Stage pratique de 3 jour(s)
Réf : INP

Participants

Toute personne devant apprendre à programmer.

Pré-requis

Aucune connaissance particulière.

Prix 2018 : 1920€ HT

Dates des sessions

Paris

26 fév. 2018, 14 mai 2018
24 sep. 2018, 10 déc. 2018

Modalités d'évaluation

L'évaluation des acquis se fait tout au long de la session au travers des multiples exercices à réaliser (50 à 70% du temps).

Compétences du formateur

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

Moyens pédagogiques et techniques

• Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.

• A l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.

• Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin

Introduction à la programmation avec Visual Basic

Best

Ce stage vous permettra de comprendre les fondements de la programmation et de l'algorithmique. Vous acquerrez des bases en programmation qui vous permettront d'aborder n'importe quel langage dans les meilleures conditions. Tous les aspects essentiels seront vus : les modèles de programmation, les éléments de lexique et de syntaxe, les outils, l'organisation du code, l'accès aux bases de données et les tests.

OBJECTIFS PEDAGOGIQUES

Structurer des programmes selon un algorithme
Maîtriser les éléments de lexique et de syntaxe d'un langage pour écrire un programme
Compiler et exécuter un programme
Déboguer et tester un programme
Accéder à une base de données
Comprendre les grands principes de la programmation orientée objet

Travaux pratiques

Ce stage contient plus de 60 % de travaux pratiques effectués selon les besoins en Visual Basic (cours réf INP), en Java (cours réf INJ), en C# (cours réf OGR) ou en Python (cours réf THO).

1) Les fondements de la programmation

- Qu'est-ce qu'un programme ? Qu'est-ce qu'un langage ? Les différents paradigmes. Quel langage pour quelle application ?
- Les compilateurs. Les exécutables.
- Les responsabilités d'un programmeur.
- Qu'est-ce qu'un algorithme ?
- Les besoins auxquels répond un algorithme.
- Le concept de pseudo-langage.

Travaux pratiques
Présentation de différents langages (Java, C#, Visual Basic, C, C++). Ecriture d'un premier algorithme en pseudo-langage.

2) Genèse d'un premier programme

- Ecriture d'un programme simple : syntaxe et instructions.
- Compilation et exécution du programme.
- Qu'est-ce qu'une librairie ? Son rôle, son usage.

Travaux pratiques
Découverte de l'environnement de développement et d'exécution. Ecriture, compilation et exécution d'un premier programme.

3) Règles de programmation

- Convention de nommage.
- Convention syntaxique.
- Utilisation des commentaires. Pourquoi commenter les développements ?
- Améliorer la lisibilité des programmes : indentation du code, découpage du code...

4) Les variables

- Qu'est-ce qu'une variable ?
- Pourquoi typer une variable ?
- Les types primitifs : entiers, chaînes de caractères, nombres réels, autres.
- Déclaration, définition et initialisation d'une variable.
- Les constantes.
- Saisie, affichage, affectation, conversion de type.
- Organiser ses données sous forme de tableaux.
- Les types évolués : enregistrement, matrice, arbre.

Travaux pratiques
Ecriture de plusieurs programmes simples manipulant les variables.

de formation si le stagiaire a bien assisté à la totalité de la session.

5) Opérateurs et expressions

- Les différents opérateurs (multiplicatif, additif, comparaison, égalité, logique, affectation).
- Combinaison d'opérateurs.
- Expression booléenne.

Travaux pratiques
Manipulation des opérateurs et des expressions booléennes.

6) Les structures de contrôle

- Les sélections alternatives (si, si-alors-sinon, sélection cas).
- Les blocs d'insctructions (notion de début... fin).
- Les boucles itératives (tant que-répéter, répéter-jusqu'à, pour-de-à).
- Imbrication des instructions.
- Les commentaires.

Travaux pratiques
Utilisation des structures de contrôle pour implémenter un algorithme.

7) Les procédures et les fonctions

- Définitions : procédure, fonction.
- Pourquoi sont-elles incontournables en programmation (réutilisabilité, lisibilité...) ?
- Le passage de paramètres.
- Le code retour d'une fonction.
- Sensibilisation aux limites du passage de la valeur d'une variable.
- Notion de passage par adresse.
- Appel de fonctions.

Travaux pratiques
Debugging de programmes exemples.

8) Introduction à la programmation objet

- Les concepts associés à la programmation objet : classe, attribut, méthode, argument.
- La modélisation objet à partir des exigences fonctionnelles.
- Introduction aux bonnes pratiques d'organisation de conception et d'organisation d'un programme.

Travaux pratiques
Illustration des concepts objets.

9) L'accès aux bases de données

- Organisation et stockage des données.
- Les traitements de base (connexion, requêtes, récupération des données).
- Application cliente et serveur de données.
- Affichage et manipulation des données dans l'application cliente.

Travaux pratiques
Création d'un formulaire de recherche d'informations dans une base de données.

10) Maintenance, débogage et test des programmes

- Savoir lire et interpréter les différents messages d'erreurs.
- Utiliser un débogueur : exécuter un programme pas à pas, points d'arrêts, inspecter les variables pendant l'exécution.
- Prévoir les tests unitaires.

Travaux pratiques
Utilisation d'un débogueur pour contrôler l'exécution des programmes.