# 微信小程序 - Uni02

# 1.微信小程序组件

· <image>组件

<image **src="图像的URL地址"**></image>

webp格式是谷歌推荐的图像格式，其体积约是JPG文件的2/3。

https://img13.360buyimg.com/pop/s295x235\_jfs/t1/128362/27/9398/73431/5f3244a6E92963d6c/ed552f095347a16a.jpg.webp

https://img.zcool.cn/community/01c3885f35eefca80120a8212fc542.jpg@520w*390h*1c*1e*1o\_100sh.jpg

图像的动态缩放

·

<swiper>组件

**<swiper>**组件用于实现**轮播图**，其语法结构是：

<swiper   
 autoplay="是否自动播放"   
 indicator-dots="是否显示面板指示点"   
 indicator-color="指示点正常显示的颜色"   
 indicator-active-color="当前指示点的颜色"   
 circular="是否采用衔接滑动"   
 interval="动画切换的时间间隔(5000ms)"   
 duration="切换的时长(500ms)">   
 <swiper-item>...</swiper-item>   
 ...   
<swiper>

**swiper组件默认的高度为150px**

·

<text>组件

**<text>**组件用于**实现文本**，类似于HTML中的**span**元素，其语法结构是：

<text selectable="是否可选" decode="是否解码">...</text>

微信小程序的解码只支持：**&lt;**、**&gt;**、**&nbsp;**、**&amp;**和**&apos;**和**&emsp;**、**&ensp;**

HTML实体列表 https://www.w3school.com.cn/tags/html*ref*entities.html

·

<icon>组件

**<icon>**组件用于**实现图标**，其语法结构是：

<icon type="图标类型" size="图标尺寸" color="颜色"></icon>

**图标类型包括**：

* **success**表示成功
* **warn**表示警告
* **waiting**，表示等待
* **clear**,表示清除
* **info**,表示信息

**……….**

· <progress>组件

**<progress>**组件用于**实现进度条**，其语法结构是：

<progress   
 percent="百分比"   
 show-info="是否显示百分比信息"  
 stroke-width="进度条的高度"  
 activeColor="已完成部分的颜色"  
 backgroundColor="未完成部分的颜色">  
</progress>

· <button>组件

**<button>**用于**实现按钮**，其语法结构是：

<button   
 type="按钮的类型(primary|default|warn)"  
 size="按钮的尺寸(default|mini)"  
 plain="是否镂空"  
 loading="是否带有加载图标"  
 disabled="是否禁用"  
 open-type="开启的微信开放能力">  
</button>

open-type属性值包括：

**contact，打开客服会话**

**share，触发用户转发**

**getPhoneNumber**,**获取用户手机号**，该能力必须与按钮的**bindgetphonenumber事件**组合使用

**getUserInfo,获取用户信息**，该能力必须与按钮的**bindgetuserinfo事件**组合使用

· <navigator>组件

**<navigator>**组件用于**实现链接**，其语法结构是：

<navigator **url="当前小程序内的页面路径"** open-type="跳转方式">  
 ...  
</navigator>

open-type属性包括:

* **switchTab**，可以**跳转到包含在tabBar属性内的页面**
* redirect，重向定向到某个页面
* navigate，导航到某个页面
* reLaunch，小程序重新加载

· <map>组件

**<map>**组件用于**实现地图**，**该组件属于原生组件，**其语法结构是：

103.692059 30.62937  
<map **longitude="经度" latitude="纬度"**></map>

高德地图坐标拾取器：https://lbs.amap.com/console/show/picker

百度地图坐标拾取器：http://api.map.baidu.com/lbsapi/getpoint/index.html

· <input>组件

**<input>**组件用于**实现输入框**，**该组件为原生组件**，其语法结构是：

<input   
 type="输入框的类型"   
 placeholder="占位符内容"   
 maxlength="最大输入字符数"   
 focus="是否自动获得焦点"   
 password="是否为密码框"   
 bindinput="输入时触发的函数"   
 bindfocus="获得焦点时触发的函数"   
 bindblur="失去焦点时触发的函数"   
 bindconfirm="点击键盘的完成时触发的函数">   
</input>

type属性值包括：

* text,文本输入键盘
* number，数字输入键盘
* digit，带有小数点的数字输入键盘
* idcard，身份证号输入键盘

# 2.WXSS

**WXSS**(Weixin Style Sheets)，是一套**样式语言**， 用于描述WXML组件的样式。

**· rpx -- 尺寸单位**

rpx (Responsive Pixel)，**响应式像素**，其可**根据屏幕尺寸进行自适应**，规则屏幕尺寸为**750rpx**

|  |  |  |
| --- | --- | --- |
| 设备 | rpx换算px (屏幕宽度/750) | px换算rpx (750/屏幕宽度) |
| iPhone5 | 1rpx = 0.42px | 1px = 2.34rpx |
| iPhone6 | 1rpx = 0.5px | 1px = 2rpx |
| iPhone6 Plus | 1rpx = 0.552px | 1px = 1.81rpx |

# 

# 3.WXML

## 3.1 数据绑定

page.wxml中的**动态数据**来源于**page.js**中的**data**，示例代码如下：

data: {  
 className:'Web2004',  
 students:[  
 {  
 studentName:'朱泽宇',  
 age:18,  
 //true表示男,false表示女  
 sex:true  
 },  
 {  
 studentName:' 孙笑川',  
 age:19,  
 sex:false  
 },  
 {  
 studentName:'张西西',  
 age:16,  
 sex:true  
 },  
 {  
 studentName:'萌老闹',  
 age:81,  
 sex:true  
 }  
 ]  
}

### 3.2 内容绑定

<component>**{{variable}}**</component>

实例代码如下：

<view>

    <text class="heading">{{className}}班级学生名单如下：</text>

</view>

### 3.3 列表渲染

<component   
  **wx:for="{{variable}}" wx:key="index"**   
 **wx:for-item="成员变量名称" wx:for-index**="成员下标名称">  
</component>

**默认的成员变量名称为item**，默认成员**下标为index**

### 

### 3.4 条件渲染

<!--类似于JavaScript中的 if 结构-->  
<component wx:if="{{condition}}">...</component>

<!--类似于JavaScript中的 if ... else 结构-->  
<component wx:if="{{condition}}">  
 ...  
</component>  
<component wx:else>  
 ...  
</component>  
  
  
<!--类似于JavaScript中的 if ... else if ... else 结构-->  
<component **wx:if=**"{{condition}}">  
 ...  
</component>  
<component wx:elif="{{condition}}">  
 ...  
</component>  
<component wx:elif="{{condition}}">  
 ...  
</component>  
<component wx:else>  
 ...  
</component>

**<block>...</block>**组件用于**保证正常的流程控制语句的执行**，但又**不会现多余的、无用的节点**

实例代码如下：

<view>

    <text class="heading">{{className}}班级学生名单如下：</text>

    <view class="student" wx:for="{{students}}" wx:key="index">

        学号：{{index+1}};

        姓名：{{item.studentName}};

        年龄：{{item.age}};

        性别：{{item.sex ? "男":"女"}};

        学历：

        <block wx:if="{{item.education == 1}}">专科</block>

        <block wx:elif="{{item.education == 2}}">本科</block>

        <block wx:elif="{{item.education == 3}}">研究生</block>

        <block wx:elif="{{item.education == 4}}">硕士</block>

        <block wx:else>博士</block>

    </view>

</view>

1.尝试用循环结构实现轮播图

2.预习小程序中的事件及数据双向绑定

https://developers.weixin.qq.com/miniprogram/dev/framework/view/wxml/event.html