

微信小程序开发

获取系统和设备的基本信息

系统和设备信息

- 这一节把API基础部分的获取系统信息和API设备部分的一些简单常用的接口放在一起说明。
- 主要是一些获取信息的场景，获取后根据信息状态做进一步处理，比如根据网络状态决定是不是要进行上传文件的操作。

获取系统信息

- 在小程序开发者文档的API菜单中，基础部分有wx.getSystemInfo接口用于获取系统详细信息。
- 这个API提供了同步（getSystemInfoSync）和异步（getSystemInfo）两个操作。

获取系统信息

```
wx.getSystemInfo({
  success: (res) => {
    this.setData({
      brand:    res.brand,
      model:    res.model,
      system:   res.system,
      platform: res.platform,
      version:  res.version
    });
  },
});
```

```
<view class="container">
  <view class="info-line">品牌: {{brand}}</view>
  <view class="info-line">型号: {{model}}</view>
  <view class="info-line">微信版本: {{version}}</view>
  <view class="info-line">系统: {{system}}</view>
  <view class="info-line">平台: {{platform}}</view>
</view>

.info-line {
  line-height: 40rpx;
  padding: 5rpx;
}
```



电量

- 获取电量的接口非常简单，通过wx.getBatteryInfo获取。

```
wx.getBatteryInfo({
  success: (res) => {
    this.setData({
      level: res.level,
      charging: res.isCharging ? '正在充电' : '未充电'
    });
  }
});
```

```
<view class="container">
  <text>电量: {{level}}%, {{charging}}</text>
</view>
```

网络状态

- 通过wx.getNetworkType可以获取当前网络状态。
- wx.onNetworkStatusChange用于监听网络状态的变化。
- 一些依赖于网络服务的场景可能需要用到这两个接口。
- 尤其是一些比较耗费流量的场景，如果用户没有连接wifi，则需要提醒用户是否继续。

网络状态接口示例

```
wx.getNetworkType({
  success: (res) => {
    this.setData({
      connStat: res.networkType == 'none' ? '未连接' : '已连接',
      netType: res.networkType
    });
  },
});
```

```
wx.onNetworkStatusChange((res) => {
  this.setData({
    connStat: res.isConnected ? '已连接' : '未连接',
    netType: res.networkType
  });
});
```

网络状态wxml模板

```
<view class="container">  
|   <text>网络状态: {{connStat}}, 类型: {{netType}}</text>  
</view>
```

- 切换手机网络状态可以看到页面数据的改变。