

## Dart

### Développer une application internet riche (RIA) avec le langage Dart de Google

En créant le langage Dart, l'objectif de Google est de proposer un langage moderne pour le développement d'applications Web.

L'équipe réunie pour la conception de Dart compte des experts reconnus sur les machines virtuelles, sur la génération de JavaScript, sur la conception de langage et sur la création d'environnement de développement.

Le résultat est une plate-forme complète composée :

- d'un langage moderne, tout en restant familier pour les développeurs Java, C# ou C++
- d'une machine virtuelle performante qui sera intégrée à Chrome
- d'un générateur de code JavaScript pour permettre l'exécution des applications Dart dans tous les navigateurs modernes
- d'un environnement de développement basé sur Eclipse

Composée de nombreux exercices, cette formation donne les bases théoriques et pratiques pour maîtriser le langage Dart et ses principales bibliothèques

### Détails

- **Code** : DW-DRT
- **Durée** : 3 jours ( 21 heures )

#### Public

- Architectes
- Chefs de projets
- Développeurs

#### Pré-requis

#### Objectifs

- Maîtriser les bases théoriques et pratiques de Dart
- Maîtriser le langage

### Programme

#### Présentation de Dart

- Intérêt et positionnement de Dart
- Principaux choix de conception du langage
- La machine virtuelle et la conversion en JavaScript
- Chromium, un navigateur pour tester les applications Dart
- Dart Editor, un environnement de développement complet basé sur Eclipse
- Pub, un gestionnaire de bibliothèques

#### Les bases de langages

- Structure d'un programme Dart
- Définition de variables
- La notion de typage optionnel
- Les types de bases
- Définition de fonctions
- Les commentaires
- Les structures de contrôles
- Programmation objet avec Dart
- Programmation fonctionnelle avec Dart
- Programmation asynchrones
- Les exceptions

- La notion de bibliothèques

#### Pub et la gestion des bibliothèques

- Fonctionnement de pub
- Gestion des configurations et des dépendances
- Le fichier pubspec.yaml
- Utilisation de bibliothèques
- Publication de nouvelles bibliothèques
- Gestion des versions

#### Les bibliothèques essentielles

- Présentation de dart:core
- Manipulation des chaînes de caractères
- Collection, List et Map
- Gestion des dates
- Collections avancées avec la bibliothèque dart:collection
- Les calculs mathématiques avec dart:math
- Programmation asynchrone avec dart:async
- Programmation concurrente avec dart:isolate

#### Le développement d'applications Web

- Structure d'une application web cliente en Dart

- Lien avec HTML5
- La librairie dart:html
- Manipulation du DOM
- Gestion des événements
- Envoi de requêtes HTTP
- La librairie dart:json
- Utiliser les WebSockets coté client
- Autres librairies pour exploiter les fonctionnalités HTML5 :dart:indexed\_db, dart:svg, ...
- Présentation de la librairie Web UI
- La notion de WebComponent
- La gestion du data-binding
- Création de templates
- Création et manipulation de composants

### Dart côté serveur

- La librairie dart:io
- Les classes HttpServer et HttpClient
- Implémentation d'une API Rest
- Utiliser les WebSockets coté serveur
- Les possibilités de communication avec des bases de données

### Concepts et librairies avancés

- Les types génériques
- La notion d'interface implicite
- Programmation réflexive avec la notion de Mirror
- Interagir avec du code JavaScript

## Modalités

- **Type d'action** :Acquisition des connaissances
- **Moyens de la formation** :Formation présentielle – 1 poste par stagiaire – 1 vidéo projecteur – Support de cours fourni à chaque stagiaire
- **Modalités pédagogiques** :Exposés – Cas pratiques – Synthèse
- **Validation** :Exercices de validation – Attestation de stages