

混合应用开发

--- Angular组件



河北师范大学软件学院
Software College of Hebei Normal University

内容提纲

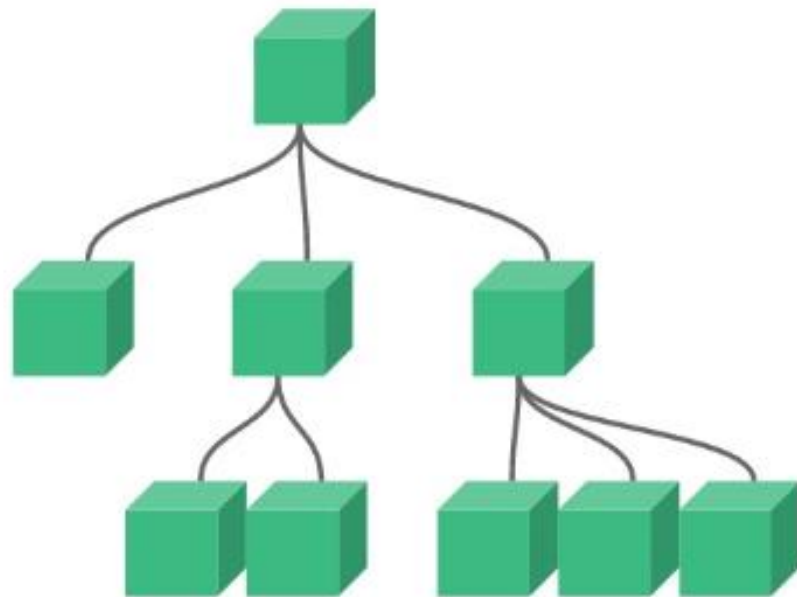
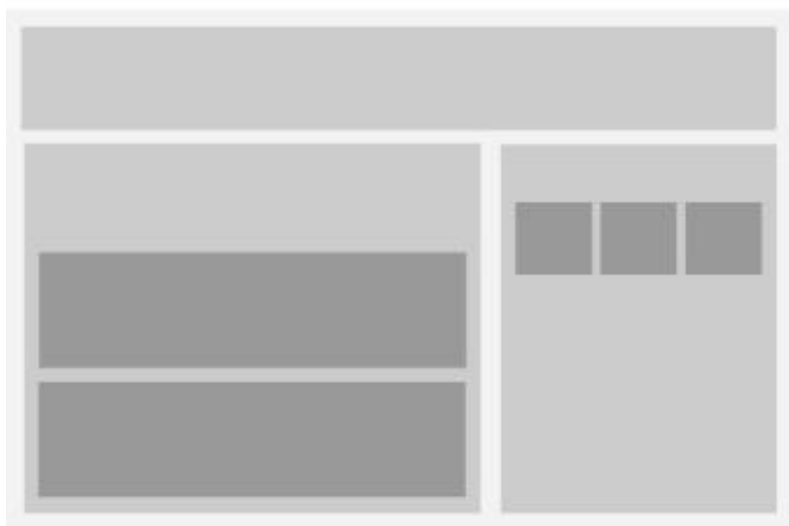
- **组件**
- **生命周期钩子**



组件

- 组件

- 组件负责控制屏幕上的某一块区域，即视图
- 组件是Angular的核心理念，模块化机制是为组件化服务的



组件

- 创建组件

- 命令: `ng g component components/componentName`
- 在AppModule中声明组件
 - 引入: `import { ComponentName } from './components/.....';`
 - 声明: 在 `declarations` 数组中声明
 - 如果通过命令创建组件, **angular**会自动添加声明

- 使用组件

- `<app-组件名> </app-组件名>`

组件

- 组件的基础构成

- 组件装饰器 -- @component()

- 每个组件类必须用 @component 修饰才能成为 Angular 组件

- 组件元数据

- selector、template、styles、providers、templateUrl、styleUrls

- 模板

- 每个组件只能关联一个模板，这个模板会渲染到页面上

- 组件类：定义组件的逻辑

组件

- 组件交互（父组件到子组件） -- @Input
 - 子组件引入 Input
 - import { Input } from '@angular/core';
 - 声明属性
 - @Input() 属性名;
 - 绑定属性
 - <app-组件名 [属性名] = “属性值” ></app-组件名>

组件

- 组件交互（子组件到父组件） -- @Output
 - 子组件引入 EventEmitter , Output
 - import { Component, **EventEmitter**, **Output** } from '@angular/core';
 - 子组件暴露 EventEmitter 属性, 并用@Output () 装饰
 - @Output() eventName = new EventEmitter();
 - 通过 EventEmitter 的 emit 方法传递数据
 - this.eventName.emit(data)

组件

- 组件交互（子组件到父组件）（续）
 - 父组件绑定事件，通过事件对象（\$event）接收数据
 - `<app-组件名 (eventName)= "fun($event)" >`

内容提纲

- 组件
- 生命周期钩子



生命周期

- 生命周期

- Angular每个组件都存在一个生命周期，从创建，变更到销毁

- 生命周期钩子

- Angular提供组件生命周期钩子，把这些关键时刻暴露出来，赋予在这些关键结点和组件进行交互的能力

生命周期钩子

- 生命周期钩子顺序

constructor

ngOnChanges

ngOnInit

ngDoCheck

ngAfterContentInit

ngAfterContentChecked

ngAfterViewInit

ngAfterViewChecked

ngOnDestroy



生命周期钩子

- `ngOnChanges()`
 - 检测到组件输入属性发生变化时调用
- `ngOnInit()`
 - 在构造函数之后马上执行复杂的初始化逻辑
 - 在Angular设置完输入属性之后，对该组件进行准备
 - 只调用一次
- `ngDoCheck()`
 - 检测被Angular忽略的更改



生命周期钩子

- `onDestroy()`
 - 执行组件注销时的清理工作





Thank You