

现代操作系统应用开发实验报告

学号： 14331058

班级： 周四班

姓名： 窦耀勇

实验名称： 实验六

一. 参考资料

- UWP 开发随笔——使用 SQLite 数据库

<http://www.cnblogs.com/ms-uap/p/4798269.html>

- 在应用中使用 SQLite 数据库 <http://www.tuicool.com/articles/zQBriuR>

- SQLite 支持的数据类型(日期函

数)<http://blog.csdn.net/kailan818/article/details/5185244>

二. 实验步骤

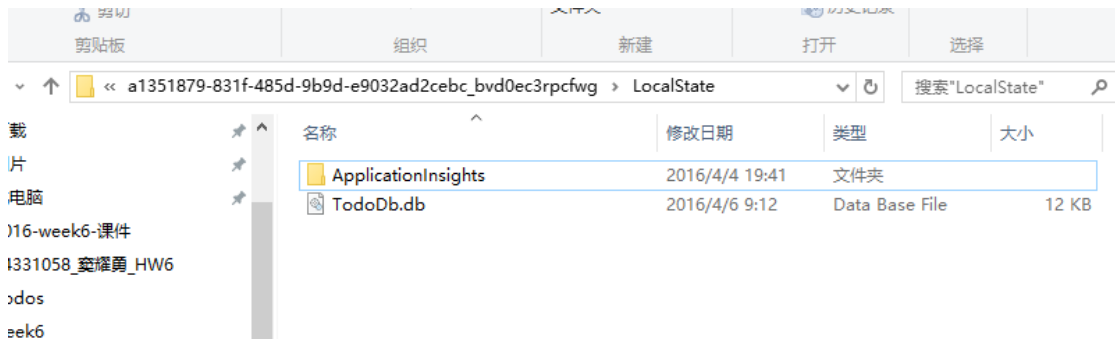
注：本次实验基于上次的实验进行。

1. 在 Mainpage.xaml.cs 中给程序建立数据库

```
private void LoadDatabase()
{
    // Get a reference to the SQLite database
    string sql = @"CREATE TABLE IF NOT EXISTS TodoItem (Id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
                                                         Title VARCHAR( 140 ),
                                                         Context VARCHAR( 140 ),
                                                         Date TEXT);";

    using (var statement = App.TODOdb.Prepare(sql))
    {
        statement.Step();
    }
}
```

运行一下程序，发现在相关文件夹中已经生成了数据库文件



2. 在 MainPage.xaml.cs 中编写数据插入函数和查询函数

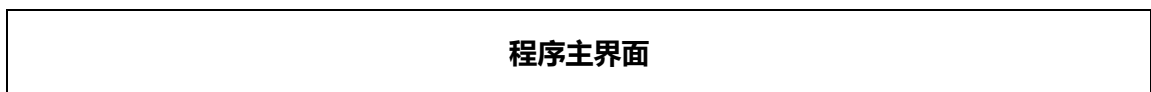
```
public void insert_to_db(object sender, RoutedEventArgs e)
{
    // Insert logic here
}

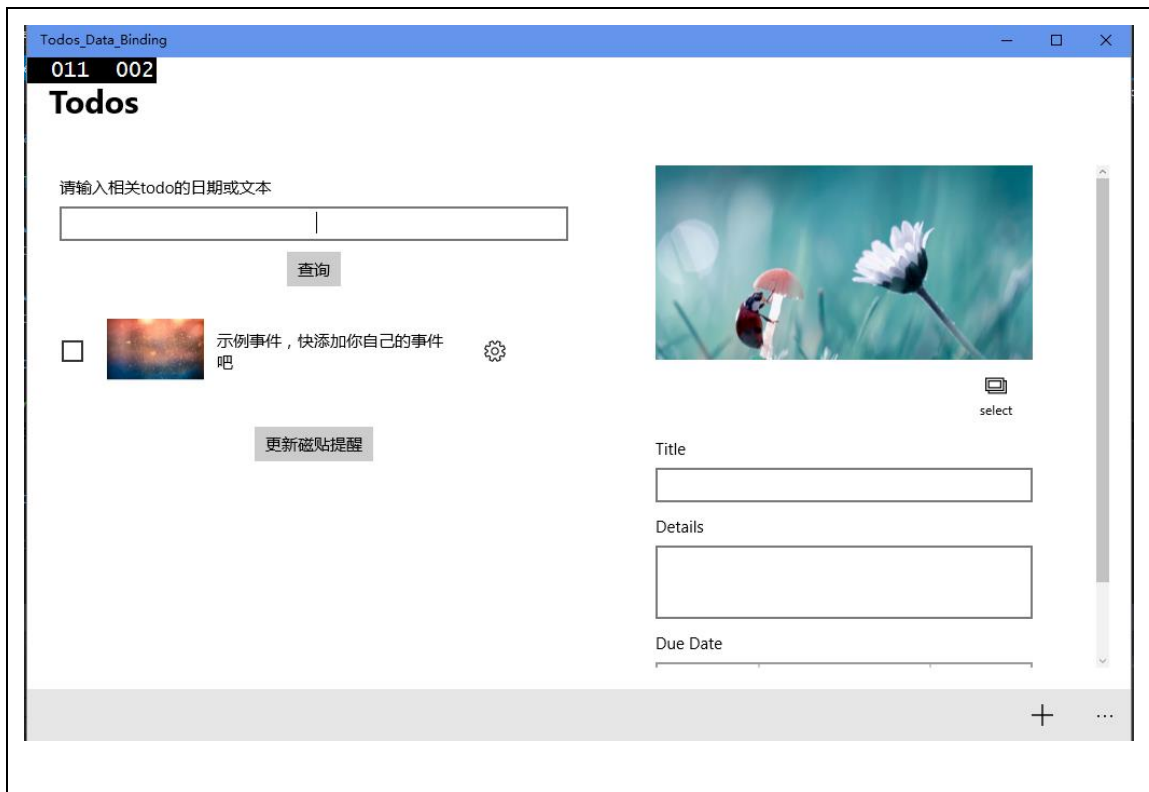
private string query_from_db(object sender, RoutedEventArgs e)
{
    // Query logic here
    ViewModels.TodoItemViewModel ViewModel { get; set; }
}
```

3. 在 MainPage.xaml.cs 中加入，数据持久化和启动读取相关代码

```
// read the data from database into todoitem
var dbconn = App.TodoDb;
using (var statement = dbconn.Prepare("SELECT * FROM TodoItem"))
{
    while (SQLiteResult.ROW == statement.Step())
    {
        ViewModel.AddTodoItem((string)statement[1], (string)statement[2], DateTimeOffset.Parse((string)statement[3]));
    }
}
```

三 . 实验结果截图





给数据库增加记录

1. 给 ToDo 添加时间项

select

Title

购物

Details

买面包

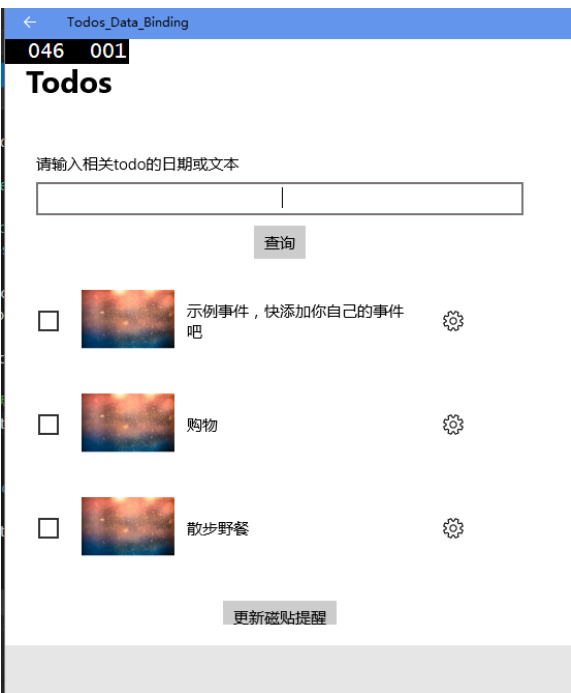
Due Date

2016年 4月 8日

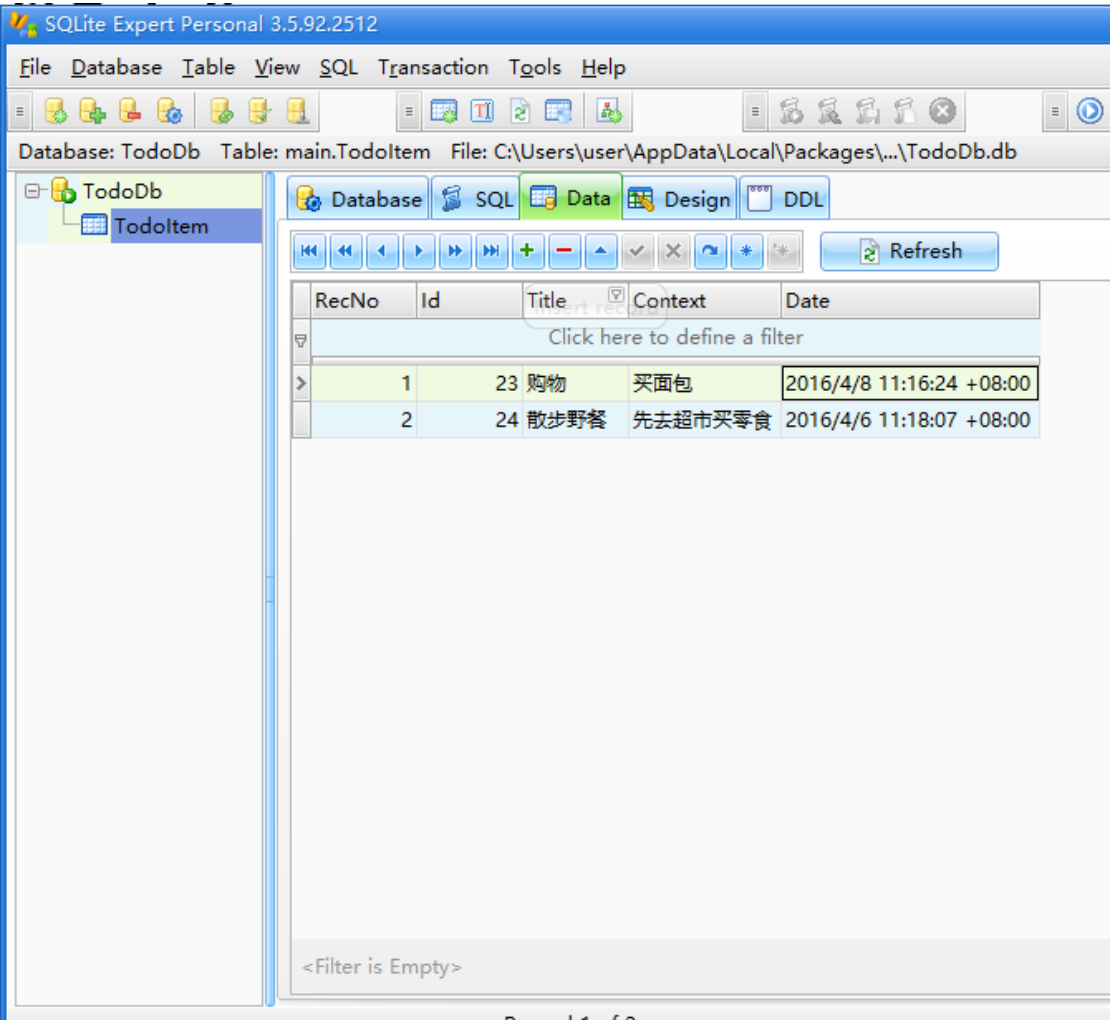
Create Cancel

+

2. 添加了两个项目



3. 查看数据库，发现已经写入数据库文件



查询记录

查询单条记录

← Todos_Data_Binding

012 001

Todos

请输入相关todo的日期或文本

购物

查询

☐



id : 23 ; Title : 购物 ; Context : 买面包 ; Date : 2016/4/8 11:16:24 +08:00

☐



☐



散步野餐

⚙️

更新磁贴提醒

关闭(C)

Details

Due Date

查询多条记录

← Todos_Data_Binding

015 001

Todos

请输入相关todo的日期或文本

2016

id : 23 ; Title : 购物 ; Context : 买面包 ; Date : 2016/4/8 11:16:24 +08:00
id : 24 ; Title : 散步野餐 ; Context : 先去超市买零食 ; Date : 2016/4/6 11:18:07 +08:00

☐



☐



☐



散步野餐

⚙️

更新磁贴提醒

关闭(C)

Details

Due Date

数据库改写和删除

在界面中更新
新 Title

select

Title

购物-看电影

Details

买面包

Due Date

2016年 4月 8日

Update

Cancel

更新结果成功
展示在数据库中

查询

☐ 示例事件，快添加吧

☐ 散步野餐

☐ 购物-看电影

更新磁贴提示

SQLite Expert Personal 3.5.92.2512

File Database Table View SQL Transaction Tools Help

Database: TodoDb Table: main.TodoItem File: C:\Users\user\AppData\Local\Packages\...\TodoDb.db

Database SQL Data Design DDL


Refresh

RecNo	Id	Title	Context	Date
Click here to define a filter				
>	1	24 散步野餐	先去超市买零食	2016/4/6 11:18:07 +08:00
	2	26 购物-看电影	买面包	2016/4/8 11:16:24 +08:00

删除结果也
反映在数据
库中


查询

☐



示例事件，快添加你自己
吧

☐



散步野餐

更新磁贴提醒

Database: ToDoDb Table: main.TodoItem File: C:\Users\user\AppData\Local\Packages\...\ToDoDb.db

Database SQL Data Design DDL

Refresh

RecNo	Id	Title	Context	Date
Click here to define a filter				
>	1	24 散步野餐	先去超市买零食	2016/4/6 11:18:07 +08:00

其他扩展的功能

实现了数据持久化，重新启动软件可以读取数据库中的数据

Todos_Data_Binding

049 001

Todos

请输入相关todo的日期或文本

查询

☐



示例事件，快添加你自己的事件
吧



☐



散步野餐



☐



lalala



更新磁贴提醒



select

Title

Details

Due Date

四 . 实验过程遇到的问题

1. 在写 sql 语句的时候，发现 VS 提示语法错误，然而明明是参照课件的代码所写。

后来请教同学后，发现需要把课件中的代码的空格重新替换才可行，估计是编码的

问题导致的“语法错误”

2. 在编写查询函数时发现 `SQLiteResult.DONE == statement.Step()` 一直返回的是 `false`, 导致无法进入 `if` 语句, 读取查询结果。

后来查询 `Step()` 函数的含义后, 发现 `statement.Step()` 函数的执行是按照在数据库中读取行, 和读取完成的状态返回不同的状态值(`SQLiteResult.DONE` 等, 是枚举类型值)

- 1) 其中读取一个记录返回的是 `SQLiteResult.ROW` ;
- 2) 读取完成(数据库中已经没有符合的记录)返回的才是 `SQLiteResult.DONE`

根据上面的理解, 要把数据库中所有符合的项目取出来, 就需要加一个循环进行遍历 :

```
using (var statement = dbconn.Prepare("SELECT Id, Title, Context, Date FROM TodoItem WHERE Title LIKE ? OR  
{  
    statement.Bind(1, "%" + Query.Text + "%");  
    statement.Bind(2, "%" + Query.Text + "%");  
    statement.Bind(3, "%" + Query.Text + "%");  
    while (SQLiteResult.ROW == statement.Step())  
    {  
        one_todo = one_todo + "id : " + statement[0].ToString() + " ;\tTitle : " + (string)statement[1] + " ;  
    }  
}  
return one_todo;
```

从而解决了问题。

五 . 思考与总结

一个好的工程师应该是能独立解决问题的, 方法总比问题多。例如这次为了查找 `SQLiteResult` 所含的枚举类型值, 没能在官网找到详细的文档, 但是对它单独推敲实验后最后也能找到问题所在。所以用耐性去锻炼解决问题的能力也许可以作为我未来的一个训练方向。