rust\_db

1. 基本架构

本数据库由三个层次组成，语法解析层，语法执行层，数据操作层。main函数的执行流程为，首先读取数据库文件，将文件内容保存在自己底层的数据结构上，之后循环获取要执行的语句，通过语法解析层生成ast树，将ast树传入语法执行层，语法执行层会具体执行该语句。

1. 数据结构
2. 数据库链表。该链表包含多个打开的数据库。
3. 数据库结构。一个数据库结构由以下部分组成：一个数据库头结构，一个模板结构，一个记录链表结构。数据库头结构包含以下信息：数据库的名称，数据库的记录数量等。模板结构是一个模板单元的链表，记录链表结构是一个记录的链表。
4. 模板单元结构。该结构包含单元类型，该属性名称，模板单元属性信息，其中模板单元属性信息又包括：是否可设置为空，是否为主键，是否设置索引，是否unique，具体记录内容的最大位数等。
5. 记录结构。该结构包含该记录的唯一索引值和一个记录单元链表。记录单元链表是记录单元结构的链表。
6. 记录单元结构。该结构包含单元类型记录的属性具体的值，记录单元属性信息，其中单元属性信息又包括：是否为空，具体的位数等。
7. 数据操作层

数据操作层是唯一能调用保存的数据结构的层级，该层主要有六个函数：

create函数用于创建一个数据库结构。该函数接受一个字符串类型和一个数据库模板，根据该模板链表创建一个数据库，返回这个新创建的数据库，如果创建失败返回空。

drop函数用于删除一个数据库结构。该函数接受一个字符串类型，为要删除的数据库的名称，之后找到对应的数据库进行删除，若成功删除返回1，否则返回0。

firstLog函数用于获取该数据库的第一条记录。该函数接受一个数据库，返回该数据库的第一条记录，若该数据库没有记录则返回空。

getLog函数用于获取该数据库已获得记录的下一条记录。该函数接受一个数据库和一条该数据库中的记录，返回该数据库中该条记录的下一条记录，若没有下一条记录则返回空。

insertLog函数用于创建一条新记录。该函数接受一个数据库和一条记录中各个属性的值，该函数新建一条对应的记录，并将该条记录插入到这个数据库中。

deleteLog函数用于删除一条记录。该函数接受一个数据库和数据库中的一条记录，将这条记录从这个数据库中删除。

1. 语法执行层

语法执行层获取要执行的ast树的节点然后解析该ast树，之后根据树的不同调用不同的数据操作层命令获取调用结果，再根据调用结果进行输出。

该层还有一个辅助函数satisfy函数，该函数接受一条记录和一个ast节点（条件语句），判断该语句是否满足该ast节点所对应的条件。

where函数接受一个ast树，返回一个链表，该链表包含满足条件的所有记录。该函数调用firstLog获取数据库的第一条记录，之后循环调用getLog获取下一条记录，对于每条记录调用satisfy函数判断是否满足，若满足则加到链表里。