现代操作系统应用开发实验报告

学号： 14970011　　**班级：** 2015 级教务 2 班

姓名： 宋思亭　　**实验名称：** homework4

一．参考资料

Windows UWP Namespaces：https://docs.microsoft.com/en-us/uwp/api/

课件、Demo

二．实验步骤

1. 阅读作业需求和所给 demo

2. APP.xaml.cs 设计：

添加变量判定是否挂起：

```
public bool IsSuspending = false;
private void OnResuming(object sender, object e)
{
    IsSuspending = false;
}
```

OnLaunched：

```
protected override void OnLaunched(LaunchActivatedEventArgs e)
{
    Frame rootFrame = Window.Current.Content as Frame;
    if (rootFrame == null)
    {
        // 创建要充当导航上下文的框架，并导航到第一页
        rootFrame = new Frame();
        rootFrame.NavigationFailed += OnNavigationFailed;
        if (e.PreviousExecutionState == ApplicationExecutionState.Terminated)
        {
            //TODO: 从之前挂起的应用程序加载状态
```

```
                    if
(ApplicationData.Current.LocalSettings.Values.ContainsKey("NavigationState"))
                    {

rootFrame.SetNavigationState((string)ApplicationData.Current.LocalSettings.Values["Navi
gationState"]);
                    }
                }
                // 将框架放在当前窗口中
                Window.Current.Content = rootFrame;
            }

            if (rootFrame.Content == null)
            {
                // 当导航堆栈尚未还原时，导航到第一页，
                // 并通过将所需信息作为导航参数传入来配置
                // 参数
                rootFrame.Navigate(typeof(MainPage), e.Arguments);
            }
            // 确保当前窗口处于活动状态
            Window.Current.Activate();

            SystemNavigationManager.GetForCurrentView().BackRequested +=
OnBackRequested;

            rootFrame.Navigated += (s, a) => {
                if (rootFrame.CanGoBack)
                {
SystemNavigationManager.GetForCurrentView().AppViewBackButtonVisibility =
AppViewBackButtonVisibility.Visible;
                }
                else
                {

SystemNavigationManager.GetForCurrentView().AppViewBackButtonVisibility =
AppViewBackButtonVisibility.Collapsed;
                }
            };
        }
```

OnSuspending：

```
        private void OnSuspending(object sender, SuspendingEventArgs e)
        {
            //TODO: 保存应用程序状态并停止任何后台活动
```

```
            var deferral = e.SuspendingOperation.GetDeferral();
            IsSuspending = true;

            Frame frame = Window.Current.Content as Frame;
            ApplicationData.Current.LocalSettings.Values["NavigationState"] =
frame.GetNavigationState();
            deferral.Complete();
        }
```

3. 实现在MainPage和NewPage中，勾选项目，挂起并关闭程序，重新启动时，勾选的项目不变，使用本地储存，下面以MainPage为例，NewPage中的内容类似。

```
protected override void OnNavigatedFrom(NavigationEventArgs e)
{
    if (((App)Application.Current).IsSuspending)
    {
        ApplicationDataContainer Item =
ApplicationData.Current.LocalSettings.CreateContainer("Item",
ApplicationDataCreateDisposition.Always);
        if (ApplicationData.Current.LocalSettings.Containers.ContainsKey("Item"))
        {
            Item.Values["imgname"] = selectName;
            Item.Values["title"] = title.Text;
            Item.Values["details"] = details.Text;
            Item.Values["date"] = date.Date;
            Item.Values["btn"] = createButton.Content;
        }
        if (ViewModel.SelectedItem != null)
        {
            ApplicationData.Current.LocalSettings.Values["selectitem"] =
ViewModel.getItems.IndexOf(ViewModel.SelectedItem);
        }
        List<string> L = new List<string>();
        var allitems = ViewModel.getItems;
        foreach (var a in allitems)
        {
            var item = new myItem(a.date, a.imgname, a.title, a.details, a.finish);
            L.Add(JsonConvert.SerializeObject(item));
        }
        ApplicationData.Current.LocalSettings.Values["allitems"] =
JsonConvert.SerializeObject(L);
    }
```

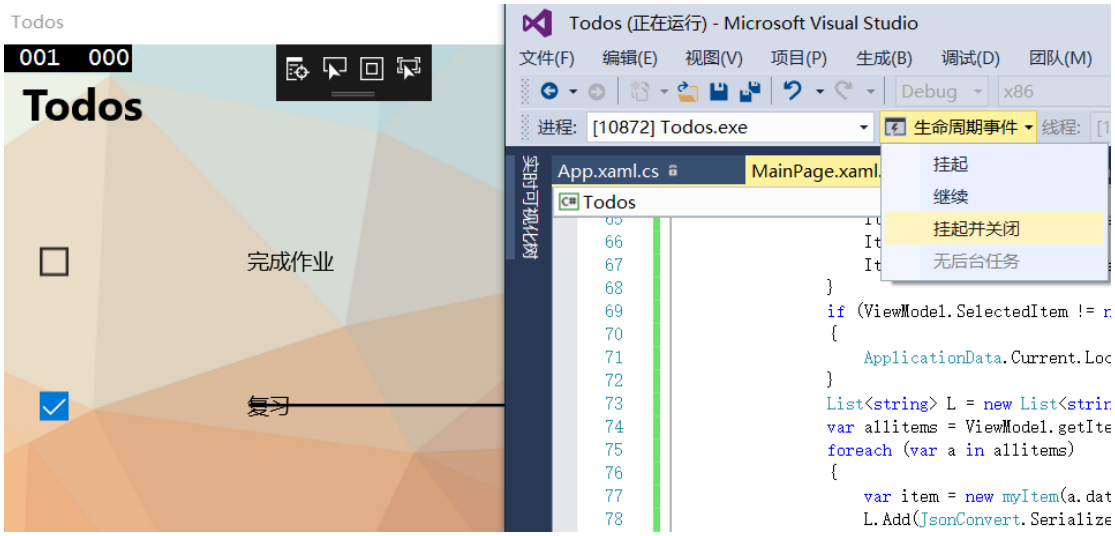
```
}

protected override async void OnNavigatedTo(NavigationEventArgs e)
{
    SystemNavigationManager.GetForCurrentView().AppViewBackButtonVisibility =
AppViewBackButtonVisibility.Collapsed;
    if (e.NavigationMode == NavigationMode.New)
    {
        ApplicationData.Current.LocalSettings.Values.Remove("Item");
        ApplicationData.Current.LocalSettings.Values.Remove("allitems");
        ApplicationData.Current.LocalSettings.Values.Remove("selectitem");
    }
    else
    {
        if (ApplicationData.Current.LocalSettings.Values.ContainsKey("allitems"))
        {
            ViewModel.getItems.Clear();
            List<string> L =
JsonConvert.DeserializeObject<List<string>>((string)ApplicationData.Current.LocalSettin
gs.Values["allitems"]);
            foreach (var l in L)
            {
                myItem a = JsonConvert.DeserializeObject<myItem>(l);
                TodoItem item = new TodoItem(a.date.Date, a.imgname, a.title, a.details,
a.finish);
                ViewModel.getItems.Add(item);
            }
        }
        if (ApplicationData.Current.LocalSettings.Values.ContainsKey("selectitem"))
        {
            ViewModel.SelectedItem =
ViewModel.getItems[(int)(ApplicationData.Current.LocalSettings.Values["selectitem"])];
        }

        if (ApplicationData.Current.LocalSettings.Containers.ContainsKey("Item"))
        {
            ApplicationDataContainer Item =
ApplicationData.Current.LocalSettings.Containers["Item"];
            createButton.Content = Item.Values["btn"] as string;
            title.Text = Item.Values["title"] as string;
            details.Text = Item.Values["details"] as string;
            date.Date = (DateTimeOffset)(Item.Values["date"]);
            selectName = Item.Values["imgname"] as string;
            if (selectName == "")
```

```
            {
                pic.Source = new BitmapImage(new Uri("ms-appx:///Assets/fruit.jpg"));
            }
            else
            {
                var file = await
ApplicationData.Current.LocalFolder.GetFileAsync(selectName);
                IRandomAccessStream fileStream = await
file.OpenAsync(FileAccessMode.Read);
                BitmapImage bitmapImage = new BitmapImage();
                await bitmapImage.SetSourceAsync(fileStream);
                pic.Source = bitmapImage;
            }
        }
    }
}
```

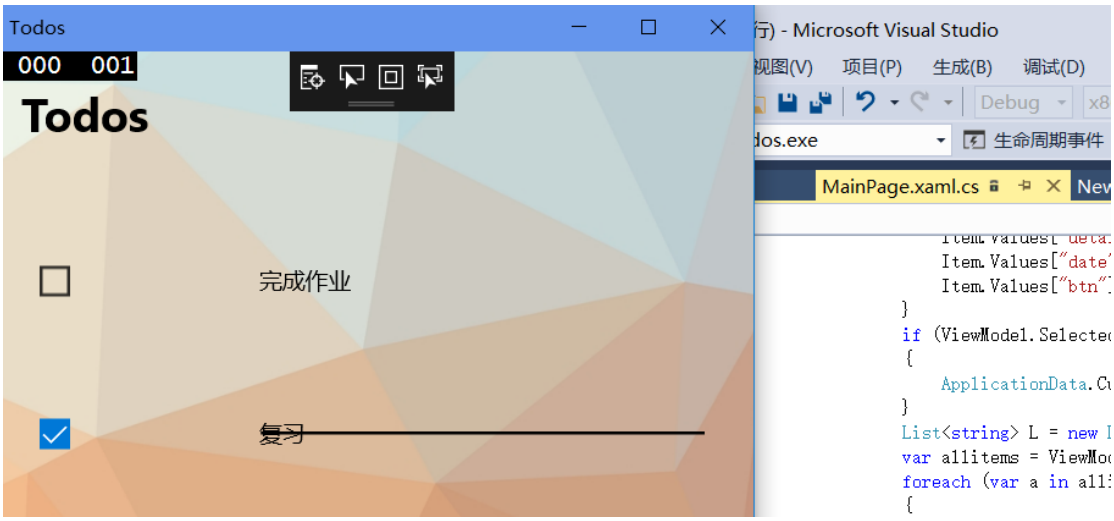
4. 调试项目

三．实验结果截图

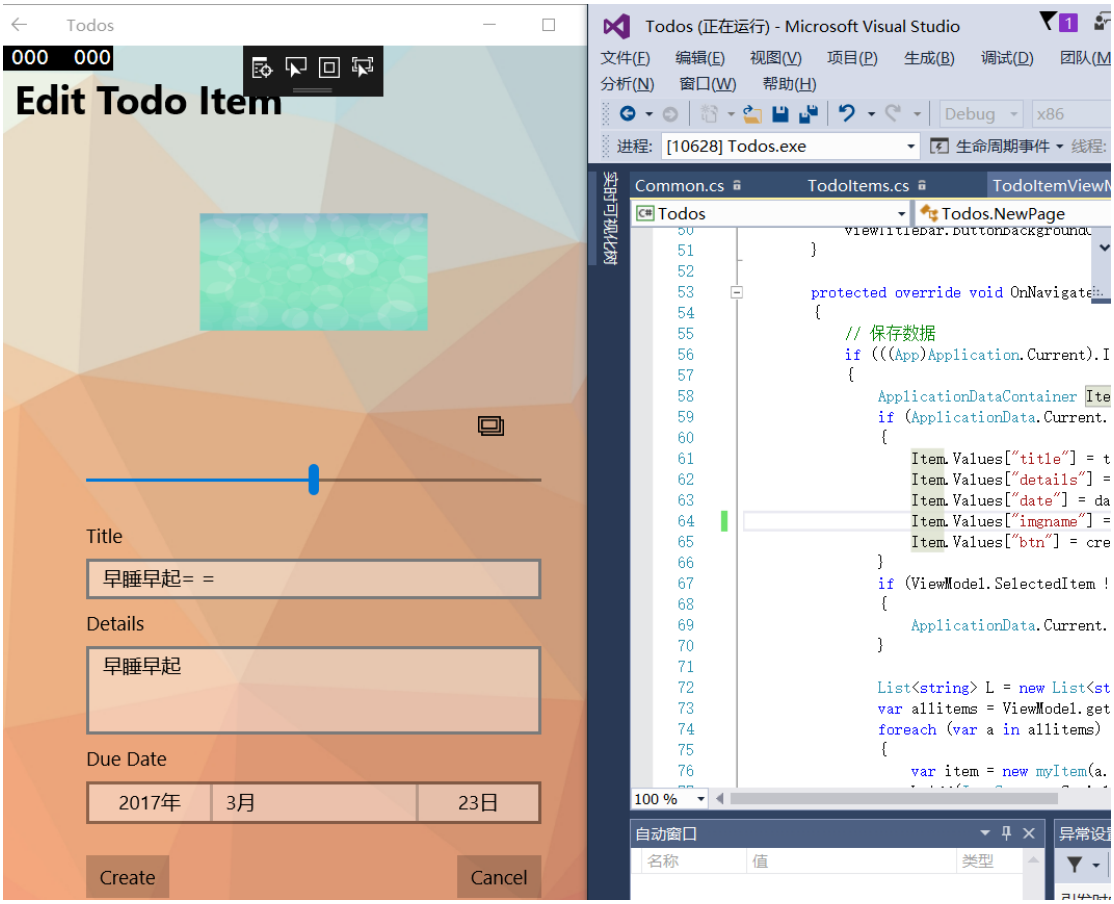
1. MainPage 中挂起并关闭项目



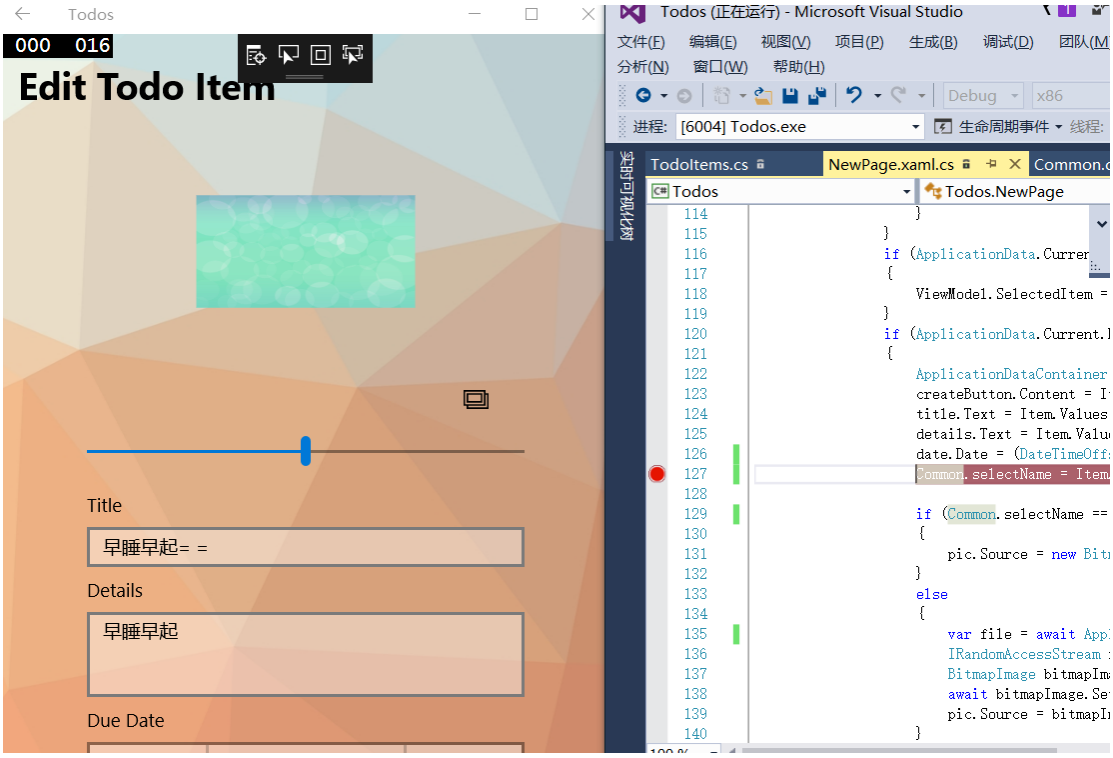
重新打开：



2. MainPage 中挂起并关闭项目



重新打开：



四．实验过程遇到的问题

1. ViewModel 作为参数在页面之间传递时，挂起动作会出错，写一个公用的 Common 类解决
2. BitmapImage 类型直接 JsonConvert.SerializeObject 转换类型存入会导致 DeserializeObject 逐条读取时报错，怀疑是{}符号导致，转换为 string 类型后存入解决
3. 页面逻辑还不完善
4. PS.请尽量在窄页面布局下测试

五．思考与总结

习得长期储存页面数据

图片类型转换好麻烦= =