**标题1**

**\*.3 查询执行**

**\*.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。