Les bases du langage Swift pour le développement Mac OS X et iOS

★★★★★

SII-361 **2 Jours (14 Heures)**

Description

Nouveau langage de programmation créé par Apple, Swift se présente comme une alternative à Objective-C, moderne et plus performante. Vous apprendrez dans cette formation à développer des Swift 3 en environnement MacOS. Vous découvrirez les concepts de base de Swift 2 et ses aspects novateurs.

À qui s'adresse cette formation ?

Pour qui

Développeurs, chefs de projet.

Prérequis

Aucun

Les objectifs de la formation

- Découvrir les concepts de base du langage Swift
- Créer un projet SwiftAppréhender Swift Standard Library
- Comprendre comment intégrer Cocoa et du code Objective-C à une application Swift

Programme de la formation

Introduction à Swift

- Pourquoi un nouveau langage de programmation dans l'univers Apple ? Les différents cas d'usage de Swift (Mac OS X, iOS).
- Les apports et nouveautés de Swift.
- Le rapport de Swift avec les outils Apple classiques : Objective-C, XCode.
- Ouverture à l'Open Source et aux autres systèmes.

L'environnement de développement

- Découverte de l'environnement XCode, nouveautés de XCode 8, migrer depuis Swift 1 et 2 (Swift Migrator).
- Installation et configuration de l'environnement de développement, ciblage d'une version d'OS spécifique.
- Création d'un projet Swift et opérations de base (édition, exécution.
-).
- Découverte de l'outil d'évaluation et de mise au point REPL (Read-Eval-Print-Loop).
- Travaux pratiques Mise en place de l'environnement de développement et découverte d'un premier projet.

Les concepts de base de Swift

- Philosophie du langage : syntaxe, gestion mémoire, inférence de type, orientation objet ou fonctionnelle.
- Les mécanismes classiques : typage, variables et constantes, instructions, fonctions.
- Les concepts Objet : notion de classes, méthode.
- La manipulation des chaînes de caractères.
- Travaux pratiques Réalisation de plusieurs applications mettant en oeuvre les concepts exposés.

Swift Standard Library

- Les types : String, Array, Dictionnary et numériques.
- Les protocoles : Equatable, Comparable, Printable.
- Les fonctions libres et algorithmes.
- Travaux pratiques Mise en oeuvre des concepts.

Les aspects novateurs et avancés

- Les mécanismes de typage de Swift, la généricité, le support des protocoles.
- La notion de terrain de jeu (Interactive Playground).
- Les clôtures (closures).
- La gestion mémoire : constructeur, destructeur, gestion de références, lazy instanciation.
- La gestion des erreurs.
- Le multi-threading.
- Introduction aux interfaces graphiques avec Swift (Cocoa, Cocoa Touch, iOS) Le futur avec Swift 3.
- Travaux pratiques Découverte du Playground.
- Mise en pratique des closures et de la généricité.
- Utilisation des frameworks Cocoa et Cocoa Touch.

Les aspects novateurs et avancés

- Les mécanismes de typage de Swift, la généricité, le support des protocoles.
- La notion de terrain de jeu (Interactive Playground).
- Les clôtures (closures).
- La gestion mémoire : constructeur, destructeur, gestion de références, lazy instanciation.
- La gestion des erreurs.
- Le multi-threading.
- Introduction aux interfaces graphiques avec Swift (Cocoa, Cocoa Touch, iOS) Le futur avec Swift 3.
- Travaux pratiques Découverte du Playground.
- Mise en pratique des closures et de la généricité.
- Utilisation des frameworks Cocoa et Cocoa Touch.

L'intégration avec Cocoa et Objective-C

- L'exploitation des frameworks Cocoa et Cocoa Touch depuis une application Swift.
- L'utilisation conjointe de Swift et Objective-C.
- Les trois modes de fonctionnement : Interoperability, Mix and Match, Migration.
- Travaux pratiques Intégration de code Swift et Objective-C.