

# 常见内置组件

讲师: 王红元  
微博: coderwhy



更多学习资源



老师答疑交流



# Text组件

- Text组件用于显示文本, 类似于span标签, 是行内元素

属性	类型	默认值	必填	说明
selectable	boolean	false	否	文本是否可选
space	string		否	显示连续空格
decode	boolean	false	否	是否解码

space 的合法值

值	说明
ensp	中文字符空格一半大小
emsp	中文字符空格大小
nbsp	根据字体设置的空格大小

- selectable属性决定文本内容是否可以让用户选中

- space有三个取值(见右图)

- decode是否解码

□ decode可以解析的有 &nbsp; &lt; &gt; &amp; &apos; &ensp; &emsp;

```
home.wxml  ×
1  <!-- 1.selectable暂时无效 -->
2  <text selectable="{{true}}">Hello World</text>
3  </view>
4
5  <!-- 2.space属性 -->
6  <view>
7  |   <text space="ensp">Hello World</text>
8  | </view>
9  <view>
10 |   <text space="emsp">Hello World</text>
11 | </view>
12 <view>
13 |   <text space="nbsp">Hello World</text>
14 | </view>
15
16 <!-- 3.decode属性: 显示 3 > 2 -->
17 <text decode="{{true}}">3 &gt; 2</text>
```



# Button组件解析

- Button组件用于创建按钮，默认块级元素
- 常见属性：

属性	类型	默认值	必填	说明
size	string	default	否	按钮的大小
type	string	default	否	按钮的样式类型
plain	boolean	false	否	按钮是否镂空，背景色透明
disabled	boolean	false	否	是否禁用
loading	boolean	false	否	名称前是否带 loading 图标
form-type	string		否	用于 <code>&lt;form&gt;</code> 组件，点击分别会触发 <code>&lt;form&gt;</code> 组件的 submit/reset 事件
open-type	string		否	微信开放能力
hover-class	string	button-hover	否	指定按钮按下去的样式类。当 <code>hover-class="none"</code> 时，没有点击态效果

size 的合法值	
值	说明
default	默认大小
mini	小尺寸

type 的合法值	
值	说明
primary	绿色
default	白色
warn	红色





# Button组件演练

one 6 ▾ 85% ▾ WiFi ▾ 模拟操作 ▾ 🔊 🖨️ 🔄

10:16 100% 🔋

小程序

按钮

按钮 按钮 按钮

按钮 按钮 按钮

🌀按钮

按钮

assetscomponentspagesbuttonbutton.jsbutton.jsonbutton.wxmlbutton.wxsshomeinputscrollviewapp.jsapp.jsonapp.wxssproject.config.jsonsitemap.json

button.wxml

```
1 <!-- 1.默认Button -->
2 <button>按钮</button>
3
4 <!-- 2.Button的size属性: default/mini -->
5 <button size="mini">按钮</button>
6
7 <!-- 3.Button的type属性: default warn primary -->
8 <button size="mini" type="default">按钮</button>
9 <button size="mini" type="warn">按钮</button>
10 <button size="mini" type="primary">按钮</button>
11
12 <!-- 4.Button的plain属性: true, 没有背景 -->
13 <button size="mini" plain="{{true}}">按钮</button>
14
15 <!-- 5.Button的disable属性: true -->
16 <button size="mini" disabled="{{true}}">按钮</button>
17
18 <!-- 6.Button的loading, 显示小菊花 -->
19 <button loading="{{true}}">按钮</button>
20
21 <!-- 7.Button按下时的样式 -->
22 <button hover-class="button-hover">按钮</button>
```



# open-type的取值

- open-type用户获取一些特殊性的权限，可以绑定一些特殊的事件：

open-type 的合法值

值	说明
contact	打开客服会话，如果用户在会话中点击消息卡片后返回小程序，可以从 bindcontact 回调中获得具体信息， <a href="#">具体说明</a>
share	触发用户转发，使用前建议先阅读 <a href="#">使用指引</a>
getPhoneNumber	获取用户手机号，可以从bindgetphonenumber回调中获取到用户信息， <a href="#">具体说明</a>
getUserInfo	获取用户信息，可以从bindgetUserinfo回调中获取到用户信息

```
Page({
  onContact(e) {
    console.log(e)
  },
  onGetPhoneNumber(e) {
    console.log(e)
  },
  onGetUserInfo(e) {
    console.log(e)
  }
})
```

```
<!-- 8.open-type -->
<view>
  <button size='mini' open-type='contact' bindcontact='onContact'>客服会话</button>
  <button size='mini' open-type='share'>程序分享</button>
  <button size='mini' open-type='getPhoneNumber' bindgetphonenumber='onGetPhoneNumber'>获取电话</button>
  <button size='mini' open-type='getUserInfo' bindgetUserinfo='onGetUserInfo'>用户信息</button>
</view>
```



# View组件

## ■ 视图组件（块级元素，独占一行，通常用作容器组件）

属性	类型	默认值	必填	说明
hover-class	string	none	否	指定按下去的样式类。当 <code>hover-class="none"</code> 时，没有点击态效果
hover-stop-propagation	boolean	false	否	指定是否阻止本节点的祖先节点出现点击态
hover-start-time	number	50	否	按住后多久出现点击态，单位毫秒
hover-stay-time	number	400	否	手指松开后点击态保留时间，单位毫秒

## ■ 常用属性：

```
<!--home.wxml-->
<view hover-class='click'>Hello World</view>
```

```
/* home.wxss */
.click {
  background: ■ #f80;
  color: □ #fff;
}
```





# Image组件介绍

- Image组件用于显示图片，有如下常见属性：

属性	类型	默认值	必填	说明
src	string		否	图片资源地址
mode	string	scaleToFill	否	图片裁剪、缩放的模式
lazy-load	boolean	false	否	图片懒加载，在即将进入一定范围（上下三屏）时才开始加载
binderror	eventhandle		否	当错误发生时触发，， event.detail = {errMsg}
bindload	eventhandle		否	当图片载入完毕时触发， event.detail = {height, width}

- 其中src可以是本地图片，也可以是网络图片

- Mode属性使用也非常关键，详情查看官网：

□ <https://developers.weixin.qq.com/miniprogram/dev/component/image.html>

- 注意：image组件默认宽度320px、高度240px（官方文档有错误）



# Image组件演练

■ 这里补充一个API: wx.chooseImage (具体用法查看文档)

The screenshot displays a development environment for a WeChat Mini Program. On the left, a simulated mobile device shows a gallery with three images: a cat, a child, and another cat. A '选择图片' (Select Image) button is at the bottom. The middle pane shows the project file structure:

- images
- pages
- JS app.js
- app.json
- wxss app.wxss
- project.config.json
- sitemap.json

The right pane shows the WXML code for 'image.wxml':

```
1 <!--pages/image/image.wxml-->
2
3 <!-- 1.加载本地图片 -->
4 <view><image src="/images/cat.jpg" mode="widthFix"/></view>
5
6 <!-- 2.加载远程图片 -->
7 <view>
8   <image
9     src="https://tva2.sinaimg.cn/crop.0.0.750.750.180/006qtd2Tjw8f27vli33g
10     jpg" mode="widthFix"/>
11 </view>
12
13 <!-- 3.选择本地图片 -->
14 <view>
15   <button size='mini' bindtap='onChooseImage'>选择图片</button>
16   <image src="{{imageUrl}}"/>
17 </view>
```

The bottom status bar shows the file path '/pages/image/image.wxml', size '426 B', and a note '07\_内置组件补充\*'.





# Input组件的介绍

■ Input组件用于接收用户的输入信息，也是开发中非常常见的，有如下属性：

属性	类型	默认值	必填	说明
value	string		是	输入框的初始内容
type	string	text	否	input 的类型
password	boolean	false	否	是否是密码类型
placeholder	string		是	输入框为空时占位符

confirm-type	string	done	否	设置键盘右下角按钮的文字，仅在type='text'时生效
--------------	--------	------	---	-------------------------------

bindinput	eventhandle	是	键盘输入时触发，event.detail = {value, cursor, keyCode}, keyCode 为键值，2.1.0 起支持，处理函数可以直接 return 一个字符串，将替换输入框的内容。
bindfocus	eventhandle	是	输入框聚焦时触发，event.detail = { value, height }, height 为键盘高度，在基础库 1.9.90 起支持
bindblur	eventhandle	是	输入框失去焦点时触发，event.detail = {value: value}
bindconfirm	eventhandle	是	点击完成按钮时触发，event.detail = {value: value}

type 的合法值

值	说明
text	文本输入键盘
number	数字输入键盘
idcard	身份证输入键盘
digit	带小数点的数字键盘

confirm-type 的合法值

值	说明
send	右下角按钮为“发送”
search	右下角按钮为“搜索”
next	右下角按钮为“下一个”
go	右下角按钮为“前往”
done	右下角按钮为“完成”



# Input组件

Simulation interface showing the Input component in a mobile app environment.

**Left Panel (App Preview):** Displays a mobile app interface with a status bar (WeChat, 11:03, 100% battery) and a page titled "pages/input/input.wxml". The page contains five input fields:

- 请输入账号 (Please enter account number)
- 请输入密码 (Please enter password)
- (Empty field)
- (Empty field)
- 1ddf

**Right Panel (Code Editor):** Shows the source code for the Input component.

**app.json:**

```
1 <!--pages/input/input.wxml-->
2 <text>pages/input/input.wxml</text>
3
4 <!-- 1.input基本使用 -->
5 <input type='text' placeholder='请输入账号' placeholder-class='p-style'/>
6
7 <!-- 2.password -->
8 <input password='{{true}}' placeholder='请输入密码' placeholder-class='p-style'/>
9
10 <!-- 3.type: 类型在真机上有效(模拟器看不到键盘) -->
11 <input type="number"/>
12
13 <!-- 4.confirm-type: 在真机上有效(模拟器看不到键盘) -->
14 <input confirm-type='send'/>
15
16 <!-- 5.时间绑定和监听 -->
17 <input bindinput='onInput'
18      bindfocus='onFocus'
19      bindblur='onBlur'
20      bindconfirm='onConfirm'/>
```

**File Explorer:** Shows the project structure:

- images
- pages
  - home
  - image
  - input
    - input.js
    - input.json
    - input.wxml
    - input.wxss
- app.js
- app.json
- app.wxss
- project.config.json
- sitemap.json

**Bottom Panel (Console):** Shows the console output:

```
用户输入内容: {type: "input", timeStamp: 1557284547785, detail: {...}, target: {...}, currentTarget: {...}, ...}
用户输入内容: {type: "input", timeStamp: 1557284548144, detail: {...}, target: {...}, currentTarget: {...}, ...}
用户输入内容: {type: "input", timeStamp: 1557284548698, detail: {...}, target: {...}, currentTarget: {...}, ...}
失去焦点: {type: "blur", timeStamp: 31752, target: {...}, currentTarget: {...}, detail: {...}}
```



# scroll-view组件介绍

■ scroll-view可以实现局部滚动，常见属性如下：

属性	类型	默认值	必填	说明
scroll-x	boolean	false	否	允许横向滚动
scroll-y	boolean	false	否	允许纵向滚动

bindscrolltoupper	eventhandle	否	滚动到顶部/左边时触发
bindscrolltolower	eventhandle	否	滚动到底部/右边时触发
bindscroll	eventhandle	否	滚动时触发，event.detail = {scrollLeft, scrollTop, scrollHeight, scrollWidth, deltaX, deltaY}

■ 注意事项：

- ❑ 实现滚动效果必须添加scroll-x或者scroll-y属性（只需要添加即可，属性值相当于为true了）
- ❑ 垂直方向滚动必须设置scroll-view一个高度





# scroll-view代码演练

## ■ scroll-view

6 100% WiFi 模拟操作

pages/scrollview/scrollview.wxml

red

blue

green

red

blue

images

pages

home

image

input

scrollview

JS app.js

app.json

WXSS app.wxss

project.config.json

sitemap.json

scrollview.wxml

scrollview.js

```
1 <!--pages/scrollview/scrollview.wxml-->
2 <text>pages/scrollview/scrollview.wxml</text>
3
4 <!-- 1.水平滚动 -->
5 <scroll-view class='container-x' scroll-x>
6   <block wx:for="{{colors}}"
7     wx:key="*this">
8     <view style='background: {{item}};'
9       class='item'>
10       {{item}}
11     </view>
12   </block>
13 </scroll-view>
14
15 <!-- 2.垂直滚动 -->
16 <scroll-view class='container-y'
17   scroll-y
18   bindscrolltoupper="upper"
19   bindscrolltolower="lower"
20   bindscroll="scroll">
21   <block wx:for="{{colors}}"
22     wx:key="*this">
23     <view style='background: {{item}};'
24       class="item-y">
25       {{item}}
26     </view>
27   </block>
28 </scroll-view>
```

```
upper() {
  console.log('滚动到顶部')
},
lower() {
  console.log('滚动到底部')
},
scroll(e) {
  console.log(e)
}
```



# 共同属性

■ 所有wxml标签(组件)都支持的属性称之为共同属性，有如下共同属性

属性名	类型	描述	注解
id	String	组件的唯一标识	整个页面唯一
class	String	组件的样式类	在对应的 WXSS 中定义的样式类
style	String	组件的内联样式	可以动态设置的内联样式
hidden	Boolean	组件是否显示	所有组件默认显示
data-*	Any	自定义属性	组件上触发的事件时，会发送给事件处理函数
bind*/catch*	EventHandler	组件的事件	