



CSDN学院 IT实战派

Vue全家桶实战仿美团外卖App

第二章

@董秀玲

| 介绍

■ Vue是什么

■ 安装

■ `{{msg}}` 文本插值

| Vue常用指令

- 原始HTML `v-html`
- 动态绑定属性 `v-bind`
- 条件渲染 `-v-if` `v-else` `v-else-if` `v-show`
- 事件绑定 `v-on`
 - 事件修饰符 `.stop` `.prevent`
- 列表渲染 `v-for`
- 表单输入绑定 `v-model`

| 组件

组件是可复用的Vue实例，在我们开发过程中我们可以把重复的工程封装成组件，达到快捷 便捷开发的目的。

- 组件是可复用的 Vue 实例

- 全局注册
- 局部注册

- data必须是一个函数

- 组件通信

- 通过prop向子组件传参
- 子组件使用 \$emit向父组件传参

生命周期

每个 Vue 实例在被创建时都要经过一系列的初始化过程，需要设置数据监听、编译模板、将实例挂载到 DOM 并在数据变化时更新 DOM 等。同时在这个过程中也会运行一些叫做**生命周期**钩子的函数

beforeCreate	组件实例刚被创建，组件属性计算之前，如 data 属性等
created	组件实例创建完成，属性已绑定，但 DOM 还未生成， \$el 属性还不存在
beforeMount	模板编译/挂载之前
mounted	模板编译/挂载之后
mounted	模板编译/挂载之后（不保证组件已在 document 中）
beforeUpdate	组件更新之前
updated	组件更新之后
beforeDestory	组件销毁前调用
destoryed	组件销毁后调用

| 计算属性computed

■ computed

- 计算属性在处理一些复杂逻辑时是很有用的。计算属性的结果会被缓存，除非依赖的响应式 property 变化才会重新计算。

■ computed vs methods

- 我们可以使用 methods 来替代 computed，效果上两个都是一样的，但是 computed 是基于它的依赖缓存，只有相关依赖发生改变时才会重新取值。而使用 methods，在重新渲染的时候，函数总会重新调用执行。

| 监听属性watch

■ watch

-通过监听属性响应数据的变化

```
vm.$watch('count', function (newVal, oldVal) {  
  console.log('计数值的变化 :' + newVal + ' 变为 ' + oldVal + '!');  
});
```

| 过渡 & 动画

Vue在插入、更新或者移除DOM时，提供多种不同的过渡效果，在进入/离开的过渡中，会有 6 个 class 切换。

■ enter

- **v-enter**：定义进入过渡的开始状态。在元素被插入之前生效，在元素被插入之后的下一帧移除。
- **v-enter-active**：定义进入过渡生效时的状态。在整个进入过渡的阶段中应用，在元素被插入之前生效，在过渡/动画完成之后移除。这个类可以被用来定义进入过渡的过程时间，延迟和曲线函数。
- **v-enter-to**：**2.1.8 版及以上**定义进入过渡的结束状态。在元素被插入之后下一帧生效（与此同时 v-enter 被移除），在过渡/动画完成之后移除。

| 过渡 & 动画

■ leave

- **v-leave** : 定义离开过渡的开始状态。在离开过渡被触发时立刻生效，下一帧被移除。
- **v-leave-active** : 定义离开过渡生效时的状态。在整个离开过渡的阶段中应用，在离开过渡被触发时立刻生效，在过渡/动画完成之后移除。这个类可以被用来定义离开过渡的过程时间，延迟和曲线函数。
- **v-leave-to** : **2.1.8 版及以上**定义离开过渡的结束状态。在离开过渡被触发之后下一帧生效 (与此同时 v-leave 被删除)，在过渡/动画完成之后移除。

JavaScript钩子

- before-enter
- enter
- after-enter
- before-leave
- leave
- after-leave

❗ 当只用 JavaScript 过渡的时候，在 `enter` 和 `leave` 中必须使用 `done` 进行回调。否则，它们将被同步调用，过渡会立即完成。