

Objectifs

Acquérir les fondements de la programmation et de l'algorithmique, 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.

Participants

Toute personne devant apprendre à programmer.

Prérequis

Aucune connaissance particulière.

Pédagogie

La pédagogie est basée sur le principe de la dynamique de groupe avec alternance d'apports théoriques, de phases de réflexion collectives et individuelles, d'exercices, d'études de cas et de mises en situations observées. Formation / Action participative et interactive : les participants sont acteurs de leur formation notamment lors des mises en situation car ils s'appuient sur leurs connaissances, les expériences et mettront en oeuvre les nouveaux outils présentés au cours de la session.

Remarque

Programmation sous java

Profil de l'intervenant

Consultant-formateur expert sur cette thématique. Suivi des compétences techniques et pédagogiques assurée par nos services.

Moyens techniques

Encadrement complet des stagiaires durant la formation. Espace d'accueil, configuration technique des salles et matériel pédagogique dédié pour les formations en centre. Remise d'une documentation pédagogique papier ou numérique à échéance de la formation.

Méthodes d'évaluation des acquis

Exercices individuels et collectifs durant la formation. Evaluation des acquis et attestation de fin de stage adressés avec la facture.

Programme

Un programme

Qu'est-ce qu'un programme ?

Qu'est-ce qu'un langage ? Les différents paradigmes.

Quel langage pour quelle application ?

Les compilateurs. Les exécutable.

Les responsabilités d'un programmeur.

Nécessité d'un algorithme

Qu'est-ce qu'un algorithme ?

Les besoins auxquels répond un algorithme.

Le concept de pseudo-langage.

Genèse d'un premier programme

Écriture d'un programme simple : syntaxe et instructions.

Compilation et exécution du programme.

Qu'est-ce qu'une librairie ? Son rôle, son usage.

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...

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.

Opérateurs et expressions

Les différents opérateurs (multiplicatif, additif, comparaison, égalité, logique, affectation).

Combinaison d'opérateurs.

Expression booléenne.

Les structures de contrôle

Les sélections alternatives (si, si-alors-sinon, sélection cas).

Les blocs d'instructions (notion de Début... Fin).

Les boucles itératives (tant que-répéter, répéter-jusqu'à, pour-de-à).

Imbrication des instructions.

Les commentaires.

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.

Introduction à la programmation



3 jours - 21 heures

Code formation : LGDV240

www.adhara.fr

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.

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.

Maintenance, débogage et test des programmes

Savoir lire et interpréter les différents messages d'erreurs.

Utiliser un débogueur : exécution d'un programme pas à pas, points d'arrêts, inspecter les variables pendant l'exécution.

Prévoir les tests unitaires.