

第十章 小程序表单与医疗急救卡（下）

在上一节，我们了解了许多表单组件的用法。在这一节，我们将会继续对小程序进行开发。

特殊功能开发

小程序请求呼出电话

添加输入框 `input` 组件和按钮 `button` 组件，用以输入紧急联系人电话号码和点击 `button` 拨打号码

```
<!-- pages/index/index.wxml -->
<view>
  <view>紧急联系人电话号码: </view>
  <input name="phoneNumber" type="number"
bindinput="phoneNumberChange" placeholder="请输入紧急联系人电话号码"
placeholder-class />
  <button size="mini" bindtap="makePhoneCall">拨打紧急联系人号码</button>
</view>
// pages/index/index.js
pages({
  data: {
    phoneNumber: "",
  },
  phoneNumberChange: function(e) {
    let phone = e.detail.value
    this.setData({
      phoneNumber: phone
    })
  },
  makePhoneCall: function() {
    let {phoneNumber} = this.data
    wx.makePhoneCall({
      phoneNumber: phoneNumber,
    })
  }
})
```

```
},  
})
```

在这里，我们用到了新的 API: `wx.makePhoneCall(OBJECT)`。先来了解一下它的参数。

- **phoneNumber**: 需要拨打的电话号码，是必填的。
- **success**: 接口调用成功的回调函数。
- **fail**: 接口调用失败的回调函数。
- **complete**: 接口调用结束的回调函数（调用成功、失败都会执行）。

动态调整按钮行为

如果用户没有提交过表单，那么 **button** 显示为「提交」，点击 **button** 提交时，后台会在数据表中新建一行数据，这行数据会有一个唯一的表示 `_id`；

当用户提交过表单后，再次提交表单不应该新建数据行，而是应该更新已有数据行的数据（该行数据的 `_id` 不会改变），同时 **button** 显示为「更新」；

```
<!-- pages/index/index.wxml -->  
<view>  
  <button size="mini" formType="submit" hidden="{{isShowUpdate}}">提交  
</button>  
  <button size="mini" formType="submit" hidden="{{!isShowUpdate}}">更新  
</button>  
</view>
```

```
// pages/index/index.js  
pages({  
  data: {  
    isShowUpdate: false,  
  },  
})
```

提交表单分为两种：初次提交和修改表单后提交。

我们可以在 `data` 中新增一项属性 `form: null`，在用户提交表单成功后，将 `form` 的值设置为表单内各组件的值。这样，我们可以通过 `form` 是否等于 `null` 来判断用户是否是初次提交。

```
// pages/index/index.js
pages({
  data: {
    form: null,
  },

  formSubmit: function(e) {
    let val = e.detail.value
    let tableID = 4066
    let Product = new wx.BaaS.TableObject(tableID)

    // 判断如果不是初次提交，则更新数据（更新服务器上已有的数据）
    if (this.data.form !== null) {
      // example 为访问自定义服务器接口函数
      example(val).then(res => {
        // 数据更新成功后，提示用户“更新成功”
        wx.showToast({
          title: '更新成功'
        })
      }, err => {
        // 数据更新失败后，提示用户“更新失败”
        wx.showToast({
          title: '更新失败'
        })
      })
      return
    }

    // 用户未选择血型，则默认为 A 型血
    if (!val.bloodType) {
      val.bloodType = this.data.index.toString()
    }
  }
})
```

```

// 初次提交，将数据上传到服务器
example(val).then(res => {
  wx.showToast({
    title: '提交成功'
  })

  // 将 data 中的 form 值设置为返回的数据，并且将按钮“提交”改为“更新”
  this.setData({
    form: { ...res.data },
    isShowUpdate: true,
  })
}, err => {
  wx.showToast({
    title: '提交失败'
  })
})
},
})

```

显示用户创建的医疗急救卡

我们想要达到目的是：用户已提交成功表单，那么，小程序启动后直接显示表单，且表单内组件默认值为已提交表单的数据。

这时，我们就需要在 **onLoad** 事件中获取数据表中的表单数据，并且将表单内组件的默认值设置为获取到的相应数据。

以下是详细的 **WXML** 代码，每个组件增加默认 **value** 属性。

```

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

```

```
<view>姓名: </view>
  <input name="name" value="{{form.name}}" placeholder="请输入姓名"
placeholder-class />
</view>
<view></view>
<view>
  <view>性别: </view>
  <radio-group name="gender">
    <label><radio value="男" checked="{{form.gender === '男'}}">男
</radio></label>
    <label><radio value="女" checked="{{form.gender === '女'}}">女
</radio></label>
  </radio-group>
</view>
<view></view>
<view>
  <view>血型: </view>
  <picker name="bloodType" value="{{form.bloodType}}"
bindchange="bloodTypeChange" range="{{bloodTypes}}">
    <view>
      当前选择: {{bloodTypes[index]}}
    </view>
  </picker>
</view>
<view></view>
<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>
<view></view>
<view>
  <view>填写服药情况: </view>
```

```

    <textarea name="medicationCompliance"
value="{{form.medicationCompliance}}" placeholder="填写服药情况"
auto-height="true"></textarea>
  </view>
</view></view>
<view>
  <view>是否为器官捐献者: </view>
  <switch name="isOrganDonor" checked="{{form.isOrganDonor}}"></switch>
</view>
<view></view>
<view>
  <view>紧急联系人电话号码: </view>
  <input name="phoneNumber" type="number"
value="{{form.phoneNumber}}" bindinput="phoneNumberChange"
placeholder="请输入紧急联系人电话号码" placeholder-class />
  <button size="mini" bindtap="makePhoneCall">拨打紧急联系人号码
</button>
</view>
<view></view>
<view>
  <button size="mini" formType="submit" hidden="{{isShowUpdate}}">提交
</button>
  <button size="mini" formType="submit" hidden="{{!isShowUpdate}}">更新
</button>
</view>
</form>
</view>

```

```

// pages/index/index.js

```

```

pages({
  onLoad: function() {
    let medical = this.data.medicalConditions
    // 获取自定义服务器上的数据
    example().then(res => {
      // 示例，所有数据存储在 res.data.objects 中
      let obj = res.data.objects
    })
  }
})

```

```
// 判断数据表有数据时，设置表单内组件的默认 value
if (obj) {
  let medical = this.data.medicalConditions
  for (let i = 0; i < medical.length; i++) {
    if (obj.medicalConditions.includes(medical[i].name)) {
      medical[i].checked = true
      this.setData({
        medicalConditions: medical,
      })
    }
  }
  this.setData({
    index: obj.bloodType,
    isShowMedicalCard: true,
    form: { ...obj },
    isShowUpdate: true,
  })
}
}, err => {
  wx.showToast({
    title: '获取数据出错'
  })
})
},
})
```

