

1. 创建小程序项目

```
# 全局安装 vue-cli
$ npm install --global vue-cli
# 创建一个基于 mpvue-quickstart 模板的新项目
$ vue init mpvue/mpvue-quickstart my-project
# 安装依赖
$ cd my-project
$ npm install
# 启动构建
$ npm run dev
```

2. 生命周期钩子函数

除了 Vue 本身的生命周期外，mpvue 还兼容了小程序生命周期，不同的是我们会在小程序 onReady 后，再去触发 vue mounted 生命周期

app 部分：

- onLaunch, 初始化
- onShow, 当小程序启动，或从后台进入前台显示
- onHide, 当小程序从前台进入后台

page 部分：

- onLoad, 监听页面加载
- onShow, 监听页面显示
- onReady, 监听页面初次渲染完成
- onHide, 监听页面隐藏
- onUnload, 监听页面卸载
- onPullDownRefresh, 监听用户下拉动作
- onReachBottom, 页面上拉触底事件的处理函数
- onShareAppMessage, 用户点击右上角分享
- onPageScroll, 页面滚动
- onTabItemTap, 当前是 tab 页时，点击 tab 时触发（mpvue 0.0.16 支持）

注意点：

1. 不要在选项属性或回调上使用箭头函数

2. 微信小程序的页面的 query 参数是通过 onLoad 获取的，mpvue 对此进行了优化，直接通过 this.\$root.\$mp.query 获取相应的参数数据，其调用需要在 onLoad 生命周期触发之后使用，比如 onShow 等

模板语法

- 1.不支持纯html，也就是说v-html不能使用
- 2.不支持部分复杂的 JavaScript 渲染表达式，但是事件里面的表达式都是支持的
- 3.不支持过滤器，渲染部分会转成 wxml，wxml 不支持过滤器，所以这部分功能不支持

计算属性

不支持在 template 内使用 methods 中的函数

Class 与 Style

Class 与 Style 的表达式通过 compiler 硬编码到 wxml，暂不支持在组件上使用 Class 与 Style 绑定

列表渲染

嵌套列表渲染，必须指定不同的索引

踩坑注意：

- 列表中没有的原生事件也可以使用例如 bindregionchange 事件直接在 dom 上将bind改为@ @regionchange ,同时这个事件也非常特殊，它的 event type 有 begin 和 end 两个，导致我们无法在 handleProxy 中区分到底是什么事件，所以你在监听此类事件的时候同时监听事件名和事件类型既 `<map @regionchange="functionName" @end="functionName" @begin="functionName"></map>`
- 小程序能力所致，bind 和 catch 事件同时绑定时候，只会触发 bind ,catch 不会被触发，要避免踩坑。
- 事件修饰符
 - .stop 的使用会阻止冒泡，但是同时绑定了一个非冒泡事件，会导致该元素上的 catchEventName 失效！
 - .prevent 可以直接干掉，因为小程序里没有什么默认事件，比如submit并不会跳转页面
 - .capture 支持 1.0.9
 - .self 没有可以判断的标识
 - .once 也不能做，因为小程序没有 removeEventListener, 虽然可以直接在 handleProxy 中处理，但非常的不优雅，违背了原意，暂不考虑
- 其他 键值修饰符 等在小程序中压根没键盘，所以。。。

表单

几乎全支持

[select 组件用 picker 组件进行代替](#)

[表元素 radio 用 radio-group 组件进行代替](#)

组件

有且只能使用单文件组件（.vue 组件）的形式进行支持

详细的不支持列表：

- 暂不支持在组件引用时，在组件上定义 click 等原生事件、v-show（可用 v-if 代替）和 class style 等样式属性(例：`<card class="class-name"> </card>` 样式是不会生效的)，因为编译到 wxml，小程序不会生成节点，建议写在内部顶级元素上。
- Slot（scoped 暂时还没做支持）
- 动态组件
- 异步组件
- inline-template
- X-Templates
- keep-alive
- transition
- class
- style

常见问题

1. 如何获取小程序在 page onLoad 时候传递的 options

在所有 页面 的组件内可以通过 this.\$root.\$mp.query 进行获取。

2. 如何获取小程序在 app onLaunch/onShow 时候传递的 options

在所有的组件内可以通过 this.\$root.\$mp.appOptions 进行获取。

3. 如何捕获 app 的 onError

由于 onError 并不是完整意义的生命周期，所以只提供一个捕获错误的方法，在 app 的根组件上添加名为 onError 的回调函数即可