# 一、Vue介绍

## 1.什么是Vue?

Vue是一套用于构建用户界面的渐进式框架。需要了解关于 HTML、CSS 和 JavaScript 的知识;

## 2.兼容性

Vue **不支持** IE8 及以下版本，因为 Vue 使用了 IE8 无法模拟的 ECMAScript 5 特性。但它支持所有[**兼容 ECMAScript 5 的浏览器**](https://caniuse.com/#feat=es5)。

## 3.Vue Devtools

在使用 Vue 时，我们推荐在你的浏览器上安装 [**Vue Devtools**](https://github.com/vuejs/vue-devtools#vue-devtools)。它允许你在一个更友好的界面中审查和调试 Vue 应用。

## 4如何使用Vue?

直接下载并用 <script> 标签引入；

<script src="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>

## 5.声明式渲染

Vue.js 的核心是一个允许采用简洁的模板语法来声明式地将数据渲染进 DOM 的系统：

<div id="app">

{{ message }}

</div>

var app = new Vue({

el: '#app',

data: {

message: 'Hello Vue!'

}

})

1.使用模板语法来声明式地将数据渲染;

2.绑定元素 attribute

2.1通过指令绑定元素的attribute;

v-bind:attribute

注意事项：

1.对应js不能放到head标签里面,因为代码从下网上执行,

页面没有加载完成就进行渲染，导致找不到DOM元素;

4.条件和循环

1.v-if

2.v-else

3.v-else-if

3.v-for

5.处理用户输入

5.1用 v-on 指令添加一个事件监听器，

通过它调用在 Vue 实例中定义的方法;

5.2v-model指令,它能轻松实现表单输入

和应用状态之间的双向绑定。

6.组件化应用构建

6.1在 Vue 里，一个组件本质上是一个拥有预定义选项的

一个 Vue 实例;

二、Vue实例

1.创建Vue实例

a.选项:

1.data

1.)类型:Object|Function

2.)限制:组件的定义只接受 function

2.el

1.)类型:string | Element

2.)限制:只在用 new 创建实例时生效

3.methods:

1.)类型:{ [key: string]: Function }

var vm = new Vue({选项});

三、模板语法

1.插值

a.文本

1.)使用“Mustache”语法 (双大括号) 的文本插值

<div id="app">{{returnResult.name}}</div>

注意事项：

a. Mustache 标签将会被替代为对应数据对象上 msg 属性的值。

无论何时，绑定的数据对象上 msg 属性发生了改变，

插值处的内容都会更新。

b.通过使用 v-once 指令，你也能执行一次性地插值，

当数据改变时，插值处的内容不会更新。

但请留心这会影响到该节点上的其它数据绑定;

<span v-once>这个将不会改变: {{ msg }}</span>

b.原始的html

v-html

3.发送Ajax请求

https://www.cnblogs.com/EricZLin/p/9380406.html

————————————————

版权声明：本文为CSDN博主「pro\_12306」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/qq\_29393273/article/details/105097164