

Sommaire

1. Concepts de base de la POO

Définition de la programmation orientée objet

Classe, objet, attributs, méthodes

Encapsulation, abstraction

2. Principes fondamentaux : héritage, polymorphisme, etc.

Héritage classes de base classes dérivées

Polymorphisme redéfinition, surcharge, liaison dynamique

Interfaces, classes abstraites

3. Programmation en UML modélisation

Diagrammes de classes

Relations entre classes association, agrégation, composition, dépendance

Modélisation des objets systèmes

4. Application dans plusieurs langages exemples & syntaxe

Implémentation en Java

Implémentation en C#

Implémentation en C++

Implémentation en Python

Implémentation en PHP

et potentiellement d'autres, selon édition

5. Techniques avancées & bonnes pratiques

Gestion de la mémoire selon langage

Exceptions gestion des erreurs

Patterns architecture orientée objet

Principes SOLID développement robuste

6. Exercices et travaux pratiques

Exercices corrigés ou guidés

Études de cas concrets

Projets d'application : concevoir une petite application objet, modéliser en UML, coder, tester

7. Comparaisons portabilité entre langages

Similitudes et différences entre Java, C#, C++, Python, PHP

Avantages et contraintes de chaque langage pour la POO

Conseils pour choisir le bon langage selon le contexte/projet

8. Annexes Ressources supplémentaires