

**

数据库系统

课程试验报告

|  |  |  |
| --- | --- | --- |
| **试验题目： MiniOB数据库管理系统功能的补充与完善** | | |
|  | 学生姓名： | 丁海桐 |
|  | 学生学号： | 202226010304 |
|  | 专业班级： | 软件2203班 |
|  | 开课时间： | 2023-2024-2学期 |
|  | 试验日期： | 2024年4月13日 |

|  |
| --- |
| 1. **试验任务：**   1）实现对 date 这一数据类型的支持；  2）实现聚合运算（sum， avg，count，max， min）； |
| 1. **试验准备情况（对照任务，实验之前给出你的预案）：**   **SQL层理解**  当用户在 miniob 中输入 SQL 语句时，该请求以字符串形式存储：   1. Parser 阶段：将 SQL 字符串进行词法解析（lex\_sql.l）和语法解析（yacc\_sql.y）， 最终转换为 ParsedSqlNode（parse\_defs.h）。 2. Resolver 阶段：将 ParsedSqlNode 转换为 Stmt（Statement），进行语义解析。 3. Transformer 和 Optimizer 阶段：将 Stmt 转换为 LogicalOperator，并在优化后输 出 PhysicalOperator。这一阶段负责执行查询优化操作。 4. 对于命令执行类型的 SQL 请求，系统会创建相应的 CommandExecutor。 5. 执行阶段 Executor：将 PhysicalOperator（物理执行计划）转换为 SqlResult（执行 结果），或者通过 SqlResult 输出 CommandExecutor 执行后的结果。 |
| 1. **试验过程记录（对照任务，对试验方案和结果进行记录和分析）：** 2. **实现Date数据结构**   在lex\_sql.l中添加DATE的token    在ycc\_sql.y中添加DATE\_T    在parse\_defs.h中添加SQlCommandFlag的枚举    在value.h中添加enum的枚举，并构造date的set和get函数      相对应的在value.cpp中实现date的set函数      在set\_value中添加DATES的case    在miniob-2023\deps\common\time中添加date.h和date.cpp        至此，DATE实现就已经完成。  **（二）实现聚合函数**  在yacc\_sql.cpp中添加函数get\_aggr\_func\_type    并且添加相应token    在parse\_def.h中添加枚举AggrFuncType    在src/observer/sql/expr/expression.cpp中实现func的get\_type和get\_value函数      在src/observer/sql/expr/tuple.h中实现具体过程      至此聚合函数结束。 |
| 1. **试验完成情况：** 2. **DATE验证**   创建一个date\_test（id int, bir date）表，并插入一项数据，查询验证。     1. **聚合函数验证**   创建func\_test(sal int)表，并插入一些数据。    验证五个聚合函数 |
| 1. **试验总结：（遇到的问题及解决措施，对试验的评价，感想和认识）**   实验难度很大，期间遇到了很多问题，比如在实现聚合函数时候，为了不破坏代码框架以及不影响后续实现，要谨慎决定代码之间的联系，后来在src/observer/sql/expr/tuple.h进行函数的实现，新创建的文件不影响代码结构，只是提供了接口。  工程代码庞大，所以阅读源码是一项很很艰巨的任务，要耐下性子认真读，可以从main中开始，也可以从SQL的层次出发，分部分阅读。期间网络上有一些比赛心得可以参考。 |