Formation Angular - Initiation

Code : DEV-ANGULAR-100 Durée : 3 jours (21 heures)

Public visé

Développeurs et chefs de projets.

{{< formation-prerequis >}}

Objectifs pédagogiques

À l'issue de cette formation, vous serez capable de :

- Utiliser la version 20 du framework Angular
- Développer et tester complètement une application
- Appliquer les bonnes pratiques de développement

Programme

Jour 1 - Matin

Architecture, installation et premier test avec Angular 20

- Architecture moderne d'une application Angular 20
- Application standard (Modules) vs approche avec Standalone Components
- Installation de l'environnement avec Angular CLI v20
- Premier projet : création via CLI et déploiement local avec Vite

TypeScript

- Présentation rapide de TypeScript 5.x
- Types, classes, interfaces et modules ES
- Asynchrone : promesses et async / await

Concepts fondamentaux

- Rôle des composants autonomes (standalone)
- Introduction aux directives : @Component, @Input, @Output
- Architecture MVVM + zones d'exécution sans Zone.js

Travaux pratiques : Création d'un projet Angular 20 et exploration du code généré

Jour 1 - Après-midi

Création de la première application et initiation aux templates

- Démarrer "from scratch" avec ng new et composants standalone
- Création d'un composant à la main

Nouveaux templates syntaxiques

- Interpolation ({{ }})
- Property binding [prop]
- Event binding (event)
- Two-way binding [(ngModel)]

Directives structurelles modernes

- @if, @for, @switch (syntaxe intuitive remplaçant *ngIf*, ngFor)

Services et injection de dépendances

- Pipes intégrés et création de pipes personnalisés
- Création et injection de services (avec inject() recommandé)
- Injection contextuelle via les options de providers

Travaux pratiques : Création d'une application avec des composants autonomes et manipulations de données via services

Jour 2 - Matin

Formulaires Angular avec typage fort

- Template-driven vs Reactive
- Création avec FormsModule et ReactiveFormsModule
- Utilisation de FormBuilder

Formulaires fortement typés

- Nouveauté Angular 17/18+ : TypedFormGroup, FormControl
- Gestion optimisée de la validation
- Affichage dynamique des erreurs

Travaux pratiques : Création de formulaires réactifs modernes avec typage strict

Jour 2 - Après-midi

HTTP, RxJS et accès au backend

- Présentation de RxJS 8+
- Nouveaux opérateurs simplifiés
- Différences entre Promises, Observables et Signals

Requêtes HTTP

- Nouvelle API HttpClient avec provideHttpClient()
- Utilisation de HttpClient.get, post, interceptors, catchError
- Présentation de httpResource
- Accès aux API REST
- Async / await vs subscribe dans les composants

Travaux pratiques : Intégration d'une API REST externe dans un service et affichage dans le composant

Jour 3 - Matin

Routage modulaire et composants autonomes

- Déclaration avec provideRouter()
- Définition des routes dans fichiers séparés
- Techniques d'organisation modulaire : lazy loading, guards, resolvers

Composants autonomes et routage

- Simplification du routage avec composants standalone
- SCAM (Single Component Angular Module) : obsolète depuis Angular 15+
- Chargement différé via loadComponent()

Travaux pratiques : Configuration d'un router avec plusieurs vues (home, about, produit) et navigation avec routerLink

Jour 3 - Après-midi

Signals et tests

- Signals introduits depuis Angular 16+
- Signal(), computed(), effect()
- Réactivité sans RxJS
- Comparaison Signals vs Observables
- Cas d'usage et intégration dans les composants

Tests dans Angular 20

- Tests unitaires avec Jasmine / Karma ou Jest
- Tests de composants standalone
- Simplification de la configuration pour composants autonomes
- E2E testing avec Playwright (préconisé)

Travaux pratiques : Création de signals, tests unitaires de service et composant, tests E2E avec Playwright

Modalités d'évaluation des acquis

En cours de formation, par des études de cas ou des travaux pratiques. En fin de formation, par un questionnaire d'auto-évaluation.

{{< formation-require >}}

{{< formation-seealso >}}