1. Add a new file named *hero-detail.component.ts* to the app folder and create *HeroDetailComponent* as follows.

import { Component, Input } from '@angular/core';

@Component({

selector: 'my-hero-detail',

})

export class HeroDetailComponent {

}

1. Let’s **cut** the Hero Detail content from *AppComponent* and **paste** it into the new template property of *HeroDetailComponent*.
   1. replace *selectedHero* with hero everywhere in our new template

template: `

<div \*ngIf="hero">

<h2>{{hero.name}} details!</h2>

<div><label>id: </label>{{hero.id}}</div>

<div>

<label>name: </label>

<input [(ngModel)]="hero.name" placeholder="name"/>

</div>

</div>

`

1. add that *hero* property we were talking about to the component class.

hero: Hero;

1. relocate the *Hero* class from *app*.*component*.*ts* to its own *hero*.ts file.

export class Hero {

id: number;

name: string;

}

1. Add the following import statement near the top of **both** *app.component.ts* **and** *hero-detail.component.ts.*

import { Hero } from './hero';

1. annotate the hero property with the *@Input* decorator that we imported earlier.

@Input()

hero: Hero;

1. return to the *AppModule*, the application's root module, and teach it to use the *HeroDetailComponent*. Import the *HeroDetailComponent* so we can refer to it.

import { HeroDetailComponent } from './hero-detail.component';

1. add *HeroDetailComponent* to the *NgModule* decorator's declarations array

@NgModule({

imports: [

BrowserModule,

FormsModule

],

declarations: [

AppComponent,

HeroDetailComponent

],

bootstrap: [ AppComponent ]

})

export class AppModule { }

1. Now that the application knows about our *HeroDetailComponent*, find the location in the *AppComponent* template where we removed the Hero Detail content and add an element tag that represents the *HeroDetailComponent*.

<my-hero-detail></my-hero-detail>

1. The two components won't coordinate until we bind the *selectedHero* property of the *AppComponent* to the *HeroDetailComponent* element's *hero* property like this:

<my-hero-detail [hero]="selectedHero"></my-hero-detail>

1. The AppComponent’s template should now look like this

template: `

<h1>{{title}}</h1>

<h2>My Heroes</h2>

<ul class="heroes">

<li \*ngFor="let hero of heroes"

[class.selected]="hero === selectedHero"

(click)="onSelect(hero)">

<span class="badge">{{hero.id}}</span> {{hero.name}}

</li>

</ul>

<my-hero-detail [hero]="selectedHero"></my-hero-detail>

`,