

MySQL用户权限配置指南

概述

本系统实现了**两层权限控制**:

1. **应用层权限** - Streamlit登录时的权限控制
2. **数据库层权限** - MySQL用户级别的权限控制（可选）

通过设置 config.py 中的 USE_DB_ROLES 参数来选择使用哪种方式。

快速部署步骤

第1步：导入主数据库

```
# 在PowerShell中执行  
mysql -u root -p < "d:\数据库\小组大作业\大作业.sql"  
# 输入密码: Ljqhcsk1@
```

或使用Navicat导入 大作业.sql 文件。

第2步：创建MySQL用户和权限

```
# 执行用户权限脚本  
mysql -u root -p < "d:\数据库\小组大作业\创建用户和权限.sql"  
# 输入密码: Ljqhcsk1@
```

这个脚本会创建以下MySQL用户：

用户名	密码	MySQL角色	权限说明
fbp_user	fbp123456	FBPRole	管理所有基础表（Model, Country, Exchange, Costs等），查看Display
manager_user	manager123456	ManagerRole	查看聚合视图（s_Display, s_Display_Model, s_Display_Country）
sales_india	india123456	SalespersonIndiaRole	只能查看和操作印度数据（DisplayIndia, Sales_Price_India）
sales_pakistan	pakistan123456	SalespersonPakistanRole	只能查看和操作巴基斯坦数据
sales_south_africa	southafrica123456	SalespersonSouthAfricaRole	只能查看和操作南非数据

用户名	密码	MySQL角色	权限说明
sales_kenya	kenya123456	SalespersonKenyaRole	只能查看和操作肯尼亚数据

第3步：配置应用

编辑 `app/config.py`:

```
# 启用数据库层权限控制
USE_DB_ROLES = True # 改为 True

# 如果要使用应用层权限（更简单）
# USE_DB_ROLES = False # 改为 False
```

第4步：安装依赖并运行

```
cd "d:\数据库\小组大作业\app"
pip install -r requirements.txt
streamlit run app.py
```

两种模式对比

模式A：数据库层权限控制 (`USE_DB_ROLES = True`)

特点：

- ✓ 更安全：每个角色使用独立的MySQL用户
- ✓ 更专业：展示了完整的数据库权限设计
- ✓ 真实场景：模拟企业级权限管理
- ✗ 需要额外配置MySQL用户

登录账号（Streamlit应用）：

- FBP / fbp123456
- Manager / manager123456
- Salesperson_India / india123456
- Salesperson_Pakistan / pakistan123456
- Salesperson_SouthAfrica / southafrica123456
- Salesperson_Kenya / kenya123456

演示重点：

- 不同用户登录后只能看到对应权限的数据
- 印度业务员无法查看其他国家数据
- 管理者可以看到对比分析
- 财务可以管理成本费用

模式B：应用层权限控制 (`USE_DB_ROLES = False`)

特点：

- 更简单：统一使用root账号
- 易部署：不需要创建MySQL用户
- 够用：对于小型项目足够
- 安全性较低：所有共用一个数据库账号

登录账号：

- Manager / manager123456
- FBP / fbp123456
- Salesperson_India / india123456
- Salesperson_Pakistan / pakistan123456
- Salesperson_SouthAfrica / southafrica123456
- Salesperson_Kenya / kenya123456

测试权限

测试1：管理者权限

```
# 使用管理者账号登录MySQL
mysql -u manager_user -pmanager123456 大作业-test4

# 应该能成功
SELECT * FROM s_Display;
SELECT * FROM s_Display_Model;
SELECT * FROM s_Display_Country;

# 应该失败（无权限）
SELECT * FROM Sales_Price;
SELECT * FROM Costs;
```

测试2：印度业务员权限

```
# 使用印度业务员账号登录
mysql -u sales_india -pindia123456 大作业-test4

# 应该能成功
SELECT * FROM DisplayIndia;
SELECT * FROM Sales_Price_India;

# 应该失败（无权限）
SELECT * FROM DisplayPakistan; # 不能看其他国家
SELECT * FROM Display; # 不能看总体数据
SELECT * FROM s_Display; # 不能看管理者数据
```

测试3：财务BP权限

```
# 使用财务BP账号登录
mysql -u fbp_user -pfbp123456 大作业-test4

# 应该能成功
SELECT * FROM Display;
SELECT * FROM Costs;
SELECT * FROM Regional_Expenses;
INSERT INTO Costs VALUES (25, 'DA 128+8', 'India', '2026-02', 78.00);

# 应该失败 (Sales_Price 只读)
UPDATE Sales_Price SET Sales = 100 WHERE id = 1;
```

视图和表的对应关系

管理者可见：

- `s_Display` - 历史预测预算对比（全部）
- `s_Display_Model` - 按产品汇总
- `s_Display_Country` - 按国家汇总

财务BP可见：

- `Display` - 总体经营情况
- `Model, Country, Exchange` 等基础表
- `Costs, Ratio_Expenses1/2/3, Regional_Expenses` 等财务表

各国业务员可见：

- `DisplayIndia / DisplayPakistan / DisplaySouthAfrica / DisplayKenya`
- `Sales_Price_India / Sales_Price_Pakistan` 等（可修改）

清理脚本

如果需要删除创建的用户：

```
-- 删除所有创建的用户和角色
DROP USER IF EXISTS 'fbp_user'@'%';
DROP USER IF EXISTS 'manager_user'@'%';
DROP USER IF EXISTS 'sales_india'@'%';
DROP USER IF EXISTS 'sales_pakistan'@'%';
DROP USER IF EXISTS 'sales_south_africa'@'%';
DROP USER IF EXISTS 'sales_kenya'@'%';

DROP ROLE IF EXISTS 'FBPRole';
DROP ROLE IF EXISTS 'ManagerRole';
```

```
DROP ROLE IF EXISTS 'SalespersonIndiaRole';
DROP ROLE IF EXISTS 'SalespersonPakistanRole';
DROP ROLE IF EXISTS 'SalespersonSouthAfricaRole';
DROP ROLE IF EXISTS 'SalespersonKenyaRole';

FLUSH PRIVILEGES;
```

部署完成！

现在系统具有：

- 完整的数据库表结构
- 多层视图设计
- MySQL角色权限控制
- Streamlit应用集成
- 灵活的配置选项

推荐配置: `USE_DB_ROLES = True` (展示完整功能)