

# 任务描述

## Task 1: 底层存储接口实现

- **Step1:** 根据注释提示完善 MemConnect 模块相关函数补齐, 实现上层功能与存储模块的交互接口
  - **目标:** 补全 `edu.whu.tmbd.query.operations.utils.MemConnect` 功能代码
  - **描述:** MemConnect是数据库查询执行和底层存储交互的中间类, 内部包含大量的系统表接口, 本任务要求学生根据代码注释完成接口补全。
  - **难度:** 简单

## Task 2: 类级别操作

- **Step1:** show table功能实现: 本项目数据库采用对象代理模型, 其中对象及其代理之间的关系采用系统表来实现。
  - **目标:** 补全文件 `edu.whu.tmbd.util.DbOperation.java` 中相关系统表展示功能函数
  - **描述:** 实现以下指令执行
    - `show ClassTable`: 展示ClassTable中的"class name", "class id", "attribute name", "attribute id", "attribute type"
    - `show DeputyTable`: 展示DeputyTable中的"origin class id", "deputy class id"
    - `show SwitchingTable`: 展示SwitchingTable中的"origin class id", "origin attribute id", "origin attribute name", "deputy class id", "deputy attribute id", "deputy attribute name"
    - `show BiPointerTable`: 展示双向指针表中的"class id", "object id", "deputy id", "deputy object id"
  - **难度:** 简单
- **Step2:** create deputy class功能实现 (可参考create class相关逻辑)
  - **目标:** 补全文件 `edu.whu.tmbd.query.operations.impl.CreateDeputyClassImpl.java` 中相关创建代理类的函数, 实现代理类创建功能
  - **描述:** 对象代理下, 类和代理之间的关系通过系统表维护, 因此在创建代理类时不仅要根据代理类型创建新的ClassTable并插入元组, 还需要更新维护DeputyTable、SwitchingTable和BiPointerTable。具体地, 补全以下函数:
    - `createDeputyClass`: 创建并插入新的classTableItem和switchingTableItem
    - `createDeputyTableItem`: 创建并插入新的deputyTableItem
    - `createBiPointerTableItem`: 插入对象, 创建并插入新的biPointerTableItem
  - **难度:** 中等
- **Step3:** drop class功能实现
  - **目标:** 补全文件 `edu.whu.tmbd.query.operations.impl.DropImpl` 中相关函数, 实现数据库drop class功能
  - **描述:** 对象代理下, 类和代理之间的关系通过系统表维护, 因此在删除一个类时, 需要同时删除类的对象、类相关的系统表, 并递归删除以该类为源类的代理类。
  - **难度:** 中等

## Task 3: 查询执行

- **Step1: 带表达式的select实现 (允许对函数逻辑进行重构)**

- **目标:** 补全文件 `edu.whu.tmdb.query.operations.impl.SelectImpl.java` 中的 `projectSelectExpression` 函数, 使系统支持带表达式的select查询
- **描述:**

```
# 当前系统仅支持形如下述表达式的查询语句
select * from test;
select a.* from a, b;

# 本任务要求补全函数projectAllTableColumns, 使系统支持如下带表达式的查询语句:
select a from test;
select a+b as c from test;
```

- **难度:** 困难

- **Step2: where子句的实现 (参考等式实现逻辑)**

- **目标:** 补全文件 `edu.whu.tmdb.query.operations.impl.where` 中相关代码, 使其支持select查询中的where表达式
- **描述:** 当前系统仅支持where子句的等式表达式, 需要补全<、<=、>、>=的相关逻辑
- **难度:** 简单

- **Step3: where子句中or字段的实现 (参考and字段实现逻辑)**

- **目标:** 补全文件 `edu.whu.tmdb.query.operations.impl.where` 中相关代码, 使where表达式支持or字段的逻辑表达
- **描述:** 当前系统支持where子句的and和in表达式, 需要补全orExpression函数加入or表达式逻辑
- **难度:** 简单

- **Step4: delete语句实现**

- **目标:** 补全文件 `edu.whu.tmdb.query.operations.impl.DeleteImpl.java` 中相关函数, 使数据库支持delete语句
- **描述:** delete语句的目标是删除满足where条件的对象 (元组), 因此在系统表ObjectTable中删除对应对象, 并删除对象在BiPointerTable中的记录。注意, 如果被执行delete语句的类包含其他代理, 也需要一并删除代理类中相关元组。
- **难度:** 中等

- **Step5: limit字段实现**

- **目标:** 补全文件 `edu.whu.tmdb.query.operations.impl.SelectImpl.java` 中limit函数
- **描述:** limit关键字用于限制select查询结果的输出数量, 因此需要在limit函数内实现查询结果的resize
- **难度:** 简单