GABRIELE ROMANATO

Novità introdotte in Angular 19

Angular 19 introduce una serie di miglioramenti significativi al framework, mirati a ottimizzare le performance, semplificare lo sviluppo e migliorare l'esperienza degli sviluppatori. Di seguito sono elencate le principali novità.

1. Signal-Based Reactivity API

Una delle principali novità è l'introduzione della nuova **Signal-Based Reactivity API**. Questa API consente una gestione reattiva dei dati più fine e dichiarativa rispetto ai tradizionali RxJS.

```
import { signal, computed } from '@angular/core';

const counter = signal(0);
const doubled = computed(() => counter() * 2);

counter.set(5);
console.log(doubled()); // 10
```

Questa API offre performance migliori e rende il flusso dei dati più chiaro e prevedibile.

2. Standalone Components come default

Con Angular 19, i **Standalone Components** diventano la modalità predefinita per la creazione dei componenti. Questo approccio riduce la necessità di definire moduli esplicitamente.

```
@Component({
  standalone: true,
  selector: 'app-hello',
  template: '<p>Hello World</p>'
})
```

```
}  
export class HelloComponent {}
```

Questa modifica semplifica l'architettura delle applicazioni e migliora il supporto alla modularità.

3. Angular CLI migliorata

L'Angular CLI ora supporta una nuova modalità di scaffolding ottimizzata per progetti standalone e offre suggerimenti più intelligenti durante la creazione di nuovi componenti, servizi e moduli.

```
ng generate component hello --standalone
```

Inoltre, sono stati ridotti i tempi di build e migliorata l'integrazione con strumenti come Vite e esbuild.

4. Integrazione con Vite

Angular 19 introduce il supporto sperimentale a **Vite** come alternativa al tradizionale Webpack, migliorando notevolmente i tempi di startup dell'applicazione in fase di sviluppo.

```
{  
  "builder": "@angular-devkit/build-angular:application",  
  "vite": true  
}
```

Questa funzionalità è ancora in fase di test, ma mostra il forte impegno del team Angular verso esperienze di sviluppo più moderne e veloci.

5. Miglioramenti nella tipizzazione

Grazie alla stretta collaborazione con il team di TypeScript, Angular 19 migliora ulteriormente la **type-safety** dell'intero ecosistema, riducendo il rischio di errori a tempo di compilazione.

```
function greet(name: string): string {  
  return `Hello, ${name}!`;  
}
```

Conclusioni

Angular 19 rappresenta un importante passo avanti nella modernizzazione del framework, offrendo strumenti più potenti e flessibili agli sviluppatori. L'adozione dei Signals, il supporto a Vite e i componenti standalone sono elementi chiave che delineano il futuro di Angular.