《数据库实践》期末项目：基于大模型与 MCP 服务的自然语言数据库查询系统

**一、项目背景与目标**

近年来，大语言模型（如通义千问、ChatGPT 等）已能基于自然语言生成结构化查询语言（SQL），在此背景下，本实验旨在引导学生综合运用数据库知识、服务端开发技能与大模型调用技术，完成一个基于 MCP 协议的数据查询系统，并探索其功能扩展与优化方式。

**目标包括：**

1. 部署并理解 MCP 服务结构；
2. 编写与LLM API交互模块，实现自然语言→SQL生成；
3. 与 MCP 服务联通，执行查询并返回结果；
4. 增强 MCP 功能（如查询日志、分页、白名单校验）；
5. 实现 CLI 或 GUI 查询界面；
6. 探索 Prompt 优化方法以提升 SQL 正确率或效率。

**二、任务说明**

基础任务（60--70分）

|  |  |
| --- | --- |
| **子任务** | **要求说明** |
| MCP 服务运行 | 下载并部署 alexcc4/mcp-mysql-server，连接 MySQL 实例 |
| 通义 API 调用模块 | 输入自然语言 → 输出 SQL；支持基础 prompt 构造 |
| 查询控制模块 | 获取 schema，执行 SQL，解析并返回 JSON 结果 |
| CLI 界面实现 | 可在终端交互输入自然语言并返回查询结果 |

MCP功能增强任务（加分，20分）

|  |  |
| --- | --- |
| **功能项** | **实现说明** |
| 查询日志记录 /logs | MCP Server 记录每次执行的 SQL 和时间戳 |
| 查询结果分页 | 长查询结果支持用户在 CLI 输入 next 或自动分页返回 |
| 表结构简化输出 | /schema 支持按表名过滤返回 schema |

MCP安全控制任务（加分，20分）

|  |  |
| --- | --- |
| **安全项** | **实现说明** |
| 只读 SQL 白名单过滤 | MCP 内部解析 SQL，仅允许 SELECT 语句 |
| 关键字段访问控制 | 禁止查询包含 password、salary 等字段 |
| 简易 SQL 注入防御机制 | 拦截明显拼接注入或关键词注入的攻击行为 |

大模型优化任务/UI扩展任务（加分，20分）

|  |  |
| --- | --- |
| **优化项** | **实现说明** |
| Prompt 模板优化 | 提高生成 SQL 的准确率（准确性提升 ≥10% 可得满分） |
| 多轮提示结构 / 示例增强 Few-shot | 在 prompt 中引入示例对 / 对话上下文优化 |
| SQL 执行计划简化建议 | 提示模型生成更高效的 SQL 查询结构（如避免子查询嵌套） |
| GUI 界面（如 Streamlit） | 可输入自然语言，展示生成 SQL 和查询结果表格 |

**三、参考资源**

**1. MCP基础代码**：MCP已有开源项目完成mysql的连接和执行查询的代码，可以参考已有开源代码自己完成针对mysql的基础和其他MCP相关功能，也可以根据mcp教程中的sqlite例子自己重新写一个针对sqlite数据库的代码以使用sqlite文件完成本项目。  
参考开源代码：<https://github.com/alexcc4/mcp-mysql-server/tree/master>  
 <https://github.com/meanands/mysql-mcp>

教程参考内容：https://blog.csdn.net/heian\_99/article/details/147253836?ops\_request\_misc=%257B%2522request%255Fid%2522%253A%2522e81e89967328cd9a16cacc3d1065186a%2522%252C%2522scm%2522%253A%252220140713.130102334.pc%255Fall.%2522%257D&request\_id=e81e89967328cd9a16cacc3d1065186a&biz\_id=0&utm\_medium=distribute.pc\_search\_result.none-task-blog-2~all~first\_rank\_ecpm\_v1~rank\_v31\_ecpm-11-147253836-null-null.142^v102^pc\_search\_result\_base2&utm\_term=Python%20mcp%E9%A1%B9%E7%9B%AE&spm=1018.2226.3001.4187

教程示例代码：https://github.com/modelcontextprotocol/servers/tree/main/src/sqlite

**2.** **通义千问API文档：**期中项目中注册并且使用过Qwen额度，如果免费额度用完了大家可以通过别的平台API，如果没用完可以继续使用。

<https://help.aliyun.com/zh/model-studio/use-qwen-by-calling-api>  
 **3. Prompt编写引导：**可以参考这些内容编写任务说明和提示语言，让大模型根据用户需求完成增删改查的SQL代码。

<https://github.com/dair-ai/Prompt-Engineering-Guide>

4. **MCP教程：**可以参考官方文档的quickstarts和example部分，仿照编写

https://modelcontextprotocol.io/introduction

**四、提交要求**

* 完整项目目录（含代码、配置、运行说明）
* 运行说明（README.md）
* 实验报告（PDF 或 Markdown），需包含以下内容：

1. 项目概述，包括代码结构说明
2. MCP 功能增强设计说明（加分选做）
3. 大模型交互流程与 prompt 优化方法（加分选做）
4. 测试案例（不少于 5 条）与结果展示
5. 困难与解决方案

**五、项目检查内容**

|  |  |
| --- | --- |
| **项目维度** | **分数** |
| MCP Server 配置与理解 | 10 |
| 大模型 API 交互调用 | 15 |
| 基础的SQL 查询准确性 | 20 |
| UI 或 CLI 界面完整度 | 10 |
| 测试用例设计与分析（参见六） | 15 |
| 项目报告与总结 | 10 |
| 拓展功能（增删改、多轮、prompt设计、正确测试结果） | 10 -- 20 |
|  |  |

**六、测试生成正确率的用例：**

**使用数据库**：Lab01中的college数据库

基础测试用例（10分）：  
**## Question 10: List the names of all courses ordered by their titles and credits. (easy)**

**### Question 16: What is the title, credit value, and department name for courses with more than one prerequisite? (medium)**

**### Question 23: What are the names of students who have more than one advisor? (medium)**

**### Question 33: What are the titles of courses without prerequisites? (hard)**

其他用例可以从之前的实验问题中选测，也可以自己编写问题测试（5--10分）。