

课题要求	
一、工具依赖	
二、技术依赖	
三、分析实现	
0. 问题分析	
1. 数据库设计实现	
1.0 初步分析设计	
2. 系统逻辑设计	
2.1 实现语句	
四、技术实现	
1. 综述	
2. 实现效果	
五、问题综述	
六、项目总结	
七、参考文献	
附录	

## 选题综述

模拟数据库设计，主要完成需求分析、概念设计与逻辑设计\*\*

## 课题要求

**项目要求：**自由一组，经阅读文献了解知识后，设计需求，设计数据库，根据项目实际描述进行分析、设计

**课题要求：**

- (1) 设计方案要合理；
- (2) 能基于该方案完成系统要求的功能；
- (3) 设计方案有一定的合理性分析。

## 一、工具依赖

- 1. pycharm
- 2. wps office

## 二、技术依赖

- 1. python
- 2. xlrt和xlrd python库

## 三、分析实现

### 0. 问题分析

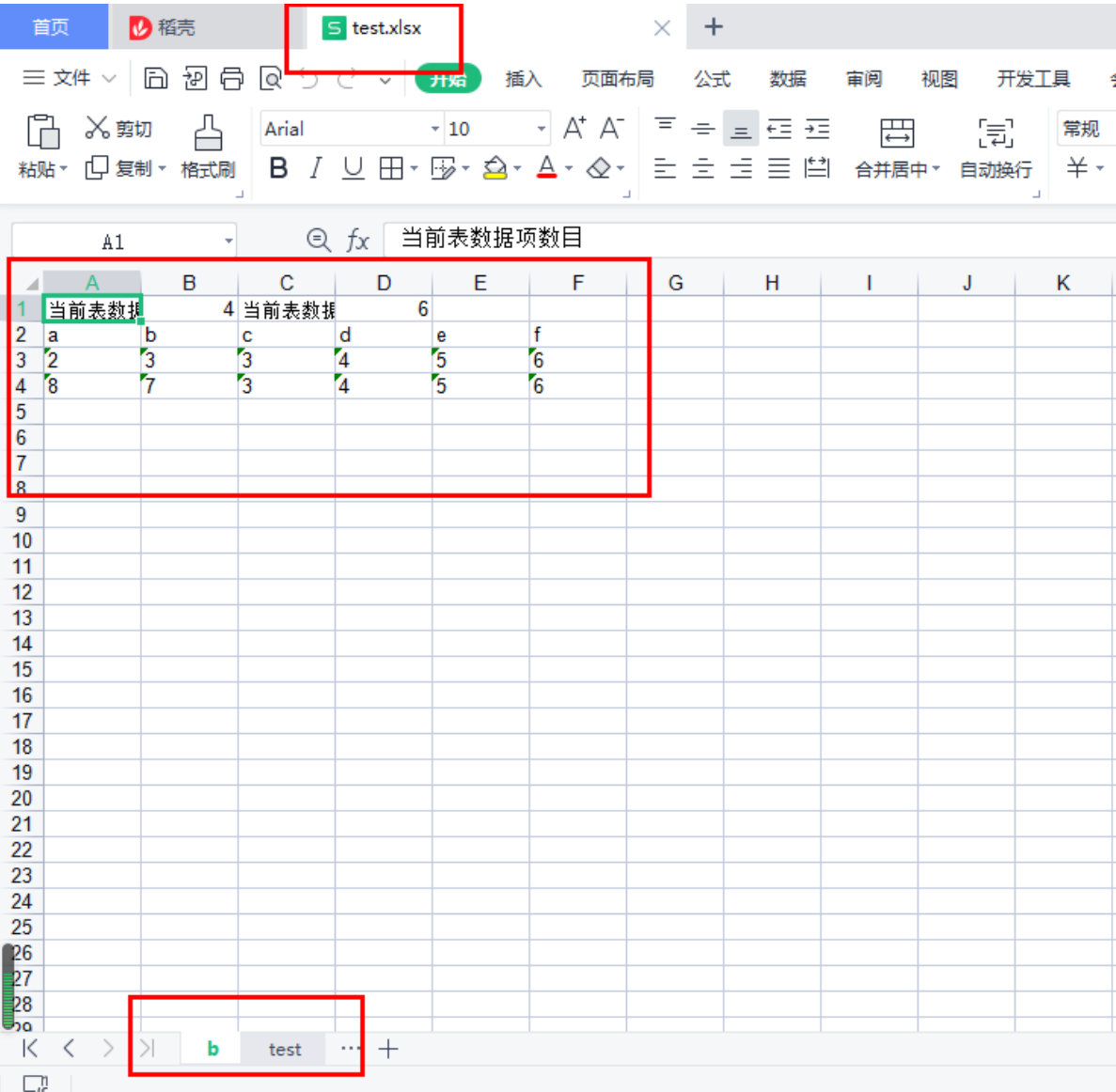
- 需要实现数据库创建、创建表、数据库表增删改查、建立索引并比较效果等

### 1、数据库设计实现

# 1.0 初步分析设计

根据如上分析，需要设计至少如下表，得出如下结论

- 1. 将数据库转换为.xlsx存储
- 2. 将数据库表转换为sheet存储
- 3. 每个数据库都有一个“系统表”，用于存储数据库基本信息
- 4. 每个数据库表sheet，第一行存储数据库表项数+字段数---->便于增删改查操作



当前缺陷：

- 1. 时间原因，暂时只实现：创建数据库、选择数据库、添加数据库表、插入数据、删除数据

## 2.系统逻辑设计

### 2.1 实现语句

实现通过识别sql语句实现，实现如下语句：

- 1. create database [database\_name] 创建数据库，例：

```
1 | create database test
```

2. use [database\_name] 使用数据库，例如：

```
1 | use test
```

3. create table [table\_name] (values) 创建数据库表,例如:

```
1 | create table test
```

4. delete from [table\_name] {where [condition]} 删除数据库表数据，包括删除整个表和选定条目删除，例如：

```
1 | delete from test / delete from test where a=1
```

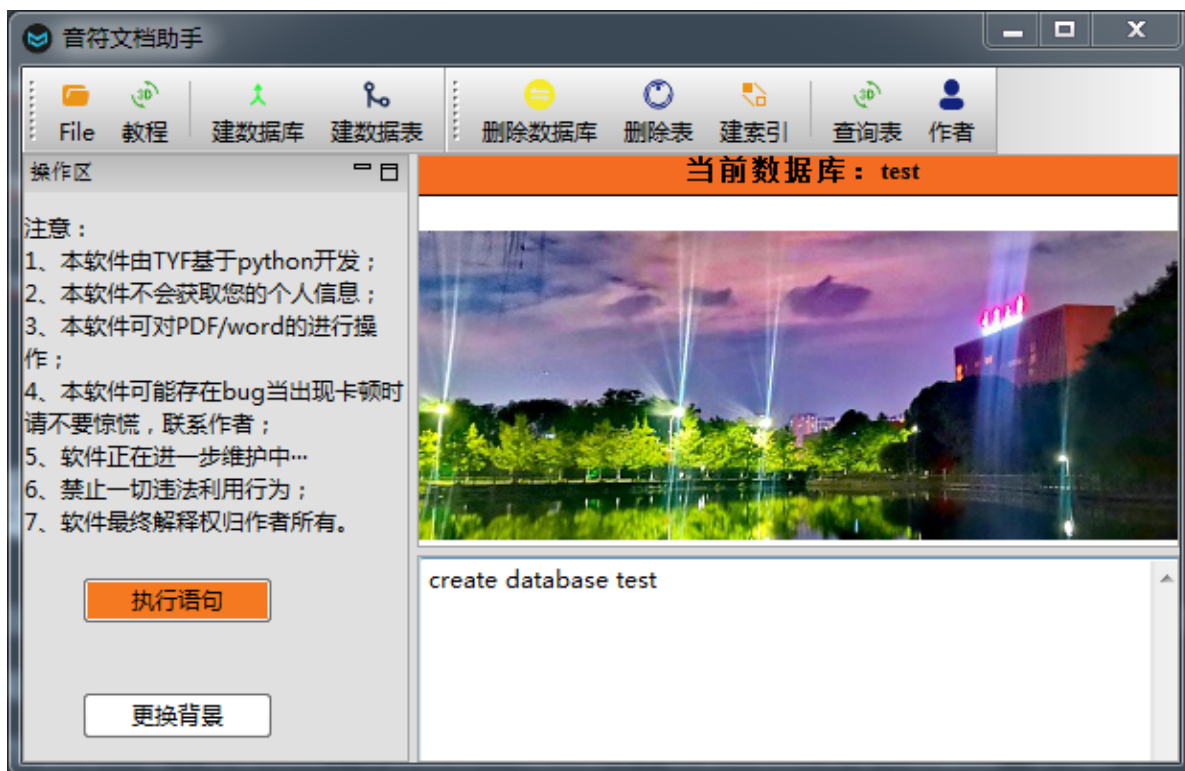
5. insert into [table\_name] (values) 向数据库表插入数据，插入数目与字段数一致，例如：

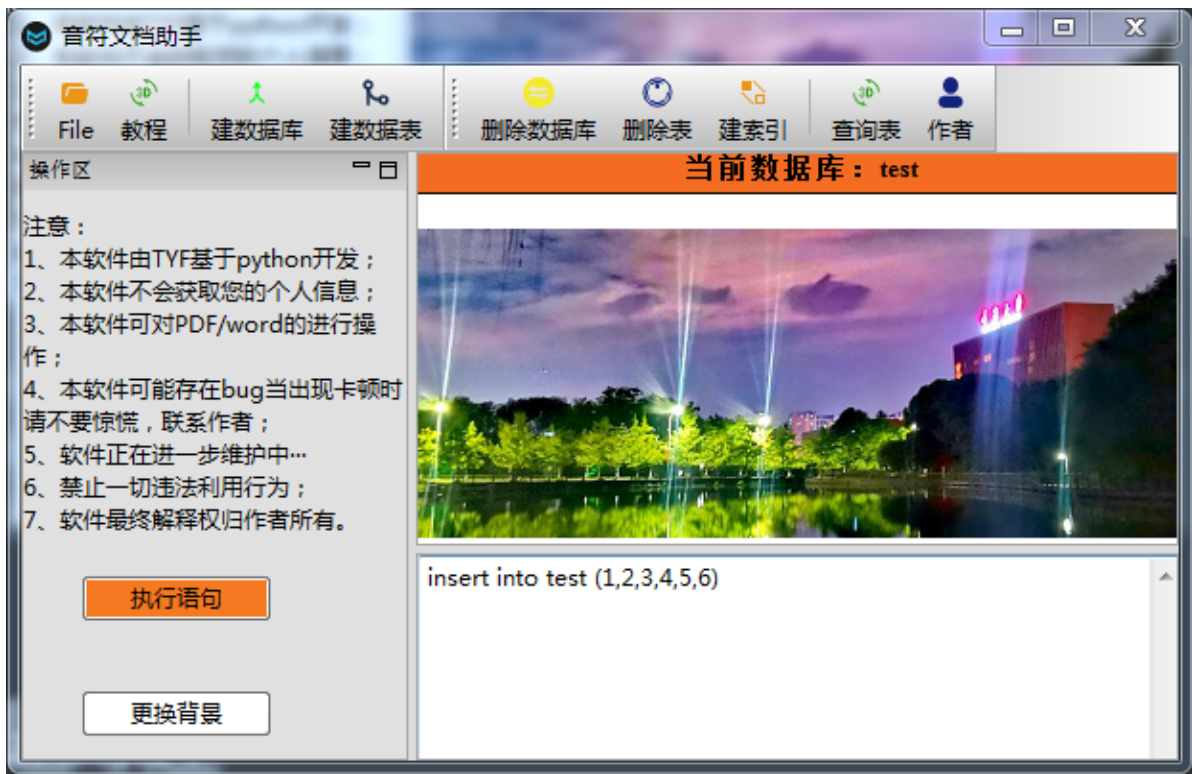
```
1 | insert into test (1,2,3,4,5,6)
```

## 四、技术实现

### 1. 综述

### 2. 实现效果





## 五、问题综述

1. 时间原因，暂时只实现：创建数据库、选择数据库、添加数据库表、插入数据、删除数据

## 六、项目总结

暂无

## 七、参考文献

暂无

## 附录

附录一：初始化数据库所用脚本

```
1 def createTable(self, tableInfo):
2     # open existing workbook
3     rb = xlrd.open_workbook(globalVar.curDatabase+".xlsx",
4 formatting_info=True)
5     # make a copy of it
6     wb = xl_copy(rb)
7     # add sheet to workbook with existing sheets
8     sht1 = wb.add_sheet(tableInfo[2])
9     # 设置字体格式
10    Font0 = xlwt.Font()
11    Font0.name = "Times New Roman"
12    Font0.colour_index = 2
13    Font0.bold = True # 加粗
14    style0 = xlwt.XFStyle()
15
16    sht1.write(0, 0, "当前表数据项数目", style0) # 存储当前表的项数
17    sht1.write(0, 1, 2, style0) # 存储当前表的项数
```

```

17         sht1.write(0, 2, "当前表数据字段数目", style0) # 存储当前表的项
数
18         sht1.write(0, 3, len(tableInfo[3][1:-1].split(',')), style0) # 存储
当前表的项数
19
20         column = 0
21         for i in tableInfo[3][1:-1].split(','):
22             # 添加字段
23             sht1.write(1, column, i, style0)
24             column += 1
25         wb.save(globalVar.curDatabase+".xlsx")
26
27     # insert into table
28     def insertTableItem(self, tableInfo):
29         # open existing workbook
30         rb = xlrd.open_workbook(globalVar.curDatabase + ".xlsx",
formatting_info=True)
31         # make a copy of it
32         wb = xl_copy(rb)
33         # add sheet to workbook with existing sheets
34         sht1 = wb.get_sheet(tableInfo[2])
35         rdSht = rb.sheet_by_name(tableInfo[2])
36         # 设置字体格式
37         Font0 = xlwt.Font()
38         Font0.name = "Times New Roman"
39         Font0.colour_index = 2
40         Font0.bold = True # 加粗
41         style0 = xlwt.XFStyle()
42
43         # 从表中修改当前行数
44         sht1.write(0, 1, rdSht.cell_value(0, 1)+1, style0)
45         rowNum = int(rdSht.cell_value(0, 1))
46
47         column = 0
48         for i in tableInfo[3][1:-1].split(','):
49             # 添加字段
50             sht1.write(rowNum, column, i, style0)
51             column += 1
52         wb.save(globalVar.curDatabase + ".xlsx")
53
54     def deleteTable(self, tableInfo, func):
55         # open existing workbook
56         rb = xlrd.open_workbook(globalVar.curDatabase + ".xlsx",
formatting_info=True)
57         # make a copy of it
58         wb = xl_copy(rb)
59         # add sheet to workbook with existing sheets
60         sht1 = wb.get_sheet(tableInfo[2])
61         rdSht = rb.sheet_by_name(tableInfo[2])
62         # 设置字体格式
63         Font0 = xlwt.Font()
64         Font0.name = "Times New Roman"
65         Font0.colour_index = 2
66         Font0.bold = True # 加粗
67         style0 = xlwt.XFStyle()
68
69         if func == 1:
70             # 从表中修改当前行数

```

```

71         filedNum = int(rdsht.cell_value(0, 3))
72         rowNum = int(rdsht.cell_value(0, 1))
73         sht1.write(0, 1, 2, style0)
74
75         for i in range(2, rowNum):
76             # 添加字段
77             for j in range(0, filedNum):
78                 sht1.write(i, j, '', style0)
79     else:
80         # 从表中修改当前行数
81         rowNum = int(rdsht.cell_value(0, 1))
82         sht1.write(0, 1, rowNum-1, style0)
83
84         filedNum = int(rdsht.cell_value(0, 3))
85
86         filedColumn = -1
87         for index in range(0, filedNum):
88             if str(tableInfo[4].split('=')[0]) ==
str(rdsht.cell_value(1, index)):
89                 filedColumn = index
90
91         if filedColumn == -1:
92             globalVar.status = "无效语句"
93             globalVar.textStatus.SetLabel("当前操作: " +
globalVar.status)
94             return
95
96         isMove = 0
97         for i in range(2, rowNum):
98             # 添加字段
99             if str(tableInfo[4].split('=')[1]) ==
str(rdsht.cell_value(i, filedColumn)):
100                 isMove = 1
101                 if isMove == 1:
102                     if i == rowNum-1:
103                         for j in range(0, filedNum):
104                             sht1.write(i, j, '', style0)
105                         break
106                         for j in range(0, filedNum):
107                             sht1.write(i, j, rdsht.cell_value(i+1, j), style0)
108         wb.save(globalVar.curDatabase + ".xlsx")
109
110     def onSwitch(self, evt):
111         print(self.text_input.GetValue())
112
113         globalVar.curOperation = self.text_input.GetValue().split()
114
115         if globalVar.curOperation[0] in globalVar.sqlStatement:
116             # create
117             if globalVar.sqlStatementMap[globalVar.curOperation[0]] == 0:
118                 if globalVar.curOperation[1] == "database":
119                     self.createDatabase(globalVar.curOperation[2])
120                 elif globalVar.curOperation[1] == "table":
121                     self.createTable(globalVar.curOperation)
122             # insert
123             elif globalVar.sqlStatementMap[globalVar.curOperation[0]] == 4:
124                 self.insertTableItem(globalVar.curOperation)
125             # use

```

```
126         elif globalVar.sqlStatementMap[globalVar.curoperation[0]] == 5:
127             globalVar.curDatabase = globalVar.curoperation[1]
128             globalVar.status = globalVar.curoperation[1]
129             globalVar.textStatus.SetLabel("当前数据
库: "+globalVar.status)
130         elif globalVar.sqlStatementMap[globalVar.curoperation[0]] == 1:
131             if len(globalVar.curoperation) == 3:
132                 self.deleteTable(globalVar.curoperation, 1)
133             elif len(globalVar.curoperation) == 5:
134                 self.deleteTable(globalVar.curoperation, 2)
135
136     else:
137         globalVar.status = "无效语句"
138         globalVar.textStatus.SetLabel("当前操作: " + globalVar.status)
139
140     self.text_input.clear()
141     globalVar.curoperation.clear()
```