1. **实训主要内容**
2. **小程序配置**
   1. 全局配置文件app.json

app.json是当前小程序项目中的全局配置文件，包括了小程序的所有页面的路径、界面表现、网络超时、底部tab等。

注意:在配置文件中不能出现注释语句

json对象:就是一种数据格式与js中的对象类型书写是一样的!

* 1. app.json对象中pages属性

pages属性是设置子页面路径的!并且在没有设置默认启动页面时,它会将pages数组中的第一项作为默认的启动页（在没有配置默认启动项时）。也可以在app.json中配置属性（entryPagePath:”路径”）来设置默认启动项。

* 1. window属性

用于设置小程序的状态栏,导航条、标题、窗口背景色等，在app.json中设置属于全局配置，也可以在pages下的每个页面单独配置当前网页的信息，局部配置和全局配置同时启用时，优先选择局部配置。

* 1. json对象tab属性

tabBar属性中,list属性接受一个数组,只能配置最少2个、最多5个tab。tab按数组的顺序排序,每个项都是一个对象,其属性值如下：pagePath、text、iconPath、selectedIconPath。

1. **数据绑定**
   1. 方式一：简单绑定 {{变量}}方式在view中使用变量
   2. 方式二：标签属性中使用data中的数据`{{}}`，view标签类似于html中的div。
2. **常见的wx指令**

指令：wx框架，提前写好的对应的代码！通过对应指令的形式去使用。指令的语法规则为： `wx:指令名称 if elif else, for...`

* 1. `wx:if-elif-else` 指令

`wx:if=表达式`，如果表达式为真，显示该组件（标签），结果为假，则隐藏。

`wx:if wx:else wx:elif` 主要用来渲染节点。

​`hidden属性（ture，false）`可以隐藏显示组件，避免页面重绘的问题：从DOM树中频繁插入和移除节点。区别：两者都具备显示和隐藏组件的功能，而wx:if比较消耗资源，如果组件切换比较频繁，不建议使用if，用hide来代替（html中的visibility:hidden和display:none的区别差不多：一个不占位，一个消失）

* 1. `wx:for` 指令：

在组件上使用 wx:for 控制属性绑定一个数组，即可使用数组中各项的数据重复渲染该组件。

**wx:for-item和wx:for-index：**

默认数组的当前项的下标变量名默认为 index ，数组当前项的变量名默认为 item ，在当前view中有用。

使用 wx:for-item 可以指定数组当前元素的变量名，使用 wx:for-index 可以指定数组当前下标的变量名。

<view wx:for="{{array}}" wx:for-index="idx" wx:for-item="itemName">

{{idx}}: {{itemName.message}}

</view>

**wx:key：**

如果列表中项目的位置会动态改变或者有新的项目添加到列表中，并且希望列表中的项目保持自己的特征和状态（如 input 中的输入内容，switch 的选中状态），需要使用 wx:key 来指定列表中项目的唯一的标识符。

wx:key 的值以两种形式提供：

1. 字符串，代表在 for 循环的 array 中 item 的某个 property（属性名），该 property 的值需要是列表中唯一的字符串或数字，且不能动态改变。（该属性的值必须在该对象中是唯一的）。

2. 保留关键字 \*this 代表在 for 循环中的 item 本身，这种表示需要 item 本身是一个唯一的字符串或者数字。

总结：\*this用于遍历的元素是数组，属性名用在对象中，用index时不一定是最好的选择。

当数据改变触发渲染层重新渲染的时候，会校正带有 key 的组件，框架会确保他们被重新排序，而不是重新创建，以确保使组件保持自身的状态，并且提高列表渲染时的效率。

如不提供 wx:key ，会报一个 warning ， 如果明确知道该列表是静态，或者不必关注其顺序，可以选择忽略。

1. **当日实训收获**

本次实训是对小程序的一个初步了解，对基础的全局配置、局部配置和整个程序的架构有了很好的了解，为后面学习打下基础，同时熟练的掌握了数据绑定和一些常用的wx指令（for-if）及其特性。