

## Formation C# - Initiation

|   |   |
|---|---|
| <b>Durée :</b>                                | 3 jours   |
| <b>Public :</b>                               | Tous  |
| <b>Pré-requis :</b>                           | Notions de programmation  |
| <b>Objectifs :</b>                            | Apprendre à développer avec C#, à créer des interfaces de gestion de bases et à manipuler les objets de la plate-forme .NET |
| <b>Sanction :</b>                             | Attestation de fin de stage mentionnant le résultat des acquis  |
| <b>Taux de retour à l'emploi:</b>             | Aucune donnée disponible  |
| <b>Référence:</b>                             | .NE142-F  |
| <b>Note de satisfaction des participants:</b> | 4,76 / 5  |

### Découvrir la plateforme .NET

Présentation de .NET  
Les langages du framework .NET  
Particularités du C#  
Evolutions du C#, plateformes supportées  
Environnement de développement  
Types de projets : console, bibliothèque, WinForms, WPF, WCF, ...  
Structure minimale d'un programme C#  
Compilation, exécution et débogage

**Atelier : Manipulation de l'environnement de développement - exemples de programmes C#**

### Syntaxe de base

Variables : déclaration, types communs (Common Type System)  
Organisation en mémoire (types primitifs / types références)  
Opérateurs  
Conversion de types : transtypage implicite / explicite  
Instructions conditionnelles  
Instructions d'itération, les boucles

**Atelier : Multiples exemples d'applications**

### Tableaux

Principe de fonctionnement  
Création de tableaux quelconques  
Tableaux multidimensionnels et parcours

## **Atelier : Déclaration et manipulations de tableaux**

### **Méthodes et paramètres**

Introduction au procédural : définition, intérêt

Types et retours de fonctions

Passage de paramètres :

- par valeur

- par référence : (ref, out)

- arguments variables (params)

Surcharge de méthodes

Récursivité : définition, implémentation

### **Atelier : Découpage de l'application**

### **Gestion des exceptions**

Mécanisme d'exceptions, types

Capter et traiter une exception (try/catch/finally)

Déclenchement d'exceptions (throw)

### **Atelier : Gestions des erreurs de saisie, remontées d'exceptions**

### **Classes fondamentales à disposition**

Notions sur les classes

Les chaînes de caractères, les dates,...

Les listes et collections diverses, algorithmes

L'accès aux fichiers et les flux

### **Atelier : Manipulation de collections et de flux**

### **Construire des applications de bureau**

Présentation de Windows Forms (WinForms)

Présentation de WPF

Création de fenêtres et communication

Utilisation de contrôles

Binding de contrôles avec des données

Utilisation de contrôles utilisateurs

Utilisation de tâches

### **Atelier : une application d'annuaire**