando, rispettivamente, i comandi **INSERT**, **UPDATE** e **DELETE**.

---

## 63.5 Esecuzione di comandi memorizzati su file

Anziché eseguire comandi SQL digitandoli su terminale, spesso ci conviene scriverli in un file di testo e poi inviare il nome di questo file psql; il comando per far ciò è:

```
\i 'PercorsoFile\\NomeFile'
```

Usciamo da psql tramite il comando `\q`, e rientriamoci.

Supponiamo di avere un file di nome `prova.sql` nella cartella `C:\MiaCartella` contenente le seguenti istruzioni:

```
DROP DATABASE IF EXISTS provadb;  
CREATE DATABASE provadb;  
\c provadb;  
.. /* Istruzioni che creano tabelle. */  
.. /*Istruzioni che inseriscono tuple. */
```

Possiamo caricare le suddette istruzioni SQL inviando il file `prova.sql` a psql mediante con il comando:

```
\i 'C:\\MiaCartella\\provadb.sql'
```

**NB Ricordatevi di inserire gli apici semplici e i doppi slash correttamente (come nell'esempio suddetto).**

A questo punto dovrebbero comparire tutti i messaggi relativi alla creazione del database, alla creazione delle tabella, e all'inserimento delle tuple nella tabella.

---

## 63.6 Bibliografia

<https://www.diag.uniroma1.it/~lenzerin/index.html/files/GuidaPostgreSQLWindows.pdf>