



ANGULAR FUNDAMENTALS

Il programma del corso

Panoramica

Introduzione al corso (4:44)

Gli argomenti del corso in dettaglio (5:03)

Single Page Applications e approccio component-based (14:37)

Angular Framework: le funzionalità principali (17:17)

Installazione e configurazione

Installazione e utilizzo di NodeJS e NVM (4:16)

Installare Angular CLI e creare un progetto Angular (2:34)

Introduzione al boilerplate (2:05)

AppComponent e AppModule: parte 1 - intro (3:50)

AppComponent e AppModule: parte 2 - code (3:44)

ng new: parametri (1:59)

Fundamentals

Interpolation {{value}} (1:10)

Native events: mouse e tastiera (5:48)

Directives (5:19)

Attributi e parentesi quadre (brackets) (6:29)

Components (5:07)

Pipes (5:17)

Custom Types (5:46)

Introduzione ai service e alla dependency injection (3:52)

Comunicazione con server, REST API e HttpClient (5:03)

Angular Structural Directives

La direttiva ngIf e l'opzione ngIf...else (6:02)

La direttiva ngFor: introduzione e tipizzare collezioni (6:59)

La direttiva ngFor: proprietà index, last, odd (3:30)

La direttiva ngFor: manipolazione dati (5:58)

La direttiva ngSwitch (4:53)

Styling

Styles and Components (4:06)

Inline CSS class (2:08)

La direttiva ngClass (3:15)

Inline Styling (3:27)

La direttiva ngStyle (1:50)

Librerie CSS 3rd party e angular.json (6:12)

Template-driven Form

Input, "template reference variables", keyboard events (7:25)

La direttiva ngModel: 1way vs 2 way binding (5:49)

ngForm, ngModel e gestione data model (13:50)

Form Validation (8:19)

Form, validazione e gestione errori (4:09)

Custom components

Hello Components (5:52)

Input Properties (2:32)

Content Projection (7:35)

Componenti "stateful" e Input default value (2:32)

@Output event emitter: realizzare un TabBar component riutilizzabile (24:10)

Lifecycle hook: ngOnInit (9:23)

Lifecycle Hook: onChange (5:20)

ChangeDetectionStrategy, ChangeDetectorRef e stato immutabile: ottimizzazione performance e controllo del rendering (14:45)

BrowserAnimationModule: animazioni in Angular

Animated Collapsible Panel: trigger, style, state e animate (11:37)

Animated TabBar: animations e ciclo di vita dei componenti (15:58)

Animated Text: gestire gli eventi delle animation (14:02)

Multi-view applications con Angular Router

Introduzione ad angular router (2:03)

Utilizzo di Angular Router e creazione componenti con Angular CLI (6:15)

Navigation Bar (5:33)

AngularCLI e il modulo AppRoutingModuleModule (5:13)

Passaggio parametri e ActivatedRoute (8:30)

ActivatedRoute e Router API (6:23)

Eventi del router & RxJS operators (4:50)

Comunicazione con il server

Configurare un mock server: json-server (5:13)

GET: Caricamento dati da REST API e custom types (9:30)

DELETE: Cancellazione elementi e gestione errori XHR (8:39)

Dynamic styles e migliorare il look & feel (7:45)

POST: Aggiungere elementi alla collezione tramite form (9:55)

PUT e PATCH: aggiornare un elemento (13:34)

Dependency Injection

Dependency Injection, injector e providers (6:01)

Condividere dati utilizzando i "Service" (8:43)

Service, REST API e decoratore @injectable (10:57)

Gestire lo stato applicativo (5:49)

Applicazioni modulari, approfondimento router e lazy loading

Moduli: introduzione (8:23)

Applicazioni Modulari (5:38)

es6 modules vs ngModules (2:21)

Feature Modules (10:12)

Shared Modules (11:25)

Core Module (3:56)

Router & Lazy Loading (11:56)

Moduli e route di secondo livello (5:54)

Router children e router-outlet secondario (5:34)

Build & Deploy

Creare la build del progetto e installare un webserver locale (3:09)

Ottimizzazione build, production e tree shaking (2:41)

Deploy delle applicazioni su Surge.sh (2:37)

Pubblicare un'applicazione Angular su Surge.sh (0:54)

environment variables: production vs development (5:44)

Live Coding / SPA: Hotel-Booking App

Video demo dell'applicazione (0:20)

PARTE 1: SVILUPPO APPLICAZIONE

Layout statico HTML con CSS Grid (+ source code) (8:37)

Creazione progetto, gestione router e impaginazione layout (11:36)

Server REST, comunicazione server, model, gestione form e visualizzazione risultati (23:21)

Card component e pipe: image gallery, forms, maps, rate, ... (34:35)

Gestione del carrello (19:26)

Autenticazione, "sicurezza" router utilizzando le guardie e protezione DOM (25:59)

PARTE 2: APPROCCIO COMPONENT-BASED

Approccio component-based - PART 1: hotel-form, hotel-list e shared components (16:42)

Approccio component-based - PART 2: i componenti rimanenti (16:57)

Approccio component-based - PART 3: card component flessibile e riutilizzabile (HostBinding) (17:02)

Gestire il caso in cui il risultato della ricerca non produca risultati (4:08)

PARTE 3: MODULI e LAZY LOADING

Shared Module (5:57)

Core Module (5:34)

Features Modules (6:19)

Router & Lazy Loading (21:06)

Live Coding / Crud APP: approccio component-based, gestione stato & Best Practices

Demo dell'applicazione (0:11)

Creazione progetto, TSLint, installazione dipendenze e Mock Server (9:46)

Creare "features components" da Angular-CLI e gestione router (4:04)

GET e DELETE: visualizzazione e cancellazione dati dal server (7:40)

PUT, PATCH e POST: modifica e inserimento dati (17:34)

[Download Source Code](#)

Organizzare il layout in componenti: CatalogList (8:05)

Organizzare il layout in componenti: CatalogForm (14:20)

Metodo onChanges del ciclo di vita dei componenti e decoratore @ViewChild (9:17)

[Download Source Code](#)

Organizzazione dell'applicazione in services (5:36)

Gestire lo stato applicativo (8:18)

[Download Source Code](#)

Nested Components: componenti dentro componenti (5:12)

Shared components: componenti riutilizzabili (9:35)

Core Components: Navigation bar e router (4:17)

[Download Source Code](#)

Live Coding / REALTIME APP con Firebase : creare un remote controller per controllare un viewer di slides

Demo dell'applicazione "remote controller" (0:18)

Creazione progetto con router (3:24)

ADMIN: Creazione layout statico - HTML + CSS (16:21)

VIEWER: Creazione layout statico - HTML + CSS (1:48)

Introduzione a Firebase (3:22)

AngularFire: installazione e configurazione (4:09)

CONTROLLER: Creazione della galleria immagini realtime in Firebase (19:31)

CONTROLLER: controllare le slide (7:24)

VIEWER: visualizzazione delle immagini realtime (3:05)

VIEWER: bug fix (9:29)

VIEWER: aggiungere le animazioni (8:44)

VIEWER: animation bug fix

Testare l'applicazione da un dispositivo mobile tramite rete locale durante lo sviluppo:
ng serve --host (1:19)

EXTRA:

StackBlitz: editor online per condividere snippet (3:34)

Altri video in arrivo....