

Sommario

- Panoramica
- 1. Installazione PostgreSQL
- 2. Configurazione pgAdmin
- 3. Setup MSYS2 per Compilazione C
- 4. Creazione e Gestione Database
- 5. Sviluppo Applicazioni C con libpq
- 6. Risoluzione Errori Comuni
- Appendice: Script Automatizzati

Panoramica

Questa guida fornisce un percorso completo per configurare un ambiente di sviluppo PostgreSQL su Windows, utilizzando:

- **PostgreSQL**: Server database relazionale
- **pgAdmin**: Interfaccia grafica per gestione database
- **MSYS2**: Ambiente Unix-like per compilazione con MinGW

In particolare, è orientata allo sviluppo in linguaggio C con accesso al database tramite la libreria `libpq`.

1. Installazione PostgreSQL

1.1 Download e Installazione

1. Scarica PostgreSQL da: <https://www.postgresql.org/download/windows/>
2. Avvia l'installer e seleziona:
 - ☒ PostgreSQL Server
 - ☒ pgAdmin 4
 - ☒ Command Line Tools
3. Imposta una password per l'utente `postgres` (es. `root`)
4. Mantieni la porta 5432 (default)

1.2 Verifica installazione

- Avvia **pgAdmin 4** e accedi con la password impostata
- Verifica che il servizio `postgresql-x64-17` sia attivo (`services.msc`)

2. Configurazione pgAdmin

2.1 Primo avvio

- Avvia pgAdmin 4
- Imposta una master password se richiesto
- Crea un nuovo server:
 - Host: localhost
 - Porta: 5432
 - Database: postgres
 - Username: postgres
 - Password: quella scelta

2.2 Creazione Database

- Clic destro su "Databases" > Create > Database
- Nome: LotoBianco
- Owner: postgres

2.3 Esecuzione script SQL

- Apri Query Tool
- Carica il file db2versione.sql
- Esegui con ►

Assicurati di:

- Aggiungere prima i dati in indirizzo se esiste un vincolo su cliente

3. Setup MSYS2 per Compilazione C

3.1 Installazione MSYS2

- Scarica da <https://www.msys2.org>
- Installa in C:\msys64
- Avvia MSYS2 MinGW 64-bit

3.2 Aggiornamento pacchetti e installazione

```
pacman -Syu          # aggiorna pacchetti
gcc --version        # verifica compilatore
pacman -S mingw-w64-x86_64-toolchain
pacman -S mingw-w64-x86_64-postgresql
```

3.3 Verifica setup

```
ls /mingw64/include/libpq-fe.h
ls /mingw64/lib/libpq.dll.a
```

4. Creazione e Gestione Database

4.1 Ordine corretto di esecuzione

1. Crea tabella `indirizzo`
2. Inserisci indirizzi validi
3. Crea tabella `cliente`
4. Solo dopo inserisci i clienti

4.2 Verifica

Nel `Query Tool`, esegui:

```
SELECT * FROM cliente;
SELECT * FROM trattamento;
```

5. Sviluppo Applicazioni C con libpq

5.1 Compilazione e esecuzione

Posizionati nella cartella del file `.c`:

```
cd /c/Users/TUO_UTENTE/Desktop
gcc MainDataBase.c -lpq -o MainDataBase.exe
./MainDataBase.exe
```

5.2 Script `build.sh`

```
#!/bin/bash
if [ ! -f "/mingw64/include/libpq-fe.h" ]; then
    echo "libpq-fe.h mancante. Installa con pacman -S mingw-w64-x86_64-
postgresql"
    exit 1
fi

gcc MainDataBase.c -lpq -o MainDataBase.exe && ./MainDataBase.exe
```

6. Risoluzione Errori Comuni

Errore: libpq-fe.h: No such file or directory

- Usa MSYS2 MinGW
- Installa mingw-w64-x86_64-postgresql

Errore: libpq.dll is missing

- Copia da /mingw64/bin nella cartella dell'eseguibile

Errore: autenticazione con password fallita

- Verifica la #define PASSWORD nel codice

Errore: la relazione non esiste

- Esegui correttamente lo script SQL in pgAdmin

Errore: violazione di chiave esterna

- Inserisci prima le righe nella tabella referenziata (es. indirizzo)

Appendice: Script Automatizzati

Script check_env.sh

```
#!/bin/bash

if ! command -v gcc > /dev/null; then
    echo "GCC non installato"
else
    echo "GCC OK"
fi

if [ -f "/mingw64/include/libpq-fe.h" ]; then
    echo "libpq-fe.h trovato"
else
    echo "libpq-fe.h mancante"
fi

psql -U postgres -c "SELECT version();" 2>/dev/null && echo "Connessione OK"
|| echo "Connessione fallita"
```

Questa versione estesa ti guida passo-passo nella configurazione completa, compilazione e debug. Per approfondimenti:

- [Documentazione PostgreSQL](#)
- [MSYS2](#)
- [pgAdmin](#)