Angular2公共功能使用步骤

# 1 组件

## 1.1 自动关闭Alert 警告

显示警告框，可自定义设置显示时间，默认显示3秒

使用步骤

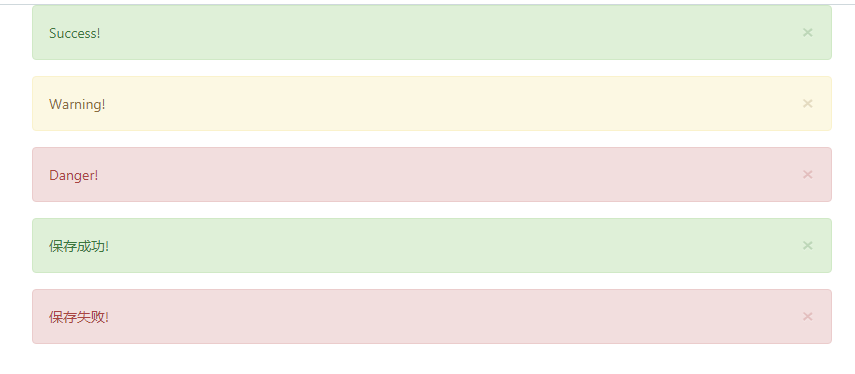
html

|  |
| --- |
| *<!--默认3秒呈现后消失-->* <**message [alerts]="alerts"**></**message**> *<!--设置5秒呈现后消失-->* <**message [alerts]="alerts" [second]="5000"**></**message**> *<!—居中呈现-->* <**message [alerts]="alerts" [isMiddle]="true"** ></**message**> |

ts

|  |
| --- |
| **import** {MessageComponent} **from "../../common/message/message.component"**; **import** {TranslateService} **from "ng2-translate"**;  **export class** DemoComponent **extends** MessageComponent **implements** OnInit {   **constructor**(**private** orderMgtService: OrderMgtService,  **public** translate:TranslateService) {  **super**(translate);  }   ngOnInit() {  **this**.showAlert();  }   showAlert(){  **this**.addAlert(**"Success!"**,**"success"**);  **this**.addAlert(**"Warning!"**,**"warning"**);  **this**.addAlert(**"Danger!"**,**"danger"**);  *//调用国际化信息进行合并 zh-cn.json en-us.json* **this**.addTransAlert(**"common.save.successfully"**,**"success"**);  **this**.addTransAlert(**"common.save.fail"**,**"danger"**);  }    } |

效果



## 1.2 日期控件

datatimerpicker基础上进行控件封装，支持单个日期、其实结束日期范围、时分秒显示。

使用步骤

注： name是必填项  
控件只读，添加属性readOnly="true"   
控件初始化赋值，添加属性value="2017-11-07"

动态绑定日期值 添加双向绑定 [(ngModel)]

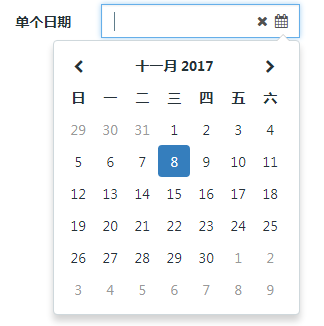
html

|  |
| --- |
| *<!--单个日期-->* <**date-picker name="pickTime1" [singleDatePicker]="true" (onInputChanged)="onInputChanged($event)"**></**date-picker**> *<!--单个日期 带分秒时-->* <**date-picker name="pickTime2" [showTime]="true" [singleDatePicker]="true" (onInputChanged)="onInputChanged($event)"**></**date-picker**> *<!--起始结束日期 不带分秒时-->* <**date-picker name="pickTime3" (onInputChanged)="onInputChanged($event)"**></**date-picker**> *<!--起始结束日期 带分秒时-->* <**date-picker name="pickTime4" [showTime]="true" (onInputChanged)="onInputChanged($event)"**></**date-picker**> |

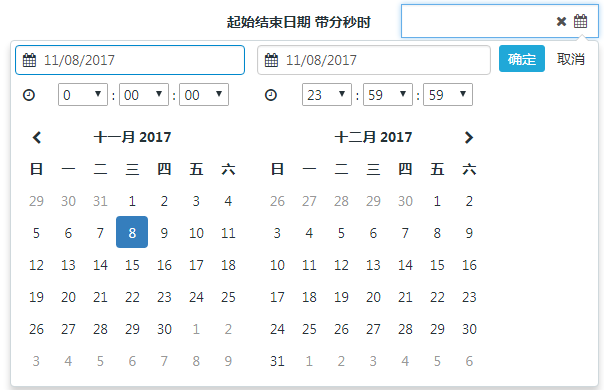
ts

|  |
| --- |
| onInputChanged(value){  ***console***.log(**"修改后的日期"**+value); } |

效果





## 1.3 模态框

代码分析

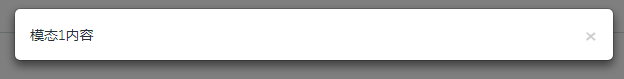
html

|  |
| --- |
| <**button (click)="showModal(*modal1*)"**>显示模态1</**button**> <**button (click)="showModal(*modal2*)"**>显示模态2</**button**>  <**eda-modal #modal1**>  <**div class="body"**>  模态1内容  </**div**> </**eda-modal**>  <**eda-modal #modal2 [width]="400" [height]="200"  [showCloseBtn]="false"  [title]="'标题'"  [confirmName]="'确定'"  [cancelName]="'取消'"  (confirmEmitter)="confirmEmitter(*modal2*)"**>  <**div class="body"**>  模态2内容  </**div**>  <**div class="errormsg"**>  出错信息  </**div**> </**eda-modal**> |

ts

|  |
| --- |
| *//显示模态框* showModal(modal){  modal.show(); }  confirmEmitter(modal){  *//业务逻辑处理  // ...  //关闭模态框* modal.hide(); } |

效果





**属性分析：**

width 设置模态框宽度

height 设置模态框高度

showCloseBtn 默认true进行显示，如设置成false将不显示此关闭按钮

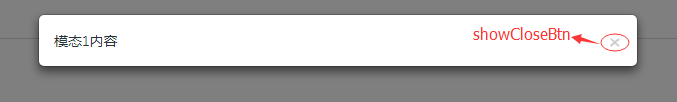
title 标题内容，不填标题将来不呈现

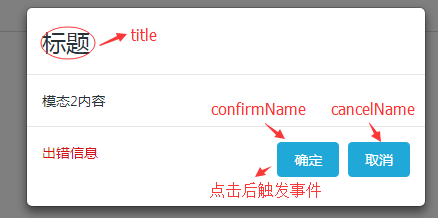
confirmName 确定按钮 无定义时，按钮不呈现

cancelName 取消或关闭按钮名称 无定义时，按钮不呈现

**事件分析：**

confirmEmitter 点击确认后触发事件到父组件





# 2 指令

## 2.1 input标签初始化焦点选中

火狐浏览器通过\*ngIf判断后才加载出来的input标签，设置autofocus属性失效，可以添加focusOnInt指令解决此问题，代码及效果如下。

代码分析

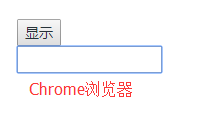
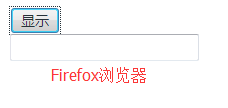
html

|  |
| --- |
| <**button (click)="showInput()"**>显示</**button**> <**div \*ngIf="isShow"**>  <**input type="text" autofocus**> </**div**> |

ts

|  |
| --- |
| **private isShow**:**boolean** = **false**; showInput(){  **this**.**isShow**=**true**; } |

效果

使用focusOnInit指令实现焦点选中，html代码加入focusOnInit属性即可

|  |
| --- |
| <**button (click)="showInput()"**>显示</**button**> <**div \*ngIf="isShow"**>  <**input type="text" focusOnInit**> </**div**> |

# 3 管道

## 3.1 单位换算

用户设置的计量单位不同，很多功能都需要进行单位换算，使用下面管道统一进行单位换算。

html

|  |
| --- |
| <**div**>原值：{{**lengthCm**}} cm ==>  新值: {{**lengthCm** | **unitToInch**:"IMPERIAL"}} inch </**div**> <**div**>原值：{{**weightKg**}} kg ==>  新值: {{**lengthCm** | **unitToLb**:"IMPERIAL"}} lb </**div**> <**div**>  根据登录用户的默认计量单位进行换算：  {{**lengthCm** | **unitToInch**:**unit**}}  <**span \*ngIf="unit=='METRIC'"**>cm</**span**>  <**span \*ngIf="unit=='IMPERIAL'"**>inch</**span**> </**div**> |

ts

|  |
| --- |
| **import** {ContextService} **from "../../../../common/Context"**;  *//数据库中usersetting表的measureunit字段值 //当前登录用户的默认计量单位 （METRIC：默认kg cm单位 IMPERIAL：lb inch单位）* **private unit**:**string** = ContextService.*getUnit*();  **private lengthCm**:**number** = 1; **private weightKg**:**number** = 1; |

效果

|  |
| --- |
|  |

# 4 异常问题

## 4.1 Tab控件内Input标签选中使用异常问题

Tab控件添加select事件，Tab控件内有Input标签，当Input标签选中同时也会触发select事件，此事件会冒泡触发Tab的select事件。

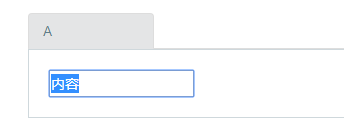
html

|  |
| --- |
| <**tabset**>  <**tab heading="A" (select)="selectTabA()"**>  <**div class="card card-block"**>  <**input type="text" value="内容" (select)="selectInput($event)"**>  </**div**>  </**tab**> </**tabset**> |

ts

|  |
| --- |
| selectTabA(){  ***console***.log(**"select tab A"**); }  selectInput(){  ***console***.log(**"select Input"**); } |

界面选中Input标签



界面console输出，会发现选中Input会冒泡同时触发Tab标签的select事件



解决方法一：

Input标签添加Select事件，在事件内添加阻止冒泡方法。上面ts中的selectInput方法进行优化。

|  |
| --- |
| selectInput($event){  ***console***.log(**"select Input"**);  $event.stopPropagation();*//阻止冒泡* } |

解决方法二：

Tab中有很多Input标签，我们在Tab内的div标签里写个select事件，当div内Input标签触发select事件后，会冒泡到div标签的select事件，这时候在div里进行阻止冒泡操作。

html

|  |
| --- |
| <**tabset**>  <**tab heading="A" (select)="selectTabA()"**>  <**div class="card card-block" (select)="stopSelect($event)"**>  <**input type="text" value="内容"**>  </**div**>  </**tab**> </**tabset**> |

ts

|  |
| --- |
| selectTabA(){  ***console***.log(**"select tab A"**); }  stopSelect($event){  $event.stopPropagation();*//阻止冒泡* } |