

Introduction à la Programmation Orientée Objet

Programme (Mis à jour le 12/02/2021)

Introduction

- Historique des langages de programmation
- Les différents paradigmes de programmation
- La programmation procédurale
- La programmation orientée objet Comparatifs entre les langages objets Java, C# (.NET) et C++

Limitations à la programmation procédurale

- Limitations à la programmation procédurale
- Gestion de la modularité
- L'intégrité des données manipulées
- La réutilisation de code
- Risques relatifs à la duplication des codes et aux codes morts

Qu'est-ce que la programmation orientée objet ?

- Quelques points de terminologie : classe, objet, attribut, ...
- Nécessité de visualisation de la structure des code
- Le formalisme UML et la notion de diagramme de classe
- La notion de classes : une autre manière de voir la modularité
- Le principe d'encapsulation et pourquoi protéger les données
- Le concept d'héritage

Mise en œuvre de l'encapsulation

- Les niveaux de visibilité public et private
- La protection des membres
- Définition de propriétés (getter/setter)
- Définition de constructeurs
- Définition de méthodes de traitement
- Formalisme UML associé à l'encapsulation

Mise en association d'objets

- Association, composition et agrégation
- Formalisme UML associé aux mises en association
- Gestions des collections d'objets

Mise en œuvre de l'héritage

- Concept de polymorphisme
- Formalisme UML associé à l'héritage
- La notion de méthodes abstraites, de classes abstraites et d'interfaces

Utilisation de design patterns dans la conception d'une application

- Un design pattern, une solution éprouvée pour répondre à une problématique récurrente en programmation informatique
- Classification des principaux design patterns (patternsstephane GoF)
- Design patterns et UML

Référence

THIL2001

Durée

3 jours / 21 heures

Prix HT / stagiaire

1875€

Objectifs pédagogiques

- Comparer la programmation orientée objet et la programmation procédurale
- Utiliser les bonnes pratiques du développement objet
- Manipuler des schémas UML

Niveau requis

- Développer avec un langage de programmation procédurale.

Public concerné

- Développeurs.
- Chefs de projets informatique.

Formateur

Les formateurs intervenants pour Themanis sont qualifiés par notre Responsable Technique Olivier Astre pour les formations informatiques et bureautiques et par Didier Payen pour les formations management.

Conditions d'accès à la formation

Délai : 3 mois à 1 semaine avant le démarrage de la formation dans la limite des effectifs indiqués

Moyens pédagogiques et techniques

Salles de formation (les personnes en situation de handicap peuvent avoir des besoins spécifiques pour suivre la formation. N'hésitez pas à nous contacter pour en discuter) équipée d'un ordinateur de dernière génération par stagiaire, réseau haut débit et vidéo-projection UHD

Documents supports de formation projetés Apports théoriques, étude de cas concrets et exercices

Mise à disposition en ligne de documents supports à la suite de la formation

Dispositif de suivi de l'exécution de l'évaluation des résultats de la formation

Feuilles d'émargement (signature électronique privilégiée)

Evaluations formatives et des acquis sous forme de questions orales et/ou écrites (QCM) et/ou mises en situation

Questionnaires de satisfaction (enquête électronique privilégiée)