**小程序生命周期**

// app.js

App({

  onLaunch (options) {

    // Do something initial when launch.

  },

  onShow (options) {

    // Do something when show.

  },

  onHide () {

    // Do something when hide.

  },

  onError (msg) {

    console.log(msg)

  },

  globalData: 'I am global data'

})

**页面生命周期和事件及自定义函数**

//index.js

Page({

  data: {

    text: "This is page data."

  },

  onLoad: function(options) {

    // 页面创建时执行

  },

  onShow: function() {

    // 页面出现在前台时执行

  },

  onReady: function() {

    // 页面首次渲染完毕时执行

  },

  onHide: function() {

    // 页面从前台变为后台时执行

  },

  onUnload: function() {

    // 页面销毁时执行

  },

  onPullDownRefresh: function() {

    // 触发下拉刷新时执行

  },

  onReachBottom: function() {

    // 页面触底时执行

  },

  onShareAppMessage: function () {

    // 页面被用户分享时执行

  },

  onPageScroll: function() {

    // 页面滚动时执行

  },

  onResize: function() {

    // 页面尺寸变化时执行

  },

  onTabItemTap(item) {

    // tab 点击时执行

    console.log(item.index)

    console.log(item.pagePath)

    console.log(item.text)

  },

  // 事件响应函数

  viewTap: function() {

    this.setData({

      text: 'Set some data for updating view.'

    }, function() {

      // this is setData callback

    })

  },

  // 自由数据

  customData: {

    hi: 'MINA'

  }

})

**页面路由**

如下路由Api以及页面生命周期的变化

| **路由方式** | **触发时机** | **路由前页面** | **路由后页面** |
| --- | --- | --- | --- |
| 初始化 | 小程序打开的第一个页面 |  | onLoad, onShow |
| 打开新页面 | 调用 API [wx.navigateTo](https://developers.weixin.qq.com/miniprogram/dev/api/route/wx.navigateTo.html) 使用组件 [<navigator open-type="navigateTo"/>](https://developers.weixin.qq.com/miniprogram/dev/component/navigator.html) | onHide | onLoad, onShow |
| 页面重定向 | 调用 API [wx.redirectTo](https://developers.weixin.qq.com/miniprogram/dev/api/route/wx.redirectTo.html) 使用组件 [<navigator open-type="redirectTo"/>](https://developers.weixin.qq.com/miniprogram/dev/component/navigator.html) | onUnload | onLoad, onShow |
| 页面返回 | 调用 API [wx.navigateBack](https://developers.weixin.qq.com/miniprogram/dev/api/route/wx.navigateBack.html) 使用组件[<navigator open-type="navigateBack">](https://developers.weixin.qq.com/miniprogram/dev/component/navigator.html) 用户按左上角返回按钮 | onUnload | onShow |
| Tab 切换 | 调用 API [wx.switchTab](https://developers.weixin.qq.com/miniprogram/dev/api/route/wx.switchTab.html) 使用组件 [<navigator open-type="switchTab"/>](https://developers.weixin.qq.com/miniprogram/dev/component/navigator.html) 用户切换 Tab |  | 各种情况请参考下表 |
| 重启动 | 调用 API [wx.reLaunch](https://developers.weixin.qq.com/miniprogram/dev/api/route/wx.reLaunch.html) 使用组件 [<navigator open-type="reLaunch"/>](https://developers.weixin.qq.com/miniprogram/dev/component/navigator.html) | onUnload | onLoad, onShow |

**全局数据**在 app.js 定义全局数据

// app.js

App({

  globalData: 1

})

在其他页面使用

// test.js

// 获取app实例

var app = getApp()

// 改变数据

app.globalData++

**微信Api调用**

**事件监听 API**

以 on 开头的 API 用来监听某个事件是否触发

示例：

wx.onCompassChange(function (res) {

  console.log(res.direction)

})

**同步 API**

以 Sync 结尾的 API 都是同步 API

try {

  wx.setStorageSync('key', 'value')

} catch (e) {

  console.error(e)

}

**异步 API**

大多数 API 都是异步 API，这类 API 接口通常都接受一个 Object 类型的参数

Object 参数说明：

| **参数名** | **类型** | **必填** | **说明** |
| --- | --- | --- | --- |
| success | function | 否 | 接口调用成功的回调函数 |
| fail | function | 否 | 接口调用失败的回调函数 |
| complete | function | 否 | 接口调用结束的回调函数（调用成功、失败都会执行） |
| 其他 | Any | - | 接口定义的其他参数 |

回调函数的参数：

success，fail，complete 函数调用时会传入一个 Object 类型参数，包含以下字段

| **属性** | **类型** | **说明** |
| --- | --- | --- |
| errMsg | string | 错误信息，如果调用成功返回 ${apiName}:ok |
| errCode | number | 错误码，仅部分 API 支持，具体含义请参考对应 API 文档，成功时为 0。 |
| 其他 | Any | 接口返回的其他数据 |

示例：

wx.login({

  success(res) {

    console.log(res.code)

  }

})

**异步 API 返回 Promise**

基础库 2.10.2 版本起，异步 API 支持 callback & promise 两种调用方式

示例：

// promise 形式调用

wx.chooseImage().then(res => console.log('res: ', res))

**网络请求**

与fetch相似，详细文档地址：

<https://developers.weixin.qq.com/miniprogram/dev/api/network/request/wx.request.html>

示例：

wx.request({

  url: 'test.php', //仅为示例，并非真实的接口地址

  data: {

    x: '',

    y: ''

  },

  method: 'POST',

  header: {

    'content-type': 'application/json' // 默认值

  },

  success (res) {

    console.log(res.data)

  }

})