**武汉大学计算机学院**

**本科生实验报告**

**tmdb实验**

专 业 名 称 ：计算机科学与技术

课 程 名 称 ：数据库系统实现

指 导 教 师 ：王胜

学 生 学 号 ：2021300004054，2021300004042，2021300004048，2021310004047，2021300004055

学 生 姓 名 ：蒋卓鑫，顾田，邢益鸣，戴威扬，黄羿君

二○二三年四月

**郑 重 声 明**

本人呈交的实验报告，是在指导老师的指导下，独立进行实验工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本实验报告不包含他人享有著作权的内容。对本实验报告做出贡献的其他个人和集体，均已在文中以明确的方式标明。本实验报告的知识产权归属于培养单位。

本人签名： 日期： 2024.05.17



摘 要

本实验采用代码填空的形式进行，学生需要根据已有的接口定义、系统注释、参考文档实现一个对象代理模型的数据库内核，使其支持基本的create、show、drop等类级别操作和insert、select、delete、update等查询执行语句。

**关键词：**tmdb；数据库

目 录

[1 底层存储接口实现](#_Toc166875641)

[1.1 底层存储接口实现 5](#_Toc166875642)

[2 类级别操作](#_Toc166875643)

[2.1 show table功能实现 6](#_Toc166875644)

[2.1.1 show ClassTable 6](#_Toc166875645)

[2.1.2 show DeputyTable 6](#_Toc166875646)

[2.1.3 show SwitchingTable 6](#_Toc166875647)

[2.1.4 show BiPointerTable 6](#_Toc166875648)

[2.2创建代理类功能实现 7](#_Toc166875649)

[2.2.1 createDeputyClass 7](#_Toc166875650)

[2.2.2 createDeputyTableItem 7](#_Toc166875651)

[2.2.3 createBiPointerTableItem 8](#_Toc166875652)

[2.3 drop class功能实现 8](#_Toc166875653)

[2.3.1 drop 8](#_Toc166875654)

[2.3.2 dropClassTable 9](#_Toc166875655)

[2.3.3 dropDeputyClassTable 9](#_Toc166875656)

[2.3.4 dropBiPointerTable 9](#_Toc166875657)

[2.3.5 dropSwitchingTable 9](#_Toc166875658)

[2.3.6 dropObjectTable 10](#_Toc166875659)

[3 查询执行](#_Toc166875660)

[3.1 带表达式的SELECT语句实现 11](#_Toc166875661)

[3.2 where子句的实现 12](#_Toc166875662)

[3.3 where子句中or字段的实现 12](#_Toc166875663)

[3.4 delete语句的实现 12](#_Toc166875664)

[3.5 limit字段实现 13](#_Toc166875665)

# 1 底层存储接口实现

## 1.1 底层存储接口实现

根据注释将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。主要补充的函数内容如下：

1. **getClassId根据类名获取对应类ID**

遍历类列表并在类名与提供参数相同时返回该类ID classid。遍历完不存在此类名则抛出异常。

1. **getColumns根据类名获取对应属性名称列表**

遍历类列表并在类名与提供参数相同时将属性名attrname添加到定义的属性名称列表中。若遍历完属性列表为空则抛出异常，若找到了对应的属性则返回属性列表。

1. **getClassAttrnum根据类名/类ID获取属性数量**

遍历类列表并在类名/类ID与提供参数相同时返回该类属性数量attrnum。遍历完不存在此类名则抛出异常。

1. **getAttridList根据类ID和属性名称列表获取对应属性ID数组**

遍历类列表并在类ID与提供参数相同时将属性ID attrid添加到定义的属性ID列表中。若遍历完属性ID列表为空则抛出异常，若找到了对应的属性ID则将其转为int数组返回。

1. **getAttrid根据类ID和属性名称获取对应属性ID**

遍历类列表并在类ID和属性名称与提供参数相同时返回该属性ID attrid。遍历完不存在此类名则抛出异常。

1. **classExist根据类名判断是否存在该类名的表**

遍历类列表并在类名与提供参数相同时返回true。

1. **columnExist根据类名和属性名判断是否存在该属性**

遍历类列表并在类名和属性名与提供参数相同时返回true。

1. **getDeputyIdList根据源类ID获取对应的所有代理类ID**

遍历类列表并在源类ID与提供参数相同时将代理类ID deputyid添加到定义的Integer类型ArrayList中，遍历完返回。

通过以上的补充，实现了部分MemConnect内部的系统表接口。

# 2 类级别操作

## 2.1 show table功能实现

要展示相关表的相应信息，需要引入task1中Memconnect模块中的相应方法。

### 2.1.1 show ClassTable

首先通过调用MemConnect类的getClassTableList()方法获取类表数据，然后通过for each循环遍历类表中的每个元素。使用System.out.println()和System.out.printf()方法打印出类表的标题行和每一行的数据，左对齐，占据20个字符的空间。类表的每一行包含类名、类ID、属性名、属性ID和属性类型。

### 2.1.2 show DeputyTable

首先通过调用MemConnect类的getDeputyTableList()方法获取代理类表，然后通过for each循环遍历列表中的每个元素，使用printf方法格式化输出源类ID和代理类ID。首先输出表头，然后输出每个元素的数据。

### 2.1.3 show SwitchingTable

首先通过MemConnect类的getSwitchingTableList()方法获取交换表项列表。然后，打印表头。接着使用for each循环遍历每个元素，并使用printf方法以对齐的方式打印出每个元素的属性信息。

### 2.1.4 show BiPointerTable

首先通过调用MemConnect类的getBiPointerTableList()方法获取双向指针表的列表。然后，打印出表头。最后，使用for each循环遍历双向指针表的每个元素，并使用printf方法按照指定格式打印出每个元素的各个属性值，即class id、object id、deputy id和deputy object id。左对齐，占据20个字符的空间，属性值之间用空格分隔。

## 2.2创建代理类功能实现

### 2.2.1 createDeputyClass

该函数的主要功能是创建并插入新的classTableItem和switchingTableItem。

根据输入的参数创建一个代理类，并在类表和交换表中分别添加相应的信息。具体步骤如下：

（1）生成新的classId，即类ID。

（2）遍历selectResult，根据其中的信息创建ClassTableItem对象，并将其添加到ClassTableList中。

（3）根据selectResult中的信息，创建SwitchingTableItem对象，并将其添加到SwitchingTableList中。

（4）返回生成的类ID。

在该函数中，主要用到了MemConnect类中的静态方法，如getClassTable()、getClassTableList()、getSwitchingTableList()等，用于获取类表、类表列表和交换表列表等信息。同时，为了获取selectResult中的属性ID、属性名、属性类型和类名等信息，还用到了SelectResult类的getAttrid()、getAttrname()、getType()、getClassName()等方法。

### 2.2.2 createDeputyTableItem

该函数的功能是创建并插入新的deputyTableItem。

通过传入的className数组、deputyType和deputyId参数，创建一个DeputyTableItem对象，并将其添加到MemConnect的deputyTableList中。具体实现过程如下：

（1）使用Arrays.stream()将className数组转换为流，然后使用Collectors.toCollection()方法将其收集到HashSet中，以去除重复元素。

（2）遍历去重后的className集合，对于每个className，通过memConnect.getClassId()方法获取其对应的classId。

（3）使用classId、deputyId和将deputyType转换为字符串数组的新数组，创建一个DeputyTableItem对象。

（4）最后，通过MemConnect.getDeputyTableList().add()方法将DeputyTableItem对象添加到deputyTableList中。

### 2.2.3 createBiPointerTableItem

该函数主要功能是创建并插入新的biPointerTableItem。

根据传入的SelectResult对象和deputyId，创建并插入BiPointerTableItem对象到BiPointerTable列表中。具体步骤如下：

（1）创建InsertImpl实例用于执行插入操作。

（2）调用Arrays.asList()方法将selectResult的属性名由字符串数组转换为列表形式。

（3）遍历selectResult中的元组。

（4）对于每个元组：使用insert.execute()方法插入代理类对象，并获取插入后的代理对象ID。调用getOriginClass()方法去除重复元素后将结果保存在origin集合中。

（5）然后遍历origin集合，对于每个元素：调用memConnect.getClassId()方法获取源类ID。根据元组信息创建BiPointerTableItem对象。通过MemConnect.getBiPointerTableList().add()将BiPointerTableItem插入到BiPointerTable列表中。

## 2.3 drop class功能实现

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。

### 2.3.1 drop

功能：删除给定类ID的所有相关数据，包括类表、代理类表、双向指针表、切换表和对象表。

实现：通过调用各个函数还删除各个类表，最后通过删除代理类表时获取的代理类id删除相关数据。

其中递归删除代理类相关部分的具体实现方法如下：

1. 通过if判断是否存在需要删除的代理类；

2. 若不存在则输出信息；若存在则遍历代理类ID列表，移除内存表中对应的项，输出删除信息。

### 2.3.2 dropClassTable

功能：给定要删除的class id，删除系统表类表(class table)中的表项。

实现：从内存表中检索所有记录,尝试移除与指定ID匹配的记录, 根据删除结果输出信息。

### 2.3.3 dropDeputyClassTable

功能：删除系统表中的deputy table，并获取class id对应源类的代理类id。

实现：遍历代理类表，删除与源类ID匹配的项，同时收集代理类ID；如果是源类，则将该项的代理类id添加到代理类id列表中再删除；如果是代理类，则直接删除。

### 2.3.4 dropBiPointerTable

功能：删除系统表中的BiPointerTable。

实现：尝试从bi指针表中移除与给定classId匹配的项，根据删除结果输出信息。

### 2.3.5 dropSwitchingTable

功能：删除系统表中的SwitchingTable。

实现：尝试从Switching表中移除与给定classId匹配的项，根据删除结果输出信息。

### 2.3.6 dropObjectTable

功能：删除源类具有的所有对象的列表

实现：通过传入的classId参数确定要删除的对象列表，使用MemConnect.getObjectTableList().removeIf来删除列表中满足条件（即类id等于classId）的所有对象，根据删除结果输出信息。

# 3 查询执行

## 3.1 带表达式的SELECT语句实现

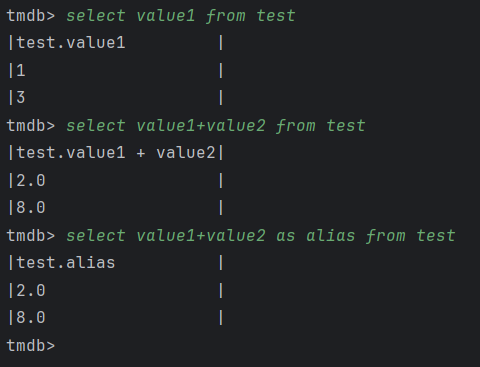
为实现带表达式的SELECT语句查询，我们引入一个名为projectSelectExpression的方法。这个方法在数据库查询操作中处理投影选择表达式，并将完整结果集中的指定属性投影到投影结果集中。具体来说，该算法从整个结果集中提取特定列的数据，并将其填充到投影结果集中，同时处理别名、数据类型和相关的元数据信息。

算法的步骤主要为：

1. **表达式解析和执行**：解析选择的表达式，并执行该表达式以获取数据。具体来说，该方法创建了一个用于存储解析后的表名和列名的TableColumn和存储结果数据的dataList，调用attributeParser方法解析表达式并调用formulaExecute方法执行表达式。
2. **处理属性名和别名**：在处理输出时，如果有别名（alias），例如

a\*b as c

这样的表达式，就需要输出别名c而不是a\*b。这里使用条件语句实现。



1. **更新投影结果**：更新投影结果集projectResult中的相关信息，包括属性名、别名、类型等。
2. **插入查询结果到resTupleList**：将表达式执行的结果插入到投影结果的元组列表中。具体来说，这一步使用循环的方式，遍历resTupleList中的每个元组，将dataList中的数据插入到相应的位置，并更新元组的大小、ID 和类ID。

通过以上步骤，算法实现了从完整结果集中提取特定列的数据并将其填充到投影结果集中，同时处理属性名和别名，确保投影结果的完整性和一致性。

## 3.2 where子句的实现

参考已给出的equalsToExpression方法，补全<、<=、>、>=的相关逻辑。算法的步骤主要为：

1. **处理参数**：使用getLeftExpression()和getRightExpression()处理参数expression，获取expression的左右两部分。
2. **实现逻辑**：compare方法中有compareTo方法来处理对比逻辑，例如当compare(a,b)大于0时即a大于b。当符合逻辑时将对应tuple添加到定义的hashset中。
3. **结果返回**：调用getSelectResultFromSet方法构建selectResult返回。

## 3.3 where子句中or字段的实现

参考已给出的andExpression方法，补全orExpression的相关逻辑。算法的步骤主要为：通过分别执行左右表达式并合并它们的结果，返回一个新的 SelectResult 对象。

其中调用execute 方法根据表达式类型调用不同的处理方法，getTupleSet 方法将 SelectResult 转换为 HashSet<Tuple>，getSelectResultFromSet 方法根据 Tuple 集合创建新的 SelectResult 对象并返回。最后配置execute方法，以确保所有处理方法都被接入。

## 3.4 delete语句的实现

delete语句的目标是删除满足where条件的对象（元组），要求在系统表ObjectTable中删除对应对象，并删除对象在BiPointerTable中的记录；如果被执行delete语句的类包含其他代理，则需要一并删除代理类中相关元组。

实现步骤如下：

1. 删除源类tuple和object table：

初始化列表targetTupleId用于存放待删除的对象Id，用于第2步。

遍历传入的tupleList，对其中每一个tuple执行以下操作：

1. 调用方法memConnect.DeleteTuple()将其删除

2. 使用MemConnect.getObjectTableList().remove()删除对象表。

3. 将其Id加入targetTupleId列表。

2. 删除源类在BiPointerTable中的记录：

初始化列表deputyTupleIdList用于存储待删除的副本元组Id，用于第3步。

遍历BiPointerTableList，对其中每一项进行检查，若其出现在targetTupleId中，则调用MemConnect.getBiPointerTableList().remove()将其从BiPointerTable中删除，并加入deputyTupleIdList。

3. 根据biPointerTable递归删除代理类相关表：

若deputyTupleIdList为空则直接返回。

创建一个新的TupleList，将deputyTupleIdList中所有待删除的副本元组添加到该列表中，然后递归调用delete()方法将其删除。

## 3.5 limit字段实现

limit关键字用于限制select查询结果的数量，需要在limit函数内实现查询结果的resize。

实现步骤如下：

1. 输入合法性检查：limit应为正数，则在limit<=0时抛出异常。

1. **if** (limit <= 0){
2. **throw**  **new** IllegalArgumentException();
3. }

2. 按照limit进行截取：初始化列表originalResult然后截取其子列表。

1. List<Tuple> originalResult = selectResult.getTpl().tuplelist;
2. List<Tuple> finalTuple = originalResult.subList(0, Math.min(limit, originalResult.size()));

3. 处理并返回结果。

教师评语评分

评语：

评分：

评阅人：

年 月 日

（备注：对该实验报告给予优点和不足的评价，并给出百分之评分。）