第九章 小程序表单与医疗急救卡（上）

医疗急救卡小程序是什么？

该小程序主要为一个表单的填写与提交。示例表单的内容包括姓名、性别、血型、医疗情况、服药情况、是否为器官捐献者和紧急联系人电话号码。提交成功后表单显示用户已填写的信息，如果用户想更新，直接更改表单内容再次提交即可。

我们所用到的组件和 API

所需组件

- 表单：form，将组件内用户输入的 <input/> <switch/> <radio> 等提交；
- 按钮：button，用以提交表单等；
- 输入框：input，用户输入姓名和紧急联系人电话号码；
- 单项选择器：radio-group 和 radio， 用户选择性别；
- 从底部弹起的滚动选择器：picker， 用户选择血型；
- 多项选择器：checkbox-group 和 checkbox，用户选择医疗情况；
- 多行输入框：textarea，用户输入自身服药情况；
- 开关选择器：switch，用户选择是否为器官捐献者。

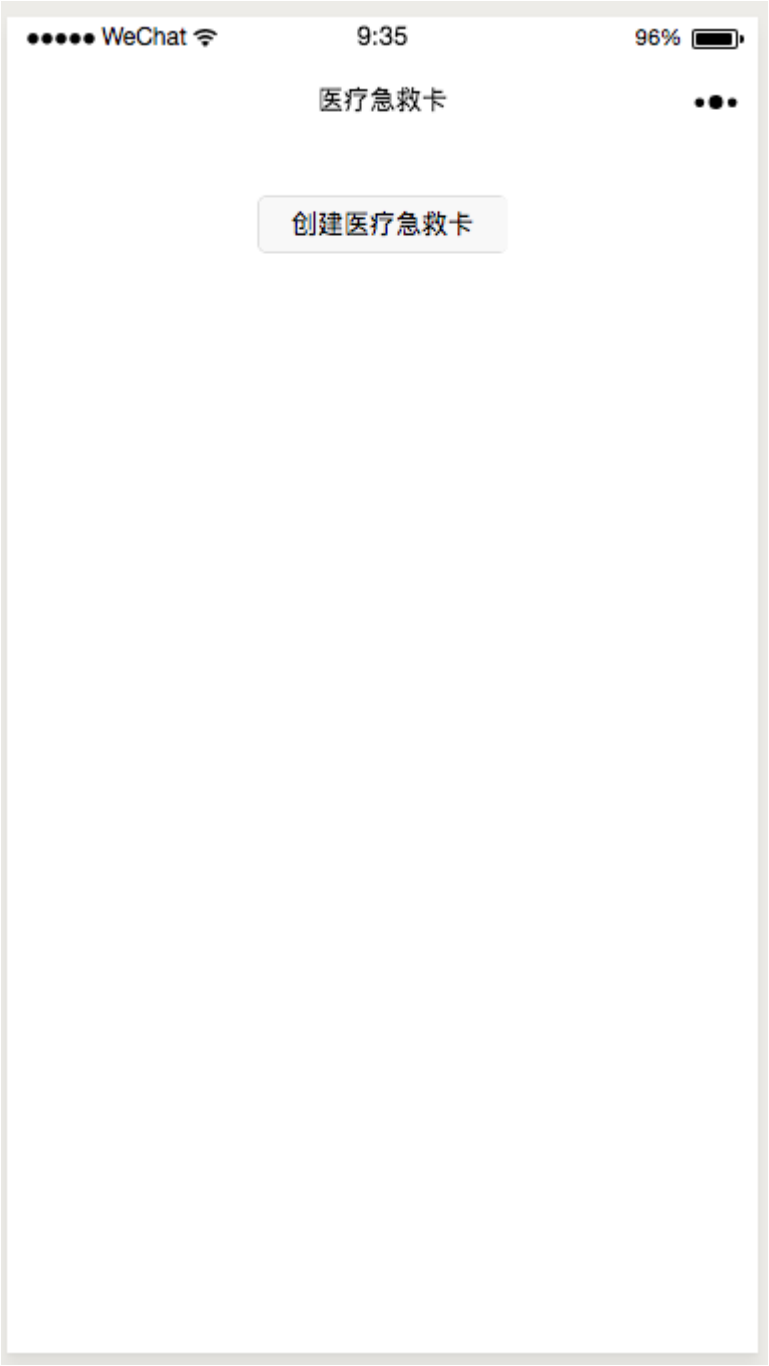
所需 API

- wx.showToast: 显示消息提示框，当用户提交表单后，给予成功或失败提示；
- wx.makePhoneCall: 拨打紧急联系人电话。

上手开发

卡片展示页面

首次使用小程序，页面将显示一个 「创建医疗急救卡」 的按钮，效果图如下：



点击该按钮后显示表单，效果图如下：

当用户提交过表单后，按钮“提交”变为“更新”，再次加载小程序直接显示表单，此时表单默认信息为上次用户所提交的信息，效果图如下：

首页「创建医疗急救卡」按钮

首次使用小程序，默认显示按钮，隐藏表单（点击按钮后，隐藏按钮，显示表单）

<!-- ./pages/index.index.wxml -->

```
<view wx:if="{{!isShowMedicalCard}}">
 <button size="mini" bindtap="showMedicalCard">创建医疗急救卡</button>
</view>

// pages/index/index.js
Page({
 data: {
  isShowMedicalCard: false,
 },

 showMedicalCard: function () {
  this.setData({
   isShowMedicalCard: true,
  })
 },
})
```

该 button 按钮中，我们绑定了一个点击事件 showMedicalCard，通过改变 isShowMedicalCard 的值，来控制表单的显示（值为 true）与隐藏（值为 false）。

医疗急救卡表单

form 表单组件

添加表单 form 组件，用以在表单内添加输入框 <input/> 等组件。

```
<!-- pages/medical-card/medical-card.wxml -->
<view wx:else>
 <form bindsubmit="formSubmit">
 </form>
</view>

// pages/index/index.js
pages({
 formSubmit: function (e) {
  let data = e.detail.value
 },
```

})

表单 form 组件中，有 bindsubmit 的属性，它携带表单中各组件的值（需要在相应的组件中加上 name 来作为 key）触发 formSubmit 事件。

该 form 组件绑定的点击事件是 formSubmit，当点击 <form/> 表单中的 <button formType="submit"></button> 组件时，将会触发该事件。

input 输入框组件

接下来，我们就要添加输入框 input 组件，用以让用户在小程序中输入姓名。

```
<!-- pages/index/index.wxml -->
<view wx:else>
 <form bindsubmit="formSubmit">
  <view>
   <view>姓名：</view>
   <input name="name" placeholder="请输入姓名" placeholder-class />
  </view>
 </form>
</view>
```

input 的组件属性，如下所示：

- type: input 的类型，例如文字、密码、邮箱等，以便小程序针对性进行适应。
- placeholder: 输入框为空时的占位符。
- placeholder-class: 置顶 placeholder 的样式类。
- bindinput: 当键盘输入时，触发 input 事件。

radio 单选框组件

接下来，我们要添加单选 radio 组件，用以选择性别。

```
<!-- pages/index/index.wxml -->
<view>
```

```
 <view>性别：</view>
 <radio-group name="gender">
  <label><radio value="男">男</radio></label>
  <label><radio value="女">女</radio></label>
 </radio-group>
</view>
```

在单选容器 radio-group 组件中，内部会由多个 radio 组成。

每个 radio-group 组件，可能会有 bindchange 属性，它代表 radio-group 中的选中项发生变化时，触发 change 事件。

每个 radio 组件都可能有以下属性：

- value ，当该 radio 选中时，radio-group 的 change 事件会携带 radio 的 value。
- checked: 当前是否选中，默认为 false。

picker 滚动选择器组件

接下来，我们添加 picker 组件（从底部滑起的滚动选择器），用以填写血型。

```
<!-- pages/index/index.wxml -->
<view>
 <view>血型：</view>
 <picker name="bloodType" bindchange="bloodTypeChange"
range="{{bloodTypes}}">
  <view>当前选择：{{bloodTypes[index]}}</view>
 </picker>
</view>

// pages/index/index.js
page({
 data: {
  bloodTypes: ['A', 'B', 'AB', 'O', 'RH+', 'RH-', 'Hh/孟买血型', '亚孟买血型', 'P 血型
'],
```

```
  index: 0,
 },
 bloodTypeChange: function (e) {
  let value = e.detail.value
  this.setData({
   index: value,
  })
 },
})
```

在每个 picker 组件（普通选择器）中，都可能有这些属性：

● mode：从底部弹起的滚动选择器，现支持五种类型，通过这个属性来改变，默认是普通选择器（mode = selector）。示例使用普通选择器。

● range: mode 为 selector 或 multiSelector 时， range 有效。

● value: 表示选择了 range 中的第几项（下标从 0 开始）。

● bindChange: value 改变时触发对应的 change 事件。

在实例中，该 picker 组件绑定的事件是 bloodTypeChange，滚动选项列表将会改变 value 值，从而触发事件。

checkbox 多选框组件

添加多项选择器 checkbox 组件，用以选择医疗情况。

```
<!-- pages/index/index.wxml -->
<view>
 <view>选择医疗情况：</view>
 <checkbox-group name="medicalConditions">
  <label>
   <checkbox wx:for="{{medicalConditions}}" wx:key="{{index}}"
value="{{item.name}}" checked="{{item.checked}}">{{item.name}}</checkbox>
  </label>
 </checkbox-group>
</view>
```

```
// pages/index/index.js
pages({
 data: {
  medicalConditions: [
   { name: '有过敏史', checked: false },
   { name: '有过大型手术', checked: false },
   { name: '有家族遗传病', checked: false }
  ],
 },
})
```

每个多项选择器，内部由多个 checkbox 组成，并使用 checkbox-group 进行组织。

每个 chheckbox-group 都可能有 bindChange 属性，它的意义是当 checkbox-group 中选中项发生改变时触发，将会执行特定函数。

在每个 checkbox 组件中，有可能有以下属性：

- value: 选中一项时触发 <checkbox-group /> 中的 bindChange 事件，并携带 checkbox 的 value 值。
- disabled: 是否禁用该 checkbox。
- checked: 设置默认选中项。
- color: checkbox 的颜色。

textarea 多行文本框组件

添加多行输入框 textarea 组件，用以填写服药情况。

```
<!-- pages/index/index.wxml -->
<view>
 <view>填写服药情况：</view>
 <textarea name="medicationCompliance" placeholder="请填写服药情况" auto-height="true"></textarea>
```

</view>

在 textarea 多行输入框属性中，有可能有这些属性：

- bindinput: 当键盘输入时，触发 bindinput 事件。
- auto-height: 是否自动增高，默认为 false。

switch 开关组件

添加开关选择器 switch 组件，用以选择是否为器官捐献者。

```
<!-- pages/index/index.wxml -->
<view>
 <view>是否为器官捐献者：</view>
 <switch name="isOrganDonor"></switch>
</view>
```

每个开关选择器 switch 组件，都可能有这些属性：

- checked: 是否选中，默认为 false。
- bindChange： checked 改变时触发 bindChange 事件。