Real Estate DBMS - Complete Setup Guide

Prerequisites

Before starting, ensure you have:

- Python 3.9 or higher installed
- MySQL 8.0 or higher installed and running
- MySQL Workbench (optional, for database management)
- Git (optional, for version control)

Step 1: Database Setup

1.1 Import the Database Schema

bash

```
# Option A: Using MySQL Workbench
# 1. Open MySQL Workbench
# 2. Connect to your MySQL server
# 3. File → Run SQL Script
# 4. Select your real_estate_schema.sql file
# 5. Click "Run"

# Option B: Using Command Line
mysql -u root -p < real_estate_schema.sql

# Enter your MySQL root password when prompted
```

1.2 Verify Database Creation

```bash
mysql -u root -p

# In MySQL prompt:
USE real_estate_db;
SHOW TABLES;
# You should see 10 tables

# Check sample data
SELECT COUNT(*) FROM Properties;
SELECT COUNT(*) FROM Users;
```

1.3 Create Django Database User (Recommended)

```sql
-- In MySQL prompt
CREATE USER 'django_user'@'localhost' IDENTIFIED BY 'django_password123';
GRANT ALL PRIVILEGES ON real_estate_db.* TO 'django_user'@'localhost';
FLUSH PRIVILEGES;
EXIT;
```

Step 2: Python Environment Setup

2.1 Create Project Directory

```bash
# Create main project folder
mkdir real_estate_project
cd real_estate_project
```

2.2 Create Virtual Environment

```bash
# Windows
python -m venv venv
venv\Scripts\activate

# macOS/Linux
python3 -m venv venv
source venv/bin/activate
```

2.3 Install Required Packages

```bash
# Install Django and dependencies
pip install django==4.2
pip install mysqlclient
pip install python-dotenv
pip install pillow
pip install django-crispy-forms
pip install crispy-bootstrap5

# Save requirements
pip freeze > requirements.txt
```

Note: If `mysqlclient` installation fails on Windows:

```bash
# Alternative: Install PyMySQL instead
pip install pymysql

# Then add this to your settings.py or manage.py:
# import pymysql
# pymysql.install_as_MySQLdb()
```

Step 3: Create Django Project

3.1 Start Django Project

bash

```
# Create Django project
django-admin startproject real_estate_system .

# Note the dot (.) at the end - it creates project in current directory
# Your structure should look like:
# real_estate_project/
#   ├── venv/
#   ├── real_estate_system/
#   │   ├── __init__.py
#   │   ├── settings.py
#   │   ├── urls.py
#   │   ├── asgi.py
#   │   └── wsgi.py
#   ├── manage.py
#   └── requirements.txt
```

3.2 Create Django App

bash

```
# Create the main application
python manage.py startapp properties

# Your structure now:
# real_estate_project/
#   ├── venv/
#   ├── real_estate_system/
#   ├── properties/
#   │   ├── migrations/
#   │   ├── __init__.py
#   │   ├── admin.py
#   │   ├── apps.py
#   │   ├── models.py
#   │   ├── tests.py
#   │   └── views.py
#   ├── manage.py
#   └── requirements.txt
```

Step 4: Configure Django Settings

4.1 Create .env File (Database Credentials)

Create a file named `.env` in your project root:

```
env

# .env file
DB_NAME=real_estate_db
DB_USER=django_user
DB_PASSWORD=django_password123
DB_HOST=localhost
DB_PORT=3306
SECRET_KEY=your-secret-key-here-change-this-in-production
DEBUG=True
```

4.2 Update settings.py

Open `real_estate_system/settings.py` and make these changes:

python

```python
# At the top of settings.py
import os
from pathlib import Path
from dotenv import load_dotenv

# Load environment variables
load_dotenv()

# Build paths inside the project
BASE_DIR = Path(__file__).resolve().parent.parent

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = os.getenv('SECRET_KEY', 'django-insecure-default-key-change-this')

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = os.getenv('DEBUG', 'False') == 'True'

ALLOWED_HOSTS = ['localhost', '127.0.0.1']


# Application definition
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'properties',  # Our app
    'crispy_forms',
    'crispy_bootstrap5',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'real_estate_system.urls'
```

```
TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [BASE_DIR / 'templates'],  # Add this
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

WSGI_APPLICATION = 'real_estate_system.wsgi.application'


# Database Configuration
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': os.getenv('DB_NAME', 'real_estate_db'),
        'USER': os.getenv('DB_USER', 'root'),
        'PASSWORD': os.getenv('DB_PASSWORD', ''),
        'HOST': os.getenv('DB_HOST', 'localhost'),
        'PORT': os.getenv('DB_PORT', '3306'),
        'OPTIONS': {
            'init_command': "SET sql_mode='STRICT_TRANS_TABLES'",
            'charset': 'utf8mb4',
        },
    }
}


# Password validation
AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
```

```
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]


# Internationalization
LANGUAGE_CODE = 'en-us'
TIME_ZONE = 'America/New_York'  # Boston timezone
USE_I18N = True
USE_TZ = True


# Static files (CSS, JavaScript, Images)
STATIC_URL = 'static/'
STATICFILES_DIRS = [BASE_DIR / 'static']
STATIC_ROOT = BASE_DIR / 'staticfiles'

# Media files (User uploaded content)
MEDIA_URL = 'media/'
MEDIA_ROOT = BASE_DIR / 'media'


# Default primary key field type
DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'


# Crispy Forms Configuration
CRISPY_ALLOWED_TEMPLATE_PACKS = "bootstrap5"
CRISPY_TEMPLATE_PACK = "bootstrap5"


# Login/Logout redirects
LOGIN_URL = 'login'
LOGIN_REDIRECT_URL = 'home'
LOGOUT_REDIRECT_URL = 'home'
```

Step 5: Create Directory Structure

bash

```
# Create necessary directories
mkdir templates
mkdir static
mkdir static/css
mkdir static/js
mkdir static/images
mkdir media
mkdir media/property_images

# Your final structure:
# real_estate_project/
#   ├── venv/
#   ├── real_estate_system/
#   ├── properties/
#   ├── templates/
#   ├── static/
#   │   ├── css/
#   │   ├── js/
#   │   └── images/
#   ├── media/
#   │   └── property_images/
#   ├── manage.py
#   ├── requirements.txt
#   └── .env
```

Step 6: Test Database Connection

bash

```
# Test if Django can connect to MySQL
python manage.py dbshell

# If successful, you'll enter MySQL prompt
# Type: SHOW TABLES;
# You should see your 10 tables
# Type: EXIT; to quit
```

Step 7: Verify Setup

```bash
# Run Django development server
python manage.py runserver

# You should see:
# Starting development server at http://127.0.0.1:8000/
# Open browser and go to: http://127.0.0.1:8000/

# You'll see Django's default page (it's normal at this stage)
# Press Ctrl+C to stop the server
```

Common Issues & Solutions

Issue 1: mysqlclient Won't Install

Windows:

```bash
# Download unofficial wheel from:
# https://www.lfd.uci.edu/~gohlke/pythonlibs/#mysqlclient
# Then install:
pip install mysqlclient-1.4.6-cp39-cp39-win_amd64.whl
```

Or use PyMySQL:

```bash
pip install pymysql

# Add to manage.py (before if __name__ == '__main__':)
import pymysql
pymysql.install_as_MySQLdb()
```

Issue 2: Access Denied for Database User

```sql
-- In MySQL:
GRANT ALL PRIVILEGES ON real_estate_db.* TO 'django_user'@'localhost';
FLUSH PRIVILEGES;
```

Issue 3: Port Already in Use

```bash
# Use different port
python manage.py runserver 8080

# Or find and kill process using port 8000
# Windows: netstat -ano | findstr :8000
# Linux/Mac: lsof -ti:8000 | xargs kill
```

Issue 4: Module Not Found Errors

```bash
# Ensure virtual environment is activated
# Windows: venv\Scripts\activate
# Mac/Linux: source venv/bin/activate

# Reinstall requirements
pip install -r requirements.txt
```

Next Steps

Once setup is complete:

1. ✅ Database imported and running
2. ✅ Django project created
3. ✅ Virtual environment activated
4. ✅ All packages installed
5. ✅ Settings configured
6. ✅ Database connection tested

You're ready to build the application!

The next files we'll create:

- `properties/models.py` - Django models mapping to MySQL tables
- `properties/views.py` - CRUD operations
- `properties/urls.py` - URL routing
- `templates/*.html` - Frontend templates
- `properties/forms.py` - Data input forms

Quick Start Commands Reference

```bash
# Activate virtual environment
source venv/bin/activate  # Mac/Linux
venv\Scripts\activate     # Windows

# Run development server
python manage.py runserver

# Create migrations (if models change)
python manage.py makemigrations

# Apply migrations
python manage.py migrate

# Create superuser for admin panel
python manage.py createsuperuser

# Open Django shell
python manage.py shell

# Access database shell
python manage.py dbshell
```

Setup Complete! Ready to code? Let me know and I'll provide the models.py file next!