**现代操作系统应用开发实验报告**

**学号：** 14331035 **班级：** 周二班

**姓名：** 陈伟宸 **实验名称：** Todos

1. **参考资料**

<http://edi.wang/post/2016/2/1/windows-10-uwp-back-button-tricks>

<https://social.msdn.microsoft.com/Forums/windowsapps/en-US/ad753362-0d52-42b3-8917-b2897033ba49/getnavigationstate-doesnt-support-serialization-of-a-parameter-type-which-was-passed-to?forum=winappswithcsharp>

<http://www.newtonsoft.com/support>

1. **实验步骤**

1. 修改App.xaml.cs，使程序在挂起并关闭之后重新运行能够回到挂起前的页面

2. 修改TodoItemViewModels.cs，添加用于保存数据和读取数据的函数

3. 在NewPage.xaml.cs中调用保存数据和读取数据的函数，保存挂起时的数据，并在程序重新运行后重新加载数据

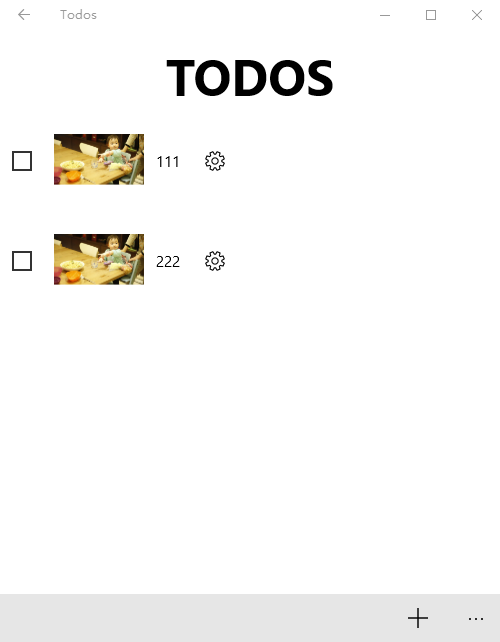
4. 在App.xaml.cs中调用保存所有数据的读取所有数据的函数，使程序关闭后，再次运行时之前的数据存在。

1. **实验结果截图**

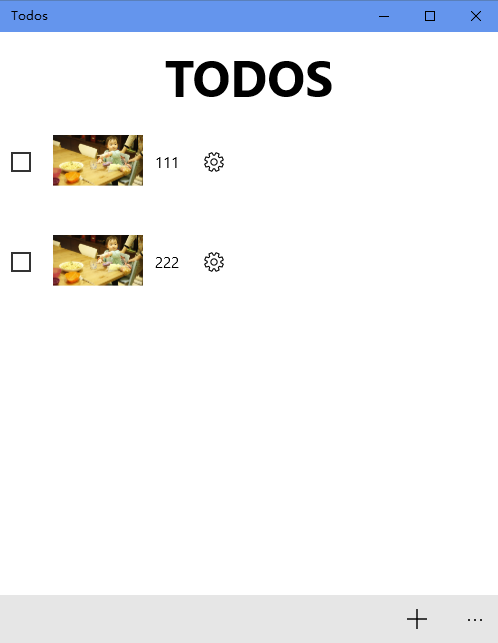
1. 运行程序，此时并未添加任何任务



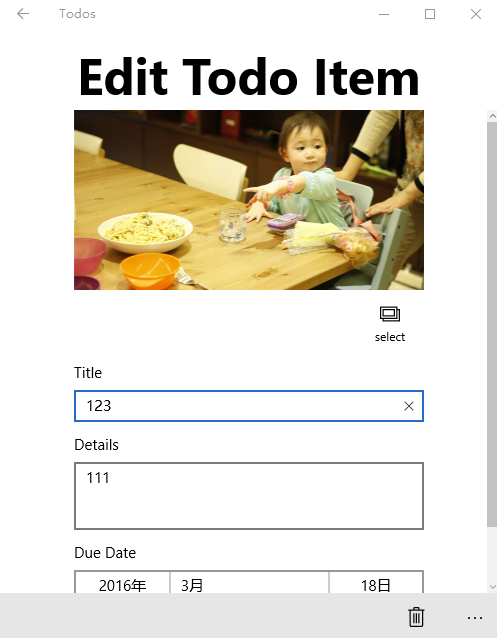
2. 添加两个任务



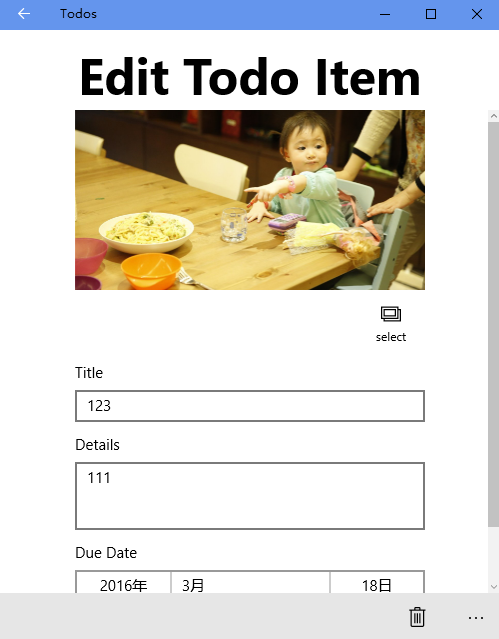
3. 关闭程序，再次运行，之前的添加的任务仍然存在



4. 点击第一个任务，修改标题为123

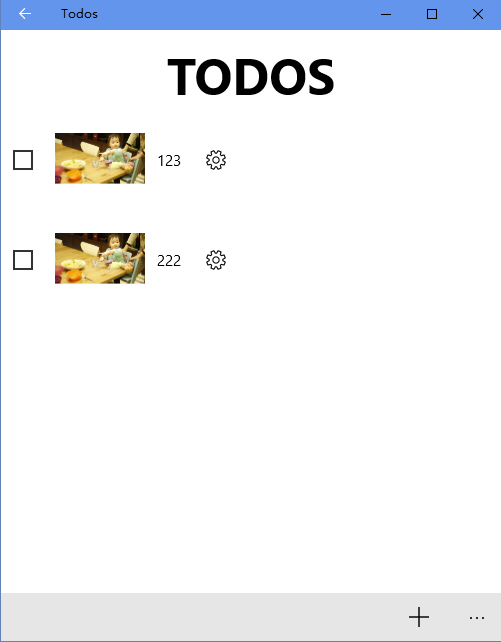


5. 将程序挂起并关闭，再次运行，



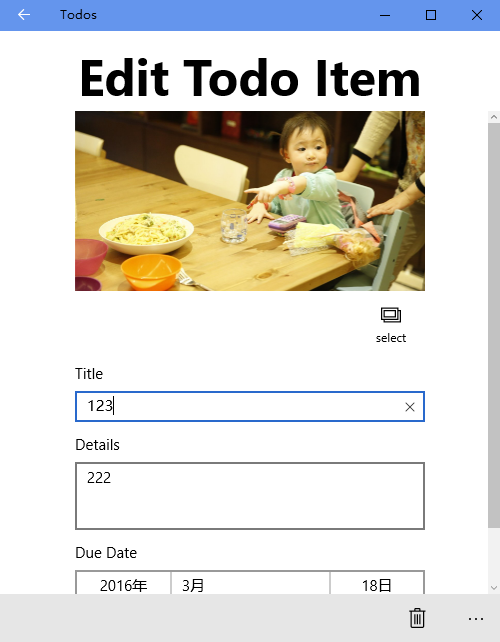
修改的数据依然存在

6. 点击Update，

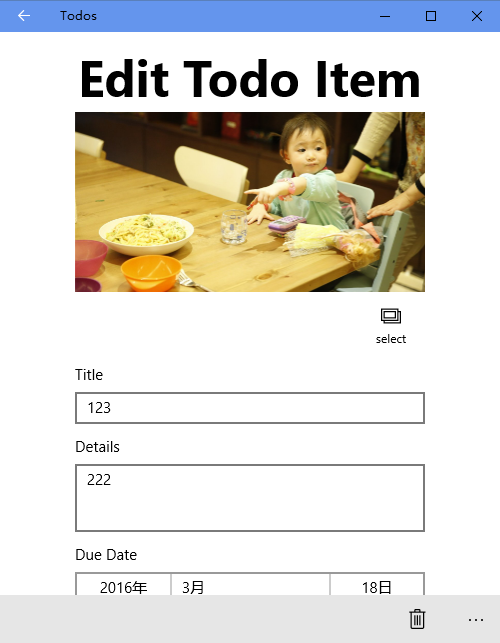


第一个任务的标题已更改为123

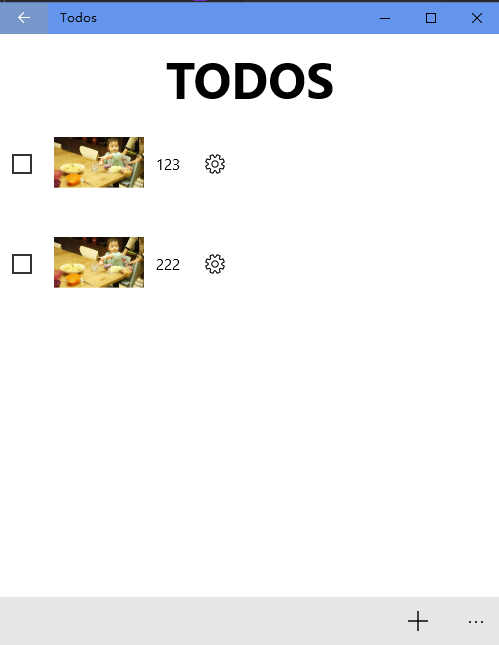
7. 点击第二个任务，修改标题为123，



8. 将程序挂起并关闭，并重新运行，

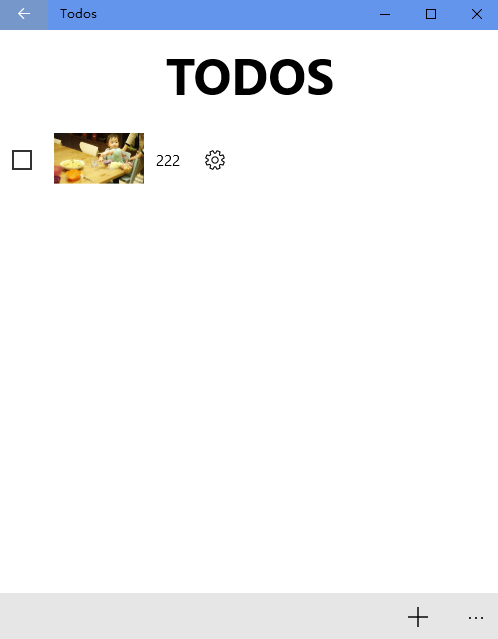


9. 点击Cancel按钮，



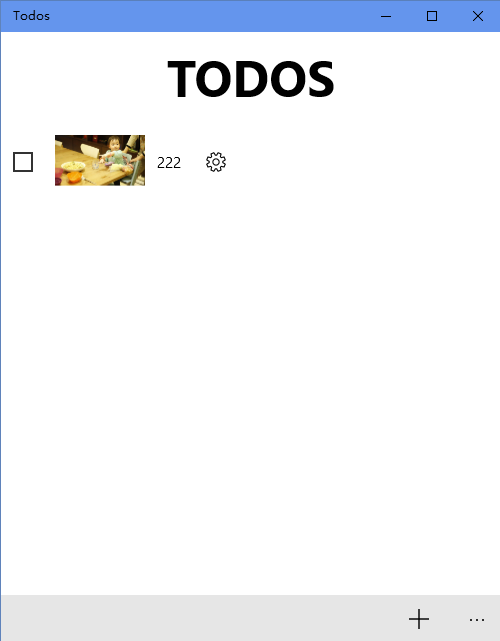
第二个任务并未改变

10. 点击第一个任务，点击删除按钮，



任务成功删除

11. 关闭程序并重新运行



数据保存正常

1. **实验过程遇到的问题**

1. 如果使用实例化的ViewModel，在每次页面跳转时需要传递ViewModel，

此时调用SetNavigationState函数会报错，错误原因是该函数要求页面跳转的时候只能传递基本类型。实验中，曾经采用将ViewModel序列化后再传递的方法，但是带来的新的问题是，将ViewModel反序列化时会调用TodoItemViewModel的构造函数，会导致TodoItem重复多个的情况。解决此问题的方法是：将TodoItemViewModel改为静态类，使用单例模式，这样就省去了传递ViewModel的步骤，自然就可以调用SetNavigationState函数了。

2. 由于反序列化会调用构造函数，之前的TodoItem的id将会跟上次运行时不同，导致不能使用GUID来定位TodoItem。此问题暂未解决，程序中采用临时的解决方案: 使用title来标识各TodoItem。

3. 在NewPage页面挂起时，数据应该保存在一个新的TodoItem中，这样在重新运行程序点击Cancel按钮才能将该TodoItem恢复回原来的状态。

**思考与总结**

反序列化会调用构造函数，可能会导致一些很难定位的问题，在调试程序时应仔细观察各变量的变化，必要时采用单步步入的方法观察程序的执行过程。

作业要求：

1. 命名要求: 学号\_姓名\_作业编号，例如14330000\_林XX\_hw1。
2. 实验报告提交格式为pdf。
3. 实验内容不允许抄袭，我们要进行代码相似度对比。如发现抄袭，按0分处理。