

Initiation à la program- mation en Visual Basic



Spoc pratique
Travaux pratiques
continus



Durée
3 jours



Votre animateur
Philippe Brutus



Code Kertify : **PVB**

S'inscrire



www.kertify.com/inscription

Apprendre à programmer une application de bureau avec Visual Basic

Cette formation présente les bases de la programmation et met en œuvre les différentes phases de la production d'un programme, en utilisant le langage Visual Basic et l'environnement Visual Studio.

Les objectifs de la formation

- ◆ Décomposer un problème et structurer un programme ;
- ◆ Maîtriser les éléments du langage Visual Basic (mots-réservés, syntaxe) pour écrire un programme ;
- ◆ Compiler et exécuter un programme ;
- ◆ Tester et mettre au point un programme ;
- ◆ Accéder à une base de données ;
- ◆ Comprendre les bases de la programmation par objets et savoir utiliser des objets prédéfinis.

Participants

- ◆ Toute personne ayant à développer des applications pour automatiser certaines activités

Prérequis

- ◆ Aucune connaissance préalable n'est requise

Le programme de notre formation

1. Les bases de la programmation

- ◆ Notions de programme et de langage de programmation
- ◆ Compilation et exécution
- ◆ Responsabilités d'un programmeur
- ◆ Notion d'algorithme et pseudo-langage de programmation
- ◆ Développer une application pour répondre à des besoins

2. Production d'une application

- ◆ Définition d'un projet
- ◆ Ecriture du code (syntaxe et instructions)
- ◆ Compilation et exécution
- ◆ Production du kit de distribution
- ◆ Notion de bibliothèque : rôle et usage

3. Règles de programmation

- ◆ Conventions de nommage
- ◆ Conventions syntaxiques
- ◆ Utilisation des commentaires
- ◆ Maîtrise du code par découpage en sous-programmes
- ◆ Amélioration de la lisibilité du code par l'indentation

4. Les variables

- ◆ Notion de variable et de constante
- ◆ Types de données : panorama des types, rôle et conséquences du typage
- ◆ Types primitifs : nombres entiers et à virgule, chaînes de caractères...
- ◆ Déclaration, initialisation et affectation d'une variable
- ◆ Instructions de saisie, affichage, affectation, conversion de type
- ◆ Gérer plusieurs valeurs de même type dans un tableau
- ◆ Gérer plusieurs valeurs de types différents dans un enregistrement

5. Opérateurs et expressions

- ◆ Opérateur d'affectation
- ◆ Opérateurs arithmétiques
- ◆ Opérateurs de comparaison (sur des nombres, sur des textes)
- ◆ Opérateurs logiques (expressions composées)
- ◆ Expression booléenne

6. Structures de contrôle

- ◆ Conditionnelles (si-alors, si-alors-sinon, selon)
- ◆ Itératives (pour, répéter-jusqu'à, tant-que-faire)
- ◆ Bloc d'instruction (début et fin de bloc)
- ◆ Imbrication des structures de contrôles (boucle dans une conditionnelle, conditionnelle dans boucle...)

7. Sous-programmes

- ◆ Notion de sous-programme : définition, rôle et importance
- ◆ Différents types de sous-programmes : procédures et fonctions
- ◆ Appel de sous-programme
- ◆ Paramètres de sous-programme, arguments d'appel de sous-programme, mode de passage
- ◆ Résultat d'une fonction : instruction de retour

8. Bases de la programmation par objets

- ◆ Notion d'objet : type-objet, données et opérations
- ◆ Représentation schématique des objets
- ◆ Déclaration, création et utilisation d'objets
- ◆ Bonnes pratiques d'organisation d'un programme

9. Accès à une base de données

- ◆ Organisation des données en vue de leur stockage
- ◆ Notion de requête (insertion, mise à jour, interrogation, suppression)
- ◆ Opérations de base : connexion, soumission d'une requête, exploitation de la réponse
- ◆ Application cliente et utilisation d'une base de données
- ◆ Exploitation des données dans l'application cliente

10. Maintenance corrective, mise au point et test des programmes

- ◆ Message d'erreur : interprétation du message, localisation de l'erreur

- ♦ Utiliser un outil de mise au point : exécution pas à pas, inspection des variables, points d'arrêt
- ♦ Tests unitaires : définition, conception