

## C# NIVEAU 1

**Durée**

**3 jours**

**Référence Formation**

**4-LC-CSBA**

### Objectifs

Apprendre à développer avec C#

Créer des interfaces de gestion de bases et manipuler les objets de la plate-forme .NET

Cette formation permettra d'avoir de solides bases pour développer des applications en C#, apprendre à utiliser le Framework .NET ainsi que la mise en œuvre de la Programmation Orienté Objet (POO)

### Participants

Cette formation conviendra aux développeurs ayant des notions de programmation et des notions sur l'orienté objet

### Pré-requis

Une expérience requise en développement logiciel avec un langage de type C/C++ ou Java est appréciée

### 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

#### Créer et utiliser des classes

- Déclarer une classe en C#
- Utiliser des différences indicateurs de visibilité
- Déclarer des constructeurs pour initialiser des objets
- Savoir comment ajouter des membres et des méthodes statiques

#### Maîtriser les types valeurs et les types références

- Appréhender la différence entre type valeur et type référence
- Savoir comment faire pour copier un type valeur

- Utiliser des types valeur non initialisés
- La technique pour écrire des méthodes qui retournent plus d'un paramètre en sortie
- Savoir comment la mémoire dans votre application est organisée
- Maîtriser le boxing et l'unboxing
- Comprendre comment faire des cast et surtout comment éviter les erreurs: vous serez capables de choisir facilement entre type valeur et type référence

### **Créer des types valeurs**

- Déclarer et utiliser des énumérations
- Convertir une énumération en entier
- Pourquoi les constantes sont à éviter dans les applications, comment créer des structures et savoir quelles sont les différences entre les structures et les classes

### **Bien utiliser les tableaux**

- Déclarer et utiliser des tableaux en C#
- Créer des tableaux multidimensionnels
- Créer des tableaux non rectangulaires
- Optimiser l'occupation mémoire pour de grands tableaux

### **Travailler avec l'héritage**

- Utiliser l'héritage de classes en C#
- Comprendre comment déclarer des constructeurs
- Savoir pourquoi le constructeur par défaut peut disparaître
- Maîtriser les appels du constructeur parent
- Comment faire passer un objet pour un autre avec l'héritage

### **Les interfaces et les classes abstraites**

- Savoir et comprendre l'intérêt des interfaces en C#, la méthode simple pour créer une interface avec Visual Studio
- Comment implémenter une interface rapidement avec Visual Studio
- Comprendre comment déclarer et utiliser les classes abstraites

### **Gérer les erreurs et les exceptions**

- Comment faire pour gérer facilement les erreurs dans votre application
- Bien comprendre l'intérêt des exceptions
- Savoir comment faire pour lancer des exceptions
- Comprendre l'intérêt de filtrer proprement les exceptions possibles, la technique pour écrire du code propre, même en cas d'exception
- Bien comprendre les structures try/catch et try/catch/finally

### **Le Garbage Collector et les ressources**

- Comprendre le fonctionnement du Garbage Collector pour la mémoire
- Savoir comment faire pour éviter d'écrire des applications trop gourmandes en mémoire, la technique simple pour libérer des ressources
- Comprendre comment utiliser le pattern IDisposable pour simplifier l'écriture de votre code
- Implémenter facilement une classe IDisposable

### Les propriétés en C#

- Comprendre pourquoi les propriétés sont importantes dans vos classes, l'intérêt d'utiliser les propriétés automatiques
- Comment gérer les accès en lecture ou en écriture
- Comment intégrer des propriétés dans des interfaces, la technique pour initialiser rapidement des objets