What is entryComponents in angular ngModule?

Asked 3 years ago Active 1 month ago Viewed 54k times

112

33

I am working on an `Ionic` app (`2.0.0-rc0`) which depends on `angular 2` . So the new introduction of `ngModules` is included. I am adding my `app.module.ts.` below.

```
import { NgModule } from '@angular/core';
import { IonicApp, IonicModule } from 'ionic-angular';
import { MyApp } from './app.component';
import { Users } from '../pages/users/users';

@NgModule({
  declarations: [
    MyApp,
    Users
  ],
  imports: [
    IonicModule.forRoot(MyApp)
  ],
  bootstrap: [IonicApp],
  entryComponents: [
    MyApp,
    Users
  ]
})
export class AppModule {}
```

What does `entryComponents` do here? `Components` are already defined in `declarations` . So what's the need of repeating them ? What would happen if I dont include a component here?

angular ionic-framework ionic2

edited Jan 23 '17 at 7:21

asked Sep 28 '16 at 19:35
raj
3,639 4 23 42

7:28

5 Answers

130

✓

This is for dynamically added components that are added using `ViewContainerRef.createComponent()` . Adding them to `entryComponents` tells the offline template compiler to compile them and create factories for them.

The components registered in route configurations are added automatically to `entryComponents` as well because `router-outlet` also uses `ViewContainerRef.createComponent()` to add routed components to the DOM.

Offline template compiler (OTC) only builds components that are actually used. If components aren't used in templates directly the OTC can't know whether they need to be compiled. With entryComponents you can tell the OTC to also compile this components so they are available at runtime.

What is an entry component? (angular.io)

NgModule docs (angular.io)

> Defines the components that should be compiled as well when this component is defined. For each components listed here, Angular will create a ComponentFactory and store it in the ComponentFactoryResolver.

If you don't list a dynamically added component to `entryComponents` you'll get an error message a bout a missing factory because Angular won't have created one.

See also https://angular.io/docs/ts/latest/cookbook/dynamic-component-loader.html

edited Nov 14 '17 at 7:53

answered Sep 28 '16 at 19:37
Günter Zöchbauer
373k 88 1196 1084

10 frankly speaking, I know its 100% correct answer but went bouncer for me, could you please elaborate more? – Pankaj Parkar Sep 28 '16 at 19:39

23 Hard to tell what's unclear. Offline template compiler (OTC) only builds components that are actually used. If components aren't used in templates directly the OTC can't know whether they need to be compiled. With `entryComponents` you can tell the OTC to also compile this components so they are available at runtime. – Günter Zöchbauer Sep 28 '16 at 19:43

3 stackoverflow.com/questions/36325212/… would be such an example – Günter Zöchbauer Sep 28 '16 at 19:48

2 So in general, if component is listed in `declarations` it should also be listed in `entryComponents`, right? – omnomnom Dec 13 '16 at 7:00

27

You won't get explanation better than Angular docs.

And below is the explanation from the angular docs.

> An entry component is any component that Angular loads imperatively by type.
>
> A component loaded declaratively via its selector is not an entry component.
>
> Most application components are loaded declaratively. Angular uses the component's selector to locate the element in the template. It then creates the HTML representation of the component and inserts it into the DOM at the selected element. These aren't entry components.
>
> A few components are only loaded dynamically and are never referenced in a component template.
>
> The bootstrapped root `AppComponent` is an entry component. True, its selector matches an element tag in index.html. But `index.html` isn't a component template and the `AppComponent` selector doesn't match an element in any component template.
>
> Angular loads AppComponent dynamically because it's either listed by type in `@NgModule.bootstrap` or boostrapped imperatively with the module's ngDoBootstrap method.
>
> Components in route definitions are also entry components. A route definition refers to a component by its type. The router ignores a routed component's selector (if it even has one) and loads the component dynamically into a `RouterOutlet`.
>
> The compiler can't discover these entry components by looking for them in other component templates. You must tell it about them by adding them to the `entryComponents` list.
>
> Angular automatically adds the following types of components to the module's `entryComponents`:
>
> - The component in the `@NgModule.bootstrap` list.
> - Components referenced in router configuration.
>
> You don't have to mention these components explicitly, although doing so is harmless.

rewolf
2,482 1 27 39

Mav55
1,877 14 16

Right now the angular docs are not available, so thank SO for that! – Caelum Feb 28 '18 at 11:43

This doesn't seem to mention that components in route configurations are automatically added to entryComponents (so you usually never need to define it). – Connor Jan 29 at 16:11

a link to the actual docs page would be appreciated – rrrafalsz Jul 12 at 9:09

7

The other answers mention this but the basic summary is:

- its needed when a Component is NOT used inside an html template `<my-component />`
- For example when using Angular Material dialog components you use the component indirectly.

Material dialog components are created inside the TS code and not the template:

```
  const dialogRef = this.dialog.open(MyExampleDialog, { width: '250px' });
}
```

This requires you to register it as an entryComponent:

- `entryComponents: [MyExampleDialog]`

Otherwise you get a error:

- `ERROR Error: No component factory found for MyExampleDialog. Did you add it to @NgModule.entryComponents?`

edited Apr 8 at 1:01

answered Jan 3 at 15:30
Mike R
2,878 3 25 39

The best explanation over here. – nop Apr 11 at 14:26

1

hint to find entry component and compile them.

There are two main types of entry components:

- The bootstrapped root component.
- A component you specify in a route definition.

For more detailed information around entry components, please refer angular.io https://angular.io/guide/entry-components

edited Jan 3 at 16:27

answered Jan 3 at 16:00

Vivek
75 12

A Bit of Background about `entryComponent`

1

`entryComponent` is any component Angular loads imperatively. You can declare `entryComponent` by bootstrapping it in `NgModule` or in route definitions.

```
@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    FormsModule,
    HttpClientModule,
    AppRoutingModule
  ],
  providers: [],
  bootstrap: [AppComponent] // bootstrapped entry component
})
```

Documentation says below

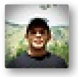
> To contrast the two types of components, there are components which are included in the template, which are declarative. Additionally, there are components which you load imperatively; that is, entry components.

There is `entryComponents` array in `@NgModule` file. You can use this to add `entryComponents` if component is bootstrapped using `ViewContainerRef.createComponent()` .

That is you're creating components dynamically and not by bootstrapping or in template.

```
const componentFactory =
this.componentFactoryResolver.resolveComponentFactory(myComp.component);
const viewContainerRef = this.compHost.viewContainerRef;
viewContainerRef.clear();
const componentRef = viewContainerRef.createComponent(componentFactory);
```

answered Aug 6 at 4:26



Nipuna
3,006 7 42 78