Vue自带的方法creat等可以直接使用，

自定义的方法要在前面加method，否则方法不生效

methods: {  
 fan: function () {  
 this.message = this.message.split('').reverse().join('')  
 this.message1 = '好吧'  
 }  
}

Vue.js 允许你自定义过滤器，被用作一些常见的文本格式化。由"管道符"指示, 格式如下

<p>{{message1|fan1}}</p>

filters: {  
 fan1: function (value) {  
 if (value != null) {  
 value = value.toString()  
 return value.charAt(0).toUpperCase() + value.slice(1)  
 }  
 }  
}

**v-bind 缩写：**

**v-on缩写@**

**split('')** 是分裂的意思，也就是把一个数据拆分。

**reverse()** 是翻转的意思，把数据反过来。

**join('')** 是重组的意思，把数组合成一个字符串。

v-if 有更高的切换消耗而 v-show 有更高的初始渲染消耗。因此，如果需要频繁切换 v-show 较好，如果在运行时条件不大可能改变 v-if 较好。

v-if 是动态添加，当值为 **false** 时，是完全移除该元素，即 **dom** 树中不存在该元素。

v-show 仅是隐藏 / 显示，值为 **false** 时，该元素依旧存在于 **dom** 树中。若其原有样式设置了 **display: none** 则会导致其无法正常显示。

//8.9

v-for 指令需要以 **site in sites** 形式的特殊语法， sites 是源数据数组并且 site 是数组元素迭代的别名。

要在后面加上：key=”site”

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。