

框架前的前端补充

二维码生成

[下载资源文件](#)

步骤:

1. 引入Jquery.js文件
2. 引入jquery.qrcode.js文件
3. 引入支持中文的编码js文件 (utf.js)
4. 在网页中编写一个div 用于显示二维码
`<div id="div1"></div>`
5. 准备二维码的规格对象(JSON)

```
var config = {  
  width: 数字, //值是number类型, 表示的单位是px 必须传递  
  height: 数字, //值是number类型, 表示的单位是px 必须传递  
  text: "内容", //text就表示二维码中存储的数据 必须传递  
  correctLevel: 数字, //取值为0|1|2|3 表示二维码的纠错级别0:L/1:M/2:Q/3:H ,默  
    认0 可选参数  
  background: "#rrggbb", //默认白色, 表示二维码的后景颜色 可选参数  
  foreground: "#rrggbb", //默认黑色, 表示二维码的前景颜色 可选参数  
  render: "绘制模式" //取值:table/canvas , 默认table 可选参数  
};
```
6. 通过选择器, 查找到上述的div ,得到Jquery对象, 通过jquery对象的qrcode函数生成二维码
`$("#div1").qrcode(config);`

layui

布局部分

与Bootstrap很像, 就是展示一些界面效果, 此处省略。

弹出层 layer

步骤:

1. 下载layer , 并将解压后的layer文件夹 移动到项目中
2. 引入jquery.js
3. 引入layer.js

layer - msg函数

用于弹出信息提示框

格式1.

```
layer.msg("文本");
```

格式2. 抖动显示

```
layer.msg("文本", function(){  
  //弹窗结束后会执行  
});
```

layer - load函数

格式:

弹出loading:

```
var index = layer.load(数字0-2);  
// 参数表示 loading的图表  
//loading窗口在弹出时，不允许进行操作。
```

关闭loading:

```
layer.close(index);
```

格式2.

超时自动关闭的loading

```
var index = layer.load(数字0-2,{time:毫秒数字})  
//在指定的毫秒后，如果没有使用layer.close关闭，则自动关闭
```

layer - msg函数(load效果)

格式:

弹出的格式:

```
var index = layer.msg("文本",{icon:16,shade:0.01})  
//因为是msg函数，所以此窗口会自动消失。
```

关闭的格式:

```
layer.close(index);
```

layer - alert函数 信息提示窗

格式:

```
layer.alert("文本内容",{icon:图片编号});
```

//图片编号：0-16之间

layer - tips函数 提示层

格式:

```
layer.tips("文本内容","选择器",{tipsMore:true,tips:数字});
```

参数:

1. 参数: tipsMore : 是否允许同时存在多个弹出的tips
2. 参数: tips : 取值为数字1-4，分别表示弹出在元素的上/右/下/左。默认弹出在右边

layer 所有弹出层的关闭

```
layer.closeAll();
```

作用: 用于在网页中 循环展示固定的布局数据。

格式1.

给元素添加属性: `v-for="item of 数组名"`

格式2.

给元素添加属性: `v-for="(item,index) of 数组名"`

注意：上述格式中的

数组名指的是：**data**中的数据**key**，这个**key**对应的数据必须是数组

item 指的是：每次循环时，从数组中取出的数据的**key**，可以使用插值表达式来显示
{{item}}

index 指的是：循环的轮数，类似数据的下标。

案例：

```
<body>
  <div id="content">
    <ul>
      <li v-for="item of msgs">{{item}}</li>
    </ul>
  </div>
</body>
<script type="text/javascript">
  var v1 = new Vue({
    el:"#content",
    data:{
      msgs:["锄禾日当午","大桥和苍井","停车坐爱枫林晚","一墙红杏出墙
来","哈哈哈哈哈"]
    }
  });
</script>
```