

Licence Informatique

COO et POO

Responsable	Descriptions	Informations
	Code : BIN5V3A	Composante : Faculté d'Économie et de Gestion
	Nature :	Nombre de crédits :
	Domaines : Sciences et Technologies	

LANGUE(S) D'ENSEIGNEMENT

Français

CONTENU

Objectifs de l'enseignement :

L'objectif de ce module est d'acquérir la maîtrise de concepts avancés de la programmation en Java permettant de concevoir des applications de bonne qualité. Plusieurs notions sensibles de la syntaxe Java sont étudiées au regard des principes de bonne programmation tels que compréhensibilité, maintenabilité, évolutivité, SOLID, cohérence avec l'API standard : visibilité, composition, agrégation, délégation, abstraction, interface, immuabilité, classe interne, classe anonyme, égalité d'objets, relation d'ordre entre objets, itérabilité. L'utilisation conjointe d'UML pour modéliser la structure des programmes est un soutien à l'activité de conception. Enfin, des outils du JDK sont expérimentés pour aider au développement.

Plan du cours :

1. Conception par objets, principes de POO, mise en œuvre avec UML et Java
2. Gestion de projets Java et documentation avec javadoc
3. Composition, agrégation, classe interne
4. Immuabilité
5. Niveaux de visibilité
6. Abstraction, interface
7. Itérations, comparaison d'objets, duplications d'objets
8. Classes utilitaires de base : Object, System, Scanner, String, StringBuilder, classes d'emballage des types primitifs, ArrayList, Objects, Arrays.

COMPÉTENCES À ACQUÉRIR

À l'issue de ce module, l'étudiant doit être apte à :

- expliquer tous les principes de programmation objet et les subtilités du langage Java permettant de produire des programmes valides, fiables, maintenables et évolutifs
- représenter la structure d'un programme en objets à l'aide de la notation graphique UML ;
- transformer une spécification avec les concepts orientés objets ;
- utiliser une méthode de type objet ;
- utiliser un ou plusieurs outils logiciels pour la mise en œuvre des méthodes ;
- justifier la qualité d'expression d'un modèle ;
- implanter un modèle objet.

MODALITÉS D'ORGANISATION

L'ensemble du cours est illustré par des exercices pratiques sur ordinateur.

BIBLIOGRAPHIE, LECTURES RECOMMANDÉES

- "Programmer en Java", C. Delannoy, Eyrolles, 2009.
- "Le langage Java", I. Charon, Hermès, 2007.
- "Java en concentré : Manuel de référence pour Java", 5e édition, David Flanagan, O'Reilly, 2006.
- "UML 2.0, guide de référence", James Rumbaugh, Ivar Jacobson, Grady Booch, Campus Press, 2005
- "UML 2 .0", Benoît Charoux, Aomar Osmani, Yan Thierry-Mieg, Pearson, 2008
- "Clean Code - A Handbook of Agile Software Craftsmanship", Robert C. Martin, Prentice Hall, 2009.

- "Refactoring - Improving the Design of Existing Code", Martin Fowler, Addison-Wesley Professional, 1999.

PRÉ-REQUIS OBLIGATOIRES

Des connaissances sur les fondements de la programmation orientée objet et leur mise en œuvre dans un langage orienté objet (Java de préférence) sont indispensables. Les concepts d'encapsulation, d'héritage et de polymorphisme doivent être maîtrisés. De plus, des bases en algorithmique et structures de données sont nécessaires.

VOLUME HORAIRE

- Volume total: 30 heures
- Cours magistraux: 15 heures
- Travaux dirigés: 15 heures

CODES APOGÉE

- BIN517A [ELP]

M3C

Aucune donnée M3C trouvée

POUR PLUS D'INFORMATIONS

[Aller sur le site de l'offre de formation...](#)



Dernière modification le 01/06/2023