

# FORMATION

## Programmation informatique (initiation)



## Objectifs

Apprendre les grands principes du développement informatique, de la programmation procédurale à la programmation objet. Réaliser son premier programme.

De nombreux cas pratiques sont traités, de la conception à la programmation sur ordinateur.

À l'issue de cette formation, les participants disposeront des bases solides leur permettant de choisir le langage de programmation le plus adapté à leurs projets et pourront commencer à développer dans ce langage.

Formation éligible au CPF - Code : 237556



► **Type de cours :**  
Stage pratique

► **Référence :**  
PROG1

► **Durée:**  
3 jours - 21h de formation

► **Lieu :**  
Paris ou en intra-entreprise

**ATTESTATION DELIVRÉE  
EN FIN DE FORMATION**

### Pré-requis

Pas de pré-requis nécessaire.

### Public

Toute personne souhaitant démarrer et évoluer en programmation informatique.

## Méthodes pédagogiques

- Vidéoprojecteur,
- Internet,
- Travaux pratiques sur ordinateur, encadrés par le formateur,
- Supports de cours,
- Évaluation en fin de stage.

# Programme du stage

## I- LA CONCEPTION D' UN PROGRAMME INFORMATIQUE

QU'EST-CE QU'UN PROGRAMME INFORMATIQUE ? .

Pourquoi et quand écrire un programme informatique ?

Illustration simple et concrète

Traduire la pensée en langage machine

Langages compilés et langages interprétés

L'algorithmique : qu'est ce que c'est ? Exemple concret d'algorithmes simples

Formaliser les étapes d'un projet informatique.

Le traduire en un programme.

### EXERCICES PRATIQUES

## II- PROGRAMMATION STRUCTURÉE ET PROGRAMMATION ORIENTÉE OBJET

Les objets. Concepts de classe, héritage, polymorphisme. L' encapsulation , intérêt.

Quel langage choisir pour développer ? Langage procédural, objet, événementiel ?

## III- ÉCRIRE SON PROGRAMME. MANIPULER LES DONNÉES DU PROGRAMME

Variables . Déclaration des variables

Types de données

Structures de programmation.

Structures conditionnelles

Opérateurs de la programmation

Tableaux

Structures de test

Structures de répétition. Traitements par boucles

### EXERCICES PRATIQUES

Création d'un programme simple

Écrire son premier script

## **IV : MODULARISER LE CODE INFORMATIQUE**

Notion de sousprogrammes : les fonctions et les procédures

Les paramètres - Gestion des paramètres

Partager le code source

Trouver et utiliser les bibliothèques d'un langage

Savoir mettre à profit le travail déjà effectué

### **EXERCICES PRATIQUES**

## **V- BONNES PRATIQUES EN PROGRAMMATION**

L'importance de bien écrire un programme.

Savoir présenter le code pour plus de clarté et compréhension,

L'importance des commentaires. Les conventions de nommage

L'aide en ligne

Les sources d'information nécessaires

### **EXERCICES PRATIQUES**

## **VI - GESTION DES DONNÉES À PARTIR D' UN FICHER**

Les étapes de manipulation d'un fichier. (Création, remplissage, interrogation )

Le format CSV. Similarité du modèle d'entrée/sortie standard

## **VII - GESTION DES DONNÉES EXTERNES - LES SGBD**

Qu' est ce qu'un SGDB ( Système de Gestion de Bases de données) ?

Comment communiquer avec un SGBD ?

Les API de base de données

## **VIII - LE LANGAGE DES SGBD**

La norme SQL

Les verbes du langage.

L'interrogation des données. La commande SELECT

### **EXERCICES PRATIQUES**



## **IX - LES ENVIRONNEMENTS DE DÉVELOPPEMENT. INTERFACE DE DEVELOPPEMENT (IDE)**

De l'éditeur de texte simple à l'IDE

Un outil pour un langage

L'éditeur pour corriger son code

Notion de BUG.

Présentation d'un débogueur

Mise en oeuvre des techniques de débogage

Points d'arrets pour connaitre la valeur des variables

### **NOMBREUX TRAVAUX PRATIQUES**

## **TEST DE VALIDATION DES COMPETENCES ACQUISES ATTESTATION DE FORMATION DELIVREE EN FIN DE STAGE**