

# C# - Programmation en C#

5 j (35 heures)

Ref : 20483

## Public

Développeur .Net

## Pré-requis

Avoir de l'expérience dans l'utilisation de C# pour réaliser des tâches de programmation de base

## Moyens pédagogiques

Formation réalisée en présentiel ou à distance selon la formule retenue  
Exposés, cas pratiques, synthèse, assistance post-formation pendant trois mois  
Un poste par stagiaire, vidéoprojecteur, support de cours fourni à chaque stagiaire

## Modalités de suivi et d'évaluation

Feuille de présence émargée par demi-journée par les stagiaires et le formateur  
Exercices de mise en pratique ou quiz de connaissances tout au long de la formation permettant de mesurer la progression des stagiaires  
Questionnaire d'évaluation de la satisfaction en fin de stage  
Auto-évaluation des acquis de la formation par les stagiaires  
Attestation de fin de formation

## Objectifs

- Créer et appeler des méthodes, gérer les exceptions et décrire les besoins de surveillance
- Créer des classes, des collections génériques, définir et mettre en oeuvre des interfaces
- Utiliser l'héritage pour créer une nouvelle hiérarchie de classe
- Accéder à une base de données pour requêter et mettre à jour des données
- Accéder et requêter des données à distance
- Construire une interface utilisateur graphique
- Intégrer les bibliothèques non gérées et les composants dynamiques
- Créer et utiliser des attributs personnalisés
- Générer des codes d'exécution et gérer des versions d'assembly
- Chiffrer et déchiffrer des données

## Programme détaillé

### SYNTAXE C#

---

Vue d'ensemble de l'écriture d'applications en utilisant C#

Types de données, opérateurs et expressions

Constructions du langage de programmation C#

## **CREATION DE METHODES, GESTION DES EXCEPTIONS ET SURVEILLANCE DES APPLICATIONS**

---

Créer et invoquer des méthodes

Créer des méthodes surchargées et utiliser les paramètres optionnels et de sortie

Gérer les exceptions

Surveiller les applications

## **DEVELOPPEMENT DU CODE POUR UNE APPLICATION GRAPHIQUE**

---

Mettre en œuvre des structures et des énumérations

Organiser des données dans des collections

Gérer des événements

## **CREATION DE CLASSES ET MISE EN ŒUVRE DES COLLECTIONS GENERIQUES**

---

Créer des classes

Définir et mettre en œuvre des interfaces

Mettre en œuvre des collections génériques

## **CREATION D'UNE HIERARCHIE DE CLASSES EN UTILISANT L'HERITAGE**

---

Créer des hiérarchies de classes

Etendre des classes du framework .NET

Créer des types génériques

## **LECTURE ET ECRITURE DE DONNEES LOCALES**

---

Lire et écrire dans des fichiers

Sérialiser et dé-sérialiser des données

Réaliser des traitements I/O en utilisant des flux

## **ACCES A UNE BASE DE DONNEES**

---

Créer et utiliser des modèles d'entités de données

Interroger des données en utilisant LINQ

Mettre à jour des données en utilisant LINQ

## **ACCES A DES DONNEES DISTANTES**

---

Accès à des données via le Web

Accès à des données dans le Cloud

## **CONCEPTION D'UNE INTERFACE UTILISATEUR POUR UNE APPLICATION GRAPHIQUE**

---

Utiliser XAML pour concevoir une interface utilisateur  
Lier des contrôles à des données  
Styliser une interface utilisateur

## **AMELIORATION DE LA PERFORMANCE DES APPLICATIONS ET DES TEMPS DE REPONSE**

---

Mettre en œuvre le multitâche en utilisant les tâches et les expressions Lambda  
Réaliser des opérations asynchrones  
Synchroniser l'accès concurrent aux données

## **INTEGRATION AVEC LE CODE NON GERE**

---

Créer et utiliser des objets dynamiques  
Gérer la durée de vie des objets et contrôler les ressources non gérées

## **CREATION DE TYPES REUTILISABLES ET D'ASSEMBLAGES**

---

Examiner les métadonnées des objets  
Créer et utiliser les attributs personnalisés  
Générer du code managé  
Gérer les versions, signer et déployer les assemblages

## **CHIFFREMENT ET DECHIFFREMENT DES DONNEES**

---

Mettre en œuvre le chiffrement symétrique  
Mettre en œuvre le chiffrement asymétrique

---