

Angular +

6 modules à découvrir et à combiner selon vos besoins

OBJECTIFS

Pour aller plus loin dans votre expertise sur Angular

PRÉ-REQUIS

Avoir un niveau confirmé sur JavaScript
Etre à l'aise avec la programmation orienté objet
(classes, interfaces, héritage, etc.)

PUBLIC

Développeurs, Lead développeurs

INFORMATIONS PRATIQUES

Durée : 0.5 jour/module

Tarif inter indicatif : 450€ HT

Tarif intra : sur devis

Évaluation : études de cas, travaux pratiques, questionnaires, QCM

[Programme détaillé](#)



Peaks Training Club



formation@peaks.fr



POUR ALLER PLUS LOIN 1/3

Modules complémentaires

MODULE 1 – Aller plus loin avec TypeScript (0.5 j)

Tirer partie de la puissance et des apports de TypeScript :
enums, tuples, inférence, etc

- ✓ **Les nouveaux types**
- ✓ **Exploiter les possibilités avancées**

MODULE 2 – Aller plus loin avec Angular (0.5 j)

Savoir optimiser les performances de son application

- ✓ **Tirer partie du lazy loading**
- ✓ **Comprendre et améliorer la détection des changements**
- ✓ **Les signaux**

MODULE 3 – Débloquer le potentiel de RxJS (0.5 j)

Exploiter le potentiel de RxJS et de la programmation fonctionnelle

- ✓ **Les bonnes pratiques**
- ✓ **Observables vs Promises**
- ✓ **L'intérêt de la programmation fonctionnelle**
- ✓ **Les opérateurs les plus utiles**
- ✓ **Combiner les Observables pour aller plus loin**

POUR ALLER PLUS LOIN 2/3

MODULE 4 – Tester son application avec Jest (0.5 j)

Tester facilement le code de son application avec Jest

- ✓ Qu'est-ce qu'on teste ?
- ✓ Créer un test
- ✓ La couverture du code

MODULE 5 – Structurer son application dans le temps (0.5 j)

Se préparer au changement par l'architecture du code

- ✓ Architectures en couche
- ✓ Bunker de confiance
- ✓ Design patterns
- ✓ Créer et gérer des librairies
- ✓ Le monorepo avec Nx

POUR ALLER PLUS LOIN 3/3

MODULE 6 – Centraliser son state avec NgRx (0.5 j)

Comment créer un cache centralisé et normalisé de ses données grâce au pattern Redux et à NgRx

- ✓ Présentation du pattern Redux et de la notion d'immuabilité
- ✓ Comment le pattern se synergise avec RxJS ?
- ✓ Utiliser NgRx
- ✓ Quels sont les pièges et les limites ?