

## LANGAGE SWIFT

<b>Durée</b>	<b>3 jours</b>	<b>Référence Formation</b>	<b>4-LG-SWIFT</b>
--------------	----------------	----------------------------	-------------------

### Objectifs

Appréhender les concepts de base du langage Swift.  
Créer un projet Swift.  
Utiliser la Swift Standard Library.

### Participants

Développeurs, chefs de projet.

### Pré-requis

Pratique d'un langage de programmation structurée.

### Moyens pédagogiques

Accueil des stagiaires dans une salle dédiée à la formation équipée d'un vidéo projecteur, tableau blanc et paperboard ainsi qu'un ordinateur par participant pour les formations informatiques.  
Positionnement préalable oral ou écrit sous forme de tests d'évaluation, feuille de présence signée en demi-journée, évaluation des acquis tout au long de la formation.  
En fin de stage : QCM, exercices pratiques ou mises en situation professionnelle, questionnaire de satisfaction, attestation de stage, support de cours remis à chaque participant.  
Formateur expert dans son domaine d'intervention  
Apports théoriques et exercices pratiques du formateur  
Utilisation de cas concrets issus de l'expérience professionnelle des participants  
Réflexion de groupe et travail d'échanges avec les participants  
Pour les formations à distance : Classe virtuelle organisée principalement avec l'outil ZOOM. Assistance technique et pédagogique : envoi des coordonnées du formateur par mail avant le début de la formation pour accompagner le bénéficiaire dans le déroulement de son parcours à distance.

### PROGRAMME

#### Swift

- Pourquoi un nouveau langage ?
- Ses apports et nouveautés.
- Les playgrounds : terrains de jeu.
- Intégration aux outils existants.
- Swift système ouvert.

#### XCode l'environnement de développement

- Présentation de l'outil et ses addons.
- L'utilisation de XCode.

- Interface Builder et les divers outils disponibles.
- Le débogueur.
- Les outils de mesure : instruments.
- Les portails du développeur Apple.

### **Les bases du langage Swift**

- Les fonctionnements du compilateur et l'architecture.
- Les bases : variables, constantes et tuples, les chaînes de caractères.
- L'optional et sa gestion, l'unwrapped.
- Les opérateurs et la surcharge.
- Les structures de contrôle.
- Les closures et les fonctions.
- Classes, structures, énumérations, propriétés.
- L'héritage et les protocoles.

### **La bibliothèque Swift**

- Que contient la Swift Standard Library ?
- Gestion des tableaux, dictionnaires et Set.
- Les protocoles Equatable et Comparable : utilisation.
- Autres classes et structures disponibles.

### **Concepts avancés en Swift**

- Les Subscripts : concept et implémentation.
- La gestion des erreurs et exceptions : try, catch, throw, throws, try, defer.
- La gestion de l'asynchronisme avec Grand Central Dispatch.
- La classe Thread, endormir un thread, annuler un thread, notification.
- Les références strong, weak et unowned.
- Le Lazy Loading.
- Créer des extensions en Swift.
- La généricité et les génériques.

### **Interaction Swift et Objective-C**

- Comparaison des syntaxes entre Objective-C et Swift.
- Comparaison des types de données.
- Mixing and matching.
- Héritage entre Swift et Objective-C.
- Points particuliers et limitations.



**CAP ÉLAN FORMATION**

[www.capelanformation.fr](http://www.capelanformation.fr) - Tél : 04.86.01.20.50

Mail : [contact@capelanformation.fr](mailto:contact@capelanformation.fr)

Organisme enregistré sous le N° 76 34 0908834

version 2025