

实验 01：在 \$HOME 上配置 PostgreSQL 服务器 (中文完整版)

适用平台：Linux / macOS / WSL (无需 root)。

目标：在 \$HOME 下本地编译安装 PostgreSQL，并初始化、启动、连接与关闭数据库；附常见问题排查。

一、实验目标

- 在 \$HOME 文件系统下建立所需目录
- 在 \$HOME 安装一个可用的 PostgreSQL 数据库服务器 (以 12.5 为例，其他版本步骤一致)
- 完成基本启停、创建数据库与导入示例数据
- 掌握日志定位与常见故障排查

建议完成时间：第 3 周周中前。

二、准备与说明

PostgreSQL 的组成主要有三块：

1. 源码与编译产物 (约 150MB)
2. 已安装可执行文件 (如 pg_ctl、psql, 约 20MB)
3. 数据目录 (含配置、WAL 与数据库本体, 至少 35MB; 放在 SSD 上体验更好)

在类 Unix 系统中，PostgreSQL 可以用普通用户直接运行，无需 root 权限。

三、速查清单 (熟手可直接按此)

```
cd "$HOME"
tar xvj postgresql-12.5.tar.bz2
cd postgresql-12.5
./configure --prefix="$HOME/pgsql"
make
make install

# 配置环境 (一次性创建，之后每个终端 source 一下即可)
cat > "$HOME/env" <<'EOF'
# ----- PostgreSQL 本地环境 (可按需修改) -----
export PGROOT="$HOME/pgsql"
# 默认把数据放在 $HOME/pgsql/data (稳定持久)；
# 如果你想把数据放在 /tmp (更快，但会话结束记得清理)，把下一行注释掉并启用 /tmp 那行。
export PGDATA="$PGROOT/data"
# export PGDATA="/tmp/pgsql.$USER"

# 端口避免冲突；多个同学共用机器时可改成 5433、5434 ...
export PGPORT=5432
```

```
# Unix 套接字目录: 建议与 PGDATA 保持一致
export PGHOST="$PGDATA"

# 路径配置
export PATH="$PGROOT/bin:$PATH"
export LD_LIBRARY_PATH="$PGROOT/lib:${LD_LIBRARY_PATH}"

# 常用别名
alias pgstart='pg_ctl -D "$PGDATA" -l "$PGDATA/log" start'
alias pgstop='pg_ctl -D "$PGDATA" stop'
alias pgrestart='pgstop && sleep 1 && pgstart'
EOF

# 启用环境
source "$HOME/env"

# 初始化与基础配置
initdb

# 编辑 $PGDATA/postgresql.conf (下面给出建议配置片段)
# listen_addresses = ''           # 只用 Unix 域套接字, 避免端口/网络冲突
# max_connections = 10
# max_wal_senders = 4             # 建议 <= max_connections
# unix_socket_directories = '<PGDATA 的绝对路径>'

# 启动、验证与停止
pgstart
psql -l
pgstop
```

四、详细步骤（新手建议跟着做）

1. 解压与编译安装

```
cd "$HOME"
tar xjf postgresql-12.5.tar.bz2           # 解压源码包
cd postgresql-12.5
./configure --prefix="$HOME/pgsql"        # 指定安装位置
make                                       # 编译（视机器性能数分钟）
make install                             # 安装到 $HOME/pgsql
```

如果 `configure` 报缺少依赖，安装常见构建工具（Ubuntu 示例）：

```
sudo apt-get update
sudo apt-get install -y build-essential libreadline-dev zlib1g-dev
```

在无法使用 `sudo` 的环境，可联系管理员预装依赖，或选用院系提供的构建容器。

2. 配置环境脚本 `env`

上文速查清单已展示了一个可直接使用的 `env` 脚本模板：

```
cat "$HOME/env"
source "$HOME/env"      # 每开新终端都要执行一次
which psql && which pg_ctl
```

若你希望把数据放到 `/tmp`（更快的 I/O）：

- 打开 `env`，将 `export PGDATA="$PGROOT/data"` 注释掉
- 取消注释 `export PGDATA="/tmp/pgsql.$USER"`
- 之后每次会话结束请手动清理 `/tmp/pgsql.$USER` 目录

3. 初始化数据目录并首启

```
source "$HOME/env"
initdb                                     # 初始化数据目录
```

编辑 `$PGDATA/postgresql.conf`：

```
# 仅示例；请将 <绝对路径> 替换为你的 $PGDATA 的绝对路径
listen_addresses = ''                      # 禁用 TCP，仅 Unix 套接字
max_connections = 10
max_wal_senders = 4
unix_socket_directories = '<绝对路径>'
```

启动与验证：

```
pgstart
psql -l      # 应能看到 postgres、template0、template1 等库
pgstop
```

4. 日常使用命令

```
# 每次登录后：
source "$HOME/env"

# 启动数据库（后台）
pgstart

# 查看已有数据库
psql -l

# 创建并进入一个新库
createdb mydb
psql mydb

# 常用 psql 命令：
```

```
# \dt      列出表
# \d <表名> 查看表结构
# \q      退出 psql

# 关闭数据库
pgstop
```

5. 导入示例数据（可选：啤酒样例库）

假设当前目录有 `beer.dump`：

```
createdb beer
psql beer -f beer.dump
psql beer -c "select count(*) from beers;"
```

五、常见问题与排错

1) 启动失败

- 查看日志：

```
tail -20 "$PGDATA/log"
```

- 典型报错与处理：

- `postmaster.pid` 已存在：先检查是否真的在运行；若未运行，删除遗留 PID 与套接字：

```
rm -f "$PGDATA/postmaster.pid"
rm -f "$PGDATA/.s.PGSQL.*"
```

- `could not bind socket` / 端口占用：

- 确保 `listen_addresses = ''`（只走 Unix 套接字）
- 或在 `env` 中修改 `PGPORT`（如 5433），并确保 `postgresql.conf` 与 `PGHOST` 设置一致

- `psql: could not connect to server: No such file or directory`：

- 确认 `PGHOST` 与 `postgresql.conf` 的 `unix_socket_directories` 指向同一目录
- 启动后检查该目录下是否生成 `.s.PGSQL.$PGPORT` 文件

2) 无法停止 / 重启

- 可能有其他终端的 `psql` 会话仍连接中，先 `\q` 退出
- 必要时使用：

```
pgstop
sleep 1
pgstart
```

3) /tmp 放数据的注意事项

- 会话结束或系统清理 /tmp 后，数据会消失；请在每次新会话重新 initdb 或改回 \$HOME/pgsql/data。

六、附：建议的 postgresql.conf 片段（复制时去掉注释）

```
# 基本连接
listen_addresses = ''                # 仅 Unix 套接字
port = 5432                          # 与 PGPORT 保持一致

# 资源与并发（教学/本地开发场景）
max_connections = 10
shared_buffers = 128MB
work_mem = 4MB
maintenance_work_mem = 64MB

# WAL 与复制（如不需要可保持默认）
max_wal_senders = 4

# 套接字目录
unix_socket_directories = '<你的 PGDATA 绝对路径>'
```

七、提交与验收（如课程有要求）

- 提交以下信息的截图或日志：
 1. which psql 与 pg_ctl --version 输出
 2. psql -l 的结果（能看到默认库）
 3. 成功创建数据库 createdb mydb 并使用 psql mydb 连接与简单查询
 4. tail -20 "\$PGDATA/log" 部分日志（掩盖个人路径即可）

八、FAQ

Q：我已经通过包管理器装过 PostgreSQL，和本地编译的会冲突吗？

A：不会。我们把 PATH 放在 \$HOME/pgsql/bin 的前面，优先使用你自己安装的版本。

Q：如何完全清理这个本地安装？

A：停止服务后删除 \$HOME/pgsql（或 /tmp/pgsql.\$USER）目录，并删除 env 文件即可。

Q：可以选择更新版本吗？

A：可以，步骤相同，把压缩包与版本号替换为目标版本即可（注意数据目录兼容性）。