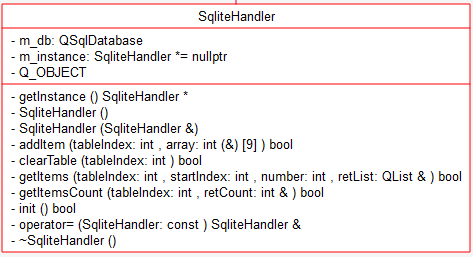
**一、数据库类**

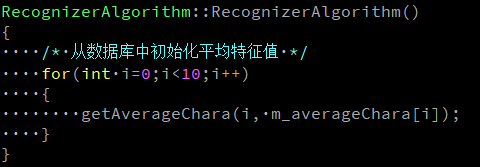
1.选用Sqlite数据库，并将数据库相关的操作封装成一共单例类，数据库类的UML图如下：



2.数据库类封装了数据项的添加，查询，数量获取，表清空操作。

**二、数据库类的使用**

1.算法类构造的时候，会从数据库中获取所有的记录项，并计算平均特征值，存放到**RecognizerAlgorithm::m\_averageChara**数组中

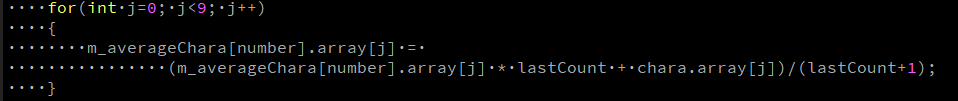


2.为了防止数据库中的表项过多，一次全部加载到内存中会占用太多的内存，采用分段加载的方式，实现函数为RecognizerAlgorithm::getAverageChara



3.RecognizerAlgorithm::*set对数据库的操作*

调用SqliteHandler::addItem*添加一个特征值，添加特征值之后不会重新获取数据库中的所有项并重新计算，而是在现用的特征值的基础上进行计算，大大减少了计算量。*



*4.*RecognizerAlgorithm::*get 不用读取数据库，直接根据***RecognizerAlgorithm::m\_averageChara**数组中的最新特征平均值计算 最小特征距离。