**1：小程序的flex布局：**

    我们的小程序是主要运行在微信，也就是手机上，手机呢也有不同的屏幕大小，而我们的flex布局，可以适应不同的屏幕分辨率。

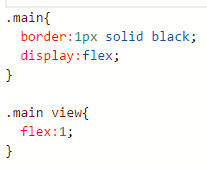
    小程序的视图层布局主要是弹性框模型，弹性框模型分为弹性容器以及弹性项目。

    我们设置最外层的容器为display:flex或者inline-flex时，该容器为弹性容器。

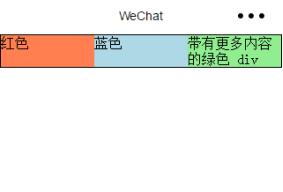
    弹性容器的子容器一般称为弹性项目。

**1.1 下面我们来感受一下，flex的魅力。**





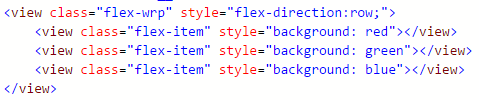
 效果图：



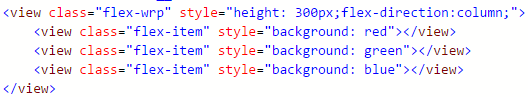
这个是最简单的flex布局样式写法。

**1.2 下面我们来尝试一些不同的写法：**

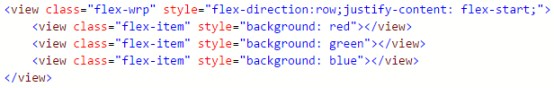
1.2.1 横向布局：（类似于浮动的效果）-- flex-direction:row



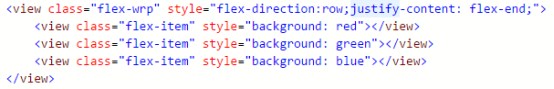
 1.2.2 纵向布局：flex-direction:column



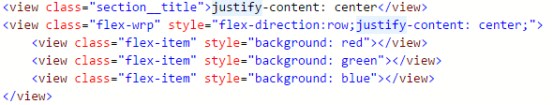
1.2.3 横向布局，从左侧布局：flex-direction:row; justify-content:flex-start



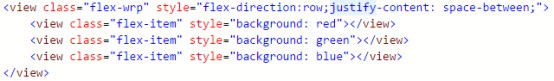
1.2.4 横向布局，从右侧布局：flex-direction:row; justify-content:flex-end



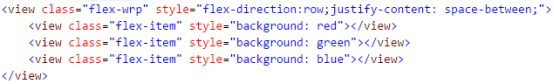
1.2.5 内容居中



 1.2.6  内容居中，内容之间有间距



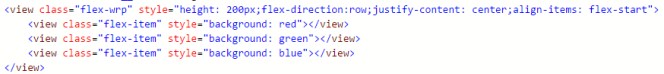
 1.2.7 内容四周有边距



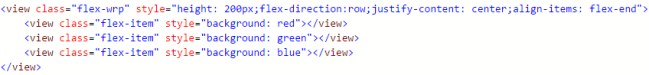
 1.2.8 垂直居中



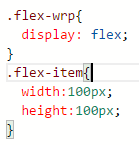
 1.2.9 左右居中，垂直居上



 1.2.0 左右居中，垂直居下



统一设置外层容器flex-wrap的样式：



 2**：小程序的视频项目：**

    首先呢，咱们的视频项目是分为俩种的，

**2.1一种是有提供链接可以在视频上面发送弹幕的**

**2.2还有一种是可以打开自己手机里的视频**

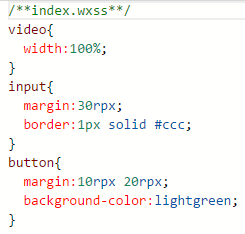
    步骤：

    清空index.wxml , index.wxss , index.js的内容

    2.1.1 先完成index.wxml的部分



    2.1.2 然后给这个项目写样式



    2.1.3 完成index.js部分

    首先我们需要理解，我们的小程序运行是有生命周期的，这里我们先了解一下什么是生命周期。

**2.1.3 知识点串讲-- 页面的生命周期**

****

**这里大家理解一下，页面的生命周期也就是页面开始加载到完成的一个执行顺序。**

**然后我们来完成index.js中的代码**

**首先强调一点：我们的页面相关的事件及交互处理都要放在Page({})函数中。**

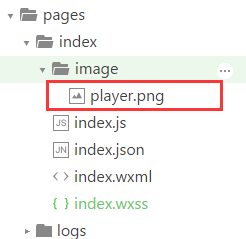
**然后我们现在是要给指定的视频发送弹幕，那么我们就需要一个视频对象**

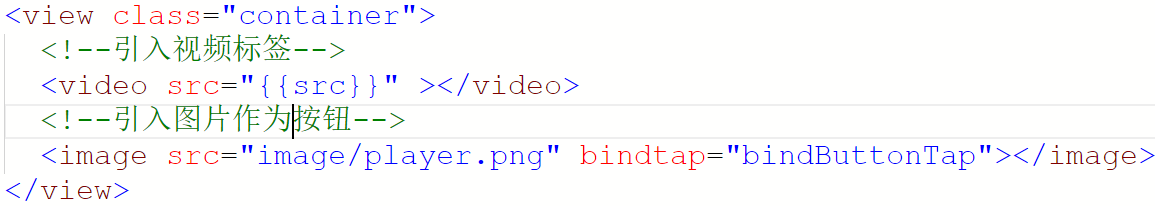
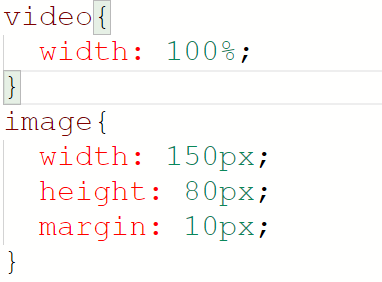
****

**C:\Users\yaq\Documents\My Knowledge\temp\cbf0e170-e92b-4d0e-96a2-0d11ffb983f4\128\index_files\a0272ce8-907d-4fe4-b47e-9cc239f15bee.png**

**2.2 可以打开自己手机里的视频**

1. **在index目录下创建 image文件夹，放入播放图片**



****

**3：小程序的地理定位：**

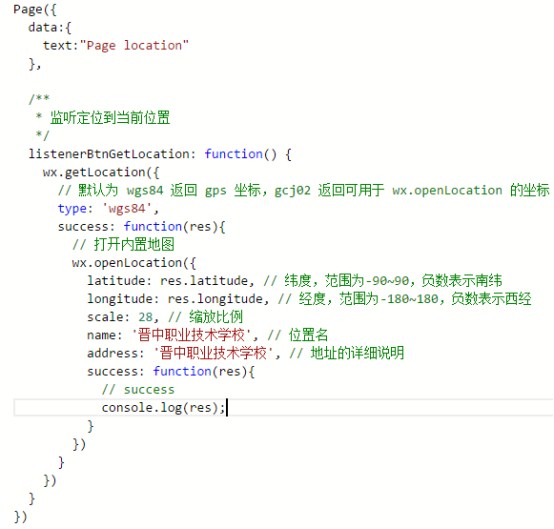
**咱们这里要完成的地理定位是使用微信的内置地图来定位，可能会有一些小误差**

**index.wxml部分**

**C:\Users\yaq\Documents\My Knowledge\temp\cbf0e170-e92b-4d0e-96a2-0d11ffb983f4\128\index_files\a6551779-f741-4b01-a937-acc9c310e5a5.jpg**

**注意：此处的type=”primary”可直接将button设置为绿色。**

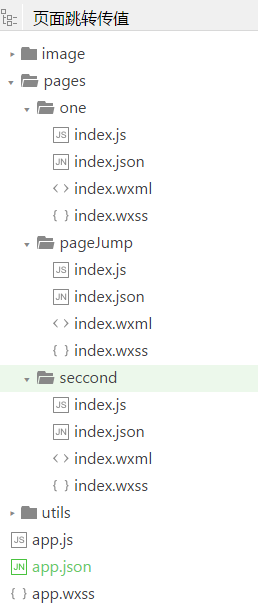
**index.js部分**

****

4**：页面跳转并传值：**

基础的页面跳转知识请参考微信公众平台的开发文档

4.1 创建项目目录



4.2 配置页面路径在app.json（设置在pages中，写在第一个的为进入程序即显示的页面）



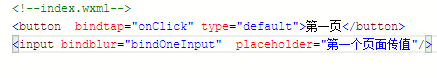
4.3 将第一个页面中输入框的值传入第二个页面的输入框

4.3.1 第一个页面pageJump

Wxml部分：

在页面中写上用于跳转的按钮，已经输入框input，将输入框中输入的值，传入到second页面的输入框中。

Bindblur:失去焦点时调用，方法名为bindOneInput.



Js部分：在输入框失去焦点时获取用户填入输入框的值

获取输入框的值使用e.detail.value

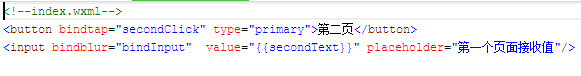
在点击按钮时，跳转页面并通过id传值

没有返回键的，使用wx.rediectTo, 并通过url进行页面跳转，?后跟上参数值（输入框的值）



4.3.2第二个页面（second）

wxml页面:



Js页面

主要是接收第一个页面pageJump中输入框里的值，通过形参options来接收参数id值

然后通过that.setData方法设置页面输入框的值，setData方法是将逻辑层的数据传输到视图层

注意：次数跳转的页面one页面，在底部导航栏（tabbar）中，因此跳转至one页面需使用switchTab

方法进行跳转

