

## C# débutant



### OBJECTIFS PÉDAGOGIQUES

- 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)



### PUBLIC CONCERNÉ

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



### PRÉREQUIS

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



### MOYENS PÉDAGOGIQUES

- Réflexion de groupe et apports théoriques du formateur
- Travail d'échange avec les participants sous forme de réunion-discussion
- Utilisation de cas concrets issus de l'expérience professionnelle
- Validation des acquis par des questionnaires, des tests d'évaluation, des mises en situation et des jeux pédagogiques
- Alternance entre apports théoriques et exercices pratiques (en moyenne 30 à 50%)
- Remise d'un support de cours.



### MODALITÉS D'ÉVALUATION

- Feuille de présence signée en demi-journée,
- Evaluation des acquis tout au long de la formation,
- Questionnaire de satisfaction,
- Attestation de stage à chaque apprenant,
- Positionnement préalable oral ou écrit,
- Evaluation formative tout au long de la formation,
- Evaluation sommative faite par le formateur ou à l'aide des certifications disponibles



### MOYENS TECHNIQUES EN PRÉSENTIEL

- Accueil des stagiaires dans une salle dédiée à la formation équipée à minima d'un vidéo projecteur et d'un tableau blanc et/ou paperboard.
- Pour les formations nécessitant un ordinateur, un PC est mis à disposition de chaque participant.



### MOYENS TECHNIQUES EN DISTANCIEL

- A l'aide d'un logiciel (Teams, Zoom...), d'un micro et éventuellement d'une caméra les apprenants interagissent et communiquent entre eux et avec le formateur.
- Sessions organisées en inter comme en intra entreprise.
- L'accès à l'environnement d'apprentissage ainsi qu'aux preuves de suivi et d'assiduité (émargement, évaluation) est assuré.
- Pour toute question avant et pendant le parcours, assistance technique à disposition au 04 67 13 45 45.



### ORGANISATION

- Délai d'accès : 5 jours ouvrés (délai variable en fonction du financeur)
- Les cours ont lieu de 9h à 12h30 et de 13h30 à 17h



### ACCESSIBILITÉ

- Les personnes en situation de handicap sont invitées à nous contacter directement, afin d'étudier ensemble les possibilités de suivre la formation.
- Pour tout renseignement, notre référent handicap reste à votre disposition : mteyssedou@ait.fr



### PROFIL FORMATEUR

- Nos formateurs sont des experts dans leurs domaines d'intervention
- Leur expérience de terrain et leurs qualités pédagogiques constituent un gage de qualité.



### CERTIFICATION POSSIBLE

- Aucune

# C# débutant

## 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