



## 课程简介



# 初识Vue

- 框架作者 尤雨溪
- 官网 <https://cn.vuejs.org/>
- github上面 6万多的star数量
- 组件化的、快速开发的特点
- 最流行前端开发框架之一

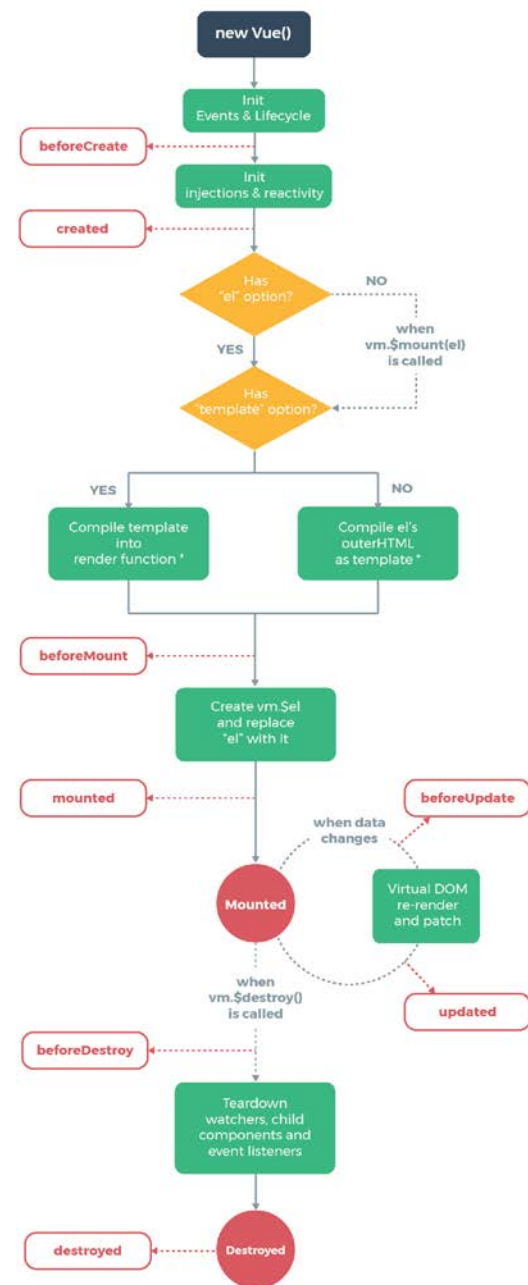


# 生命周期



# 生命周期

- **beforeCreate** 组件刚刚被创建
- **created** 组件创建完成
- **beforeMount** 挂载之前
- **mounted** 挂载之后
- **beforeDestroy** 组件销毁前调用
- **destroyed** 组件销毁后调用



# 安装

**# 全局安装 vue-cli**

**npm install --global vue-cli**

**# 创建一个基于webpack模板的新项目**

**vue init webpack my-project**

**# 安装依赖包**

**cd my-project**

**npm install**

**npm run dev**



**选项数据**



# 选项数据

- **data**
- **computed**
- **methods**



## 模版语法





# 模版语法

- data
- 模板中直接嵌入js代码
- 指令 v-html、v-on、v-bind等
- 计算属性
- 过滤器

计算属性

# 计算属性

- 在模板中放入太多的逻辑会让模板过重且难以维护
- 计算属性下所有函数可以放到computed中

# Class 与 Style 绑定

## v-bind指令

- 动态地绑定一个或多个特性
- 缩写 ‘:’
- 原始写法 v-bind:class 缩写写法 :class

# Class绑定

- 绑定1 : { active: isActive, 'text-danger': hasError }
- 绑定2 : classObject
- 绑定3 : [activeClass, errorClass]

# Style绑定

- 绑定1 : { color: activeColor, fontSize: '16px' }
- 绑定2 : styleObject
- 绑定3 : [baseStyles, overridingStyles]



**条件渲染**





# 条件渲染

- **v-if 指令**
- **v-if v-else**
- **v-if v-else-if v-else**

# 列表渲染



# 列表渲染

- **v-for指令**
- **用法1 : v-for="item in items" 数组**
- **用法2 : v-for="(item, index) in items" 数组**
- **用法3 : v-for="(value, key) in object" 对象**

# 事件处理器

# 事件处理器

- **v-on指令**
- **缩写 @**
- **原始写法 v-on:click 缩写 @click**
- **事件修饰符**

# 表单控件绑定

# 表单控件绑定

- **v-model 指令在表单控件元素上创建双向数据绑定**
- **复选框**
- **多个勾选框**
- **选择列表**

# 自定义组件



# 自定义组件

- 组件写到components文件夹下
- 自定义一个倒计时组件
- 组件基本要素：props、\$emit等
- 通过import导入自定义组件



实例



# 实例

- **Vue中操作DOM**
- **this.\$nextTick**
- **this.\$refs**



过渡效果



# 过渡效果

- 过渡transition
- 用法1：通过样式方式写过渡
- 用法2：通过vue自带属性方式写过渡

# 路由vue-router

# vue-router

- npm install引入vue-router包
- 页面跳转功能
- 用法1 : `<router-link to="/demo1">demo1</router-link>`
- 用法2 : `to="{ name: 'demo9', params: { userId: 123 } }"`
- 用法3 : `this.$router.push({ path: '/demo1' })`
- 用法4 : `this.$router.push({ name: 'demo1', params: { userId: 123 } })`

# 状态管理vuex



# 状态管理vuex

- **npm install引入vuex包**
- **全局状态管理，所有页面共享数据**
- **设置数据：this.\$store.dispatch('increment', 100000);**
- **获取数据：this.\$store.state.num;**



**Slot插槽**



# Slot插槽

- 常用语组件调用中

**vue-resource请求**

# vue-resource请求

- **npm install引入vue-resource包**
- **this.\$http.get('/someUrl')**
- **this.\$http.post('/someUrl', {foo: 'bar'})**

# 移动组件库Mint UI

# 移动组件库Mint UI

- 官网：<http://mint-ui.github.io/docs/#/zh-cn>
- 带着大家跑一个小实例