程序设计实验项目报告

姓名： 彭康杰 学号： 20192131080

专业： 计算机科学与技术 班级： 2班

实验名称：Windows日历 实验日期：2020年8月至2020年10月

目录

一、前言 **2**

二、设计思路**3**

三、设计过程5

四、测试情况 11

五、优点与改进14

六、感想15

一、前言

随着计算机网络技术的高速发展和普及，信息化已成为人类社会发展的大趋势。对于计算机学生的综合素养要求也逐渐升高，对于大二本科生而言，已掌握部分编程语言知识是远远不够的，指由于用到实际工程中才是真正的学以致用。本实验我选择了UWP来制作Windows日历桌面应用程序，运用了xaml和C#语言，完成本次实验的设计。

二、设计思路

思维导图：

设计思路讲解

1.首先创建一个UWP工程项目，创建一个初始界面，运用Grid控件

2.在上述Grid中建立两行子Grid控件，分别用于存放标题TextBlock控件和日历Grid界面。

3.在日历Grid界面中设置两列Grid控件分别放置日期GridView控件和备忘录Page界面

4.整体布局完成后，通过后期事件处理以及相关代码完成日期GridView中的数据显示，同时通过事件响应控制备忘录界面

三、设计过程

1.建立UWP项目，使用“空白项目（通用Windows）C#”，开启一个新应用建立界面；



2.打开文件后分别有几个需要操作的文件：MainPage.xaml，MainPage.xaml.cs，App.xaml.cs，进行页面布局操作在MainPage.xaml中操作，同时MainPage.xaml.cs进行辅助MainPage.xaml控件内容的详细操作，App.xaml.cs则负责处理应用所有的工程项目执行：

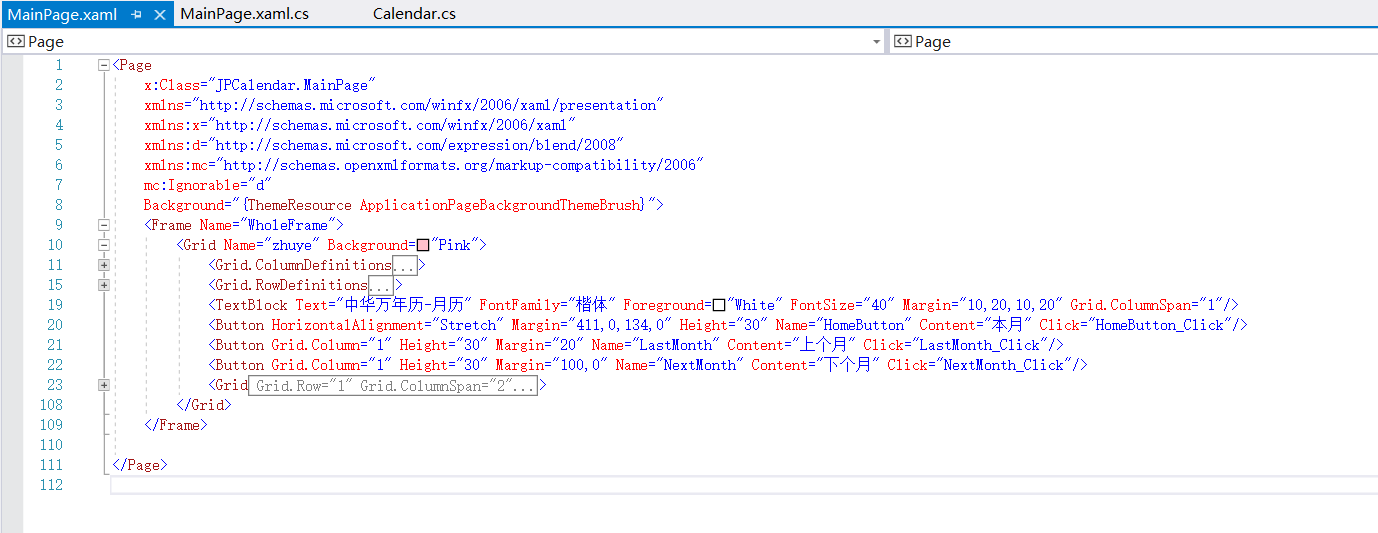
（1）设计一个总Grid并分成两行，第一行存放“中华万年历-月历”文字控件以及几个Button控件，第二行则用于存放整个操作核心的日期界面

（2）在日期界面中继续分成两列，左侧放置日期GridView控件，右侧放置备忘录Page界面

（3）根据页面视图调整每个模块之间的相对位置，同时进行背景以及其他高级功能的设置，美化页面设置。

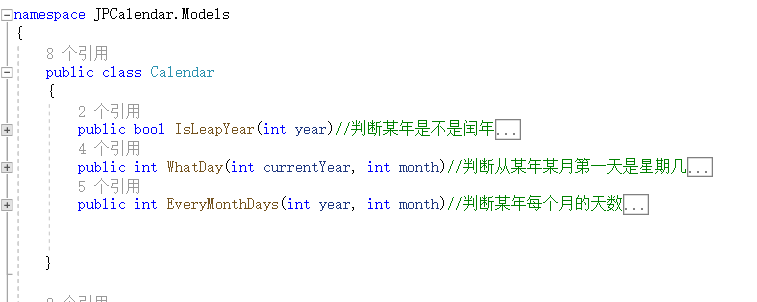


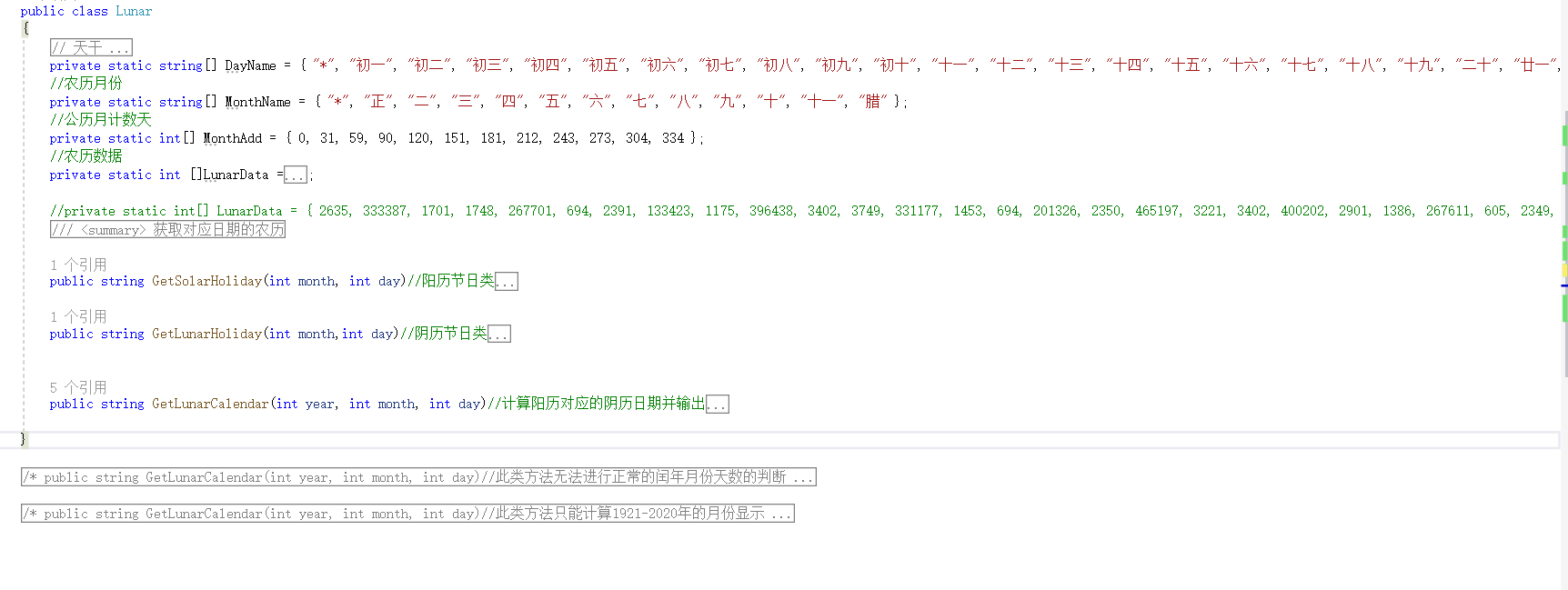
设计页面的部分代码（已隐藏部分内容）：



3.对日期GridView进行细节设计：对边框设计、鼠标悬停和点击效果以及一系列控件属性设计。同时添加一个calendar.cs文件用于存放日期类相关数据以及函数，在MainPage.xaml.cs文件中进行数据整理并绑定，从而在日历GridView控件内容中显示各个日期的具体数据。

其中涉及的类的作用：（其中两个农历类为经测试不完整故没采用此方法实现）





class Calendar:

public bool IsLeapYear(int year):判断某年是否为闰年

public int WhatDay(int currentYear,int month):判断某年某月第一天是星期几

public int EveryMonthDays(int year,int month):判断某年每个月的天数

class Lunar:

private static string[]DayName={}:存放农历日期day的名字

private static string[]MonthName={}:存放农历日期month的名字

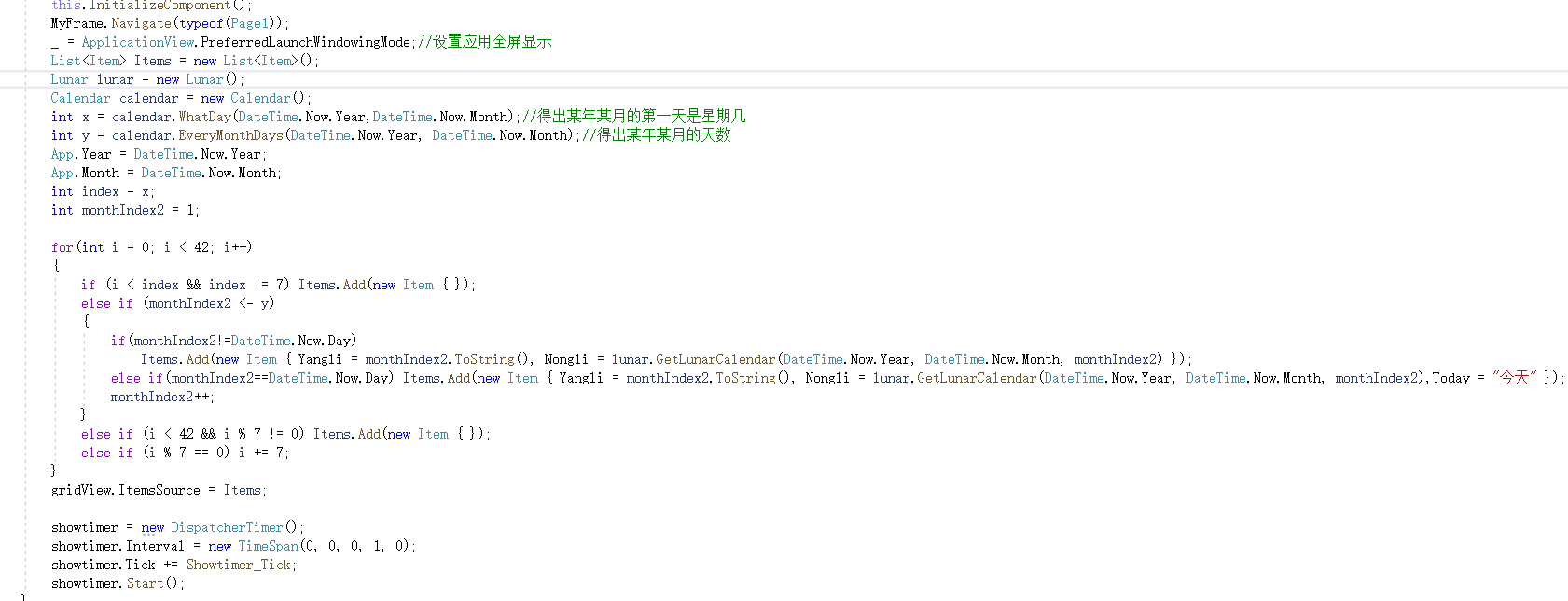
private static int[]MonthAdd={}:存放每个月的累计天数

private static int[]LunarData={}:存放每年农历的数据

public string GetSolarHoliday(int month,int day):存放阳历节日

public string GetLunarHoliday(int month,int day):存放阴历节日

public string GetLunarCalendar(int year,int month,int day):计算阳历对应的阴历日期并输出

日历GridView数据的实现：

4.将备忘录Page页面关联到31个空白页面Page.xaml，分别对应GridView中有内容的（<=31）GridView控件响应，即通过点击日历格子触发备忘录Page页面并跳转到相应界面。



上述代码为通过点击GridView控件跳转Page页面

5.对每一个备忘录Page.xaml进行页面设置，无日期和有日期采用不同的页面布局加以区分。

无日期类

有日期类

四、测试情况

1.初始界面



2.添加备忘录事件

3.查看上个月



4.查看下个月

5.回到本月



6.运行视频如压缩包内文件所示

五、优点与改进

优点：

1.相比于WPF、MFC、VC、Qt等设计平台，UWP具有的独特的设计界面与风格符合现代App的效果，不光是界面的美观程度还是整体的布局效果，都使得UWP设计出的应用别具特色；

2.这个日历App具有可视化的操作以及动态处理，对于不同事件的响应效果也有不同的设计，无论是从整体布局，还是对于其中的控件功能的合理运用都让这个App能够在功能上满足用户需求；

改进：

1.因为UWP基于Windows10上的应用设计学习为闭源，大部分资料网上几乎都查不到，所以在这个日历设计上有很多可以优化的地方，但查找不到资料只能放弃这个步骤；

2.对于日期模块的显示，因为UWP知识储备不够，在页面小屏显示时会出现排版错误，即找不到方法控制GridView的行列数，故只能在全屏显示时才能处在正确排版当中。

3.对于备忘录的功能，没有实现关闭程序后将备忘录中的事件存贮下来，以至于程序仅在执行中存储该月的备忘录，退出后则重新执行备忘录的设置。

六、感想

通过本次程序设计实验项目课程，我明白了一个App的设计不是想象中的那么简单，仅仅通过一个人设计是很难的，不光是页面视图的设计，还是其中类的功能实现，都需要日复一日的思考与研究，需要自己通过学习资料找到最适合自己设计的方法去设计应用。通过这次实验，我也掌握了基本的C#语言和xaml语言，知道如何通过这些语言去完成一个应用的设计，虽然有着C++语言学习的基础，但在新知识的冲撞中也会难免有不懂的地方，这个时候更需要自己抽时间去了解如何使用这个新功能，新语法，不断的尝试各种控件以及属性，才能设计出令人满意的效果。在完成这个日历App项目后，我发现通过UWP设计的项目和Swift设计的项目有一些共同之处，那就是美观的可视化界面，相比于WPF、MFC等设计平台，UWP所具有的可视化界面更符合现代App设计的审美，即UWP具有超前的设计理念，但由于其资料有限，我也明白了学习资料的重要性，我花了很久才找到了一本关于UWP设计的学习资料《深入浅出Windows10通用应用开发》，并且通过Bilibili上的UWP开发入门教程和一些现有代码教学，我才能将日历完成。这对于我日后的程序项目设计给予了启发和灵感，在下次做这种类似的项目时，就不会再像一个无头的苍蝇碰壁，而是有条有理的完成一个程序设计。