

Angular 17: Introduzione



Introduzione

Cos'è Angular?

Definizione

- Angular è un **framework front-end** sviluppato da Google per la creazione di applicazioni web moderne e scalabili.
- Utilizza **TypeScript** come linguaggio principale.
- Basato su un'architettura a **componenti**, che favorisce il riutilizzo del codice.

Perché usare Angular?

- Struttura chiara e modulare.
- Supporto a SPA (Single Page Applications).
- Eccezionale gestione dello stato e del routing.
- Ampia comunità e supporto a lungo termine.

La struttura di un'applicazione Angular

Struttura dei file

Un'applicazione Angular è composta da diversi file e cartelle chiave:

```
my-angular-app/  
|-- src/  
    |-- app/  
        |-- components/  
        |-- services/  
        |-- app.config.ts  
        |-- app.component.ts  
    |-- assets/  
    |-- environments/  
|-- angular.json  
|-- package.json  
|-- tsconfig.json
```

Struttura dei file

File principali:

- `app.config.ts`: Definisce la configurazione dell'applicazione.
- `app.component.ts`: Il componente principale.
- `angular.json`: Configurazioni generali del progetto.
- `tsconfig.json`: Configurazioni TypeScript.

Comandi CLI Angular

Installazione Angular CLI

```
npm install -g @angular/cli
```

Verificare la versione:

```
ng version
```

Creazione di un nuovo progetto Angular

```
ng new my-angular-app
```

Opzioni comuni:

- `--style=scss` → Specifica il pre-processor CSS.
- `--routing` → Aggiunge il modulo di routing.

Comandi principali CLI

Avviare il server di sviluppo

```
ng serve [-o]
```

il parametro `-o` apre automaticamente il browser

Comandi principali CLI

Creare un nuovo componente

```
ng generate component nome-componente
```

Creare un nuovo servizio

```
ng generate service nome-servizio
```

Creare un nuovo modulo autonomo

```
ng generate component nome-componente --standalone
```

Components in Angular

Cosa sono i Componenti?

- Un **componente** è l'unità base dell'interfaccia utente in Angular.
- Ogni componente ha:
 - Un file TypeScript (`.ts`)
 - Un template HTML (`.html`)
 - Un file di stile CSS/SCSS (`.css` o `.scss`)

Esempio di un componente standalone

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-saluto',
  standalone: true,
  template: `<h1>{{ messaggio }}</h1>`,
  styleUrls: ['./saluto.component.css']
})
export class SalutoComponent {
  messaggio = 'Ciao da Angular!';
}
```

Definire un Componente

Struttura di un componente Angular

Ogni componente è definito con il decoratore `@Component` .

```
@Component({  
  selector: 'app-mio-componente',  
  standalone: true,  
  templateUrl: './mio-componente.component.html',  
  styleUrls: ['./mio-componente.component.css']  
})  
export class MioComponente {  
  titolo = 'Benvenuto in Angular';  
}
```

selector → Il nome HTML del componente.

templateUrl → Il file HTML associato.

styleUrls → Il file CSS associato.

standalone → componente autonomo e non necessita di un modulo.

Creare Componenti con Angular CLI

Creazione di un componente standalone con Angular CLI

```
ng generate component nome-componente --standalone
```

Esempio:

```
ng generate component header --standalone
```

Questo comando genera:

- `header.component.ts`
- `header.component.html`
- `header.component.css`
- `header.component.spec.ts`

Importazione del componente

Nei componenti standalone non si usa più `@NgModule`. I componenti possono essere direttamente importati:

```
import { HeaderComponent } from './header/header.component';

@Component({
  selector: 'app-root',
  standalone: true,
  imports: [HeaderComponent],
  template: `<app-header></app-header>`
})
export class AppComponent {}
```

Conclusione

- Angular è un framework potente per applicazioni web.
- La CLI semplifica la creazione e gestione del progetto.
- I componenti standalone sono la nuova preferenza in Angular 17, eliminando la necessità di `@NgModule`.
- Con la CLI possiamo generare rapidamente componenti modulari e riutilizzabili.

Domande?

Grazie per l'attenzione!

 Q&A Icon

Esempi pratici o chiarimenti?