

# Programmation orientée objet / Object oriented programming

 Composante  
Polytech  
Grenoble - INP,  
UGA

- > **Langue(s) d'enseignement:** Français
- > **Ouvert aux étudiants en échange:** Oui
- > **Code d'export Apogée:** KARI5M12

## Présentation

### Description

Ce cours revient sur la programmation impérative et ses concepts pour faire le contraste avec la programmation orientée-objet. Pour cela, nous utilisons le langage de programmation Java qui permet de faire à la fois de la programmation impérative et orientée-objet. Nous abordons les concepts du langage Java mais aussi les techniques et les outils de programmation. Les connaissances et l'expérience pratique acquises seront utiles en développement logiciel au delà du seul langage de programmation Java, quel que soit le langage et les outils utilisés.

Il est organisé en "sprints", avec plusieurs tâches par sprint. Le cours suit une progression en trois phases, guidant l'étudiant vers une meilleure maîtrise du développement logiciel.

1. La première phase est sur la programmation impérative et les outils d'aide à la programmation. Le but est d'établir une fondation commune aux étudiants venant d'horizons divers.
2. La seconde phase est la découverte de la programmation orientée-objet, au travers des concepts cœurs d'objets et de classes. Le focus est sur le graphe d'objets et l'exécution orientée-objet.
3. La troisième phase est l'introduction du polymorphisme et de l'héritage, découvrant la puissance de la programmation orientée-objet et des outils associés.

This course revisits imperative programming in order to contrast it with object-oriented programming. To that end, we use the Java programming language since it is a language that permits both imperative and object-oriented programming. We will discuss the programming concepts of the language but also coding techniques and tools. The skills and know-how acquired will be useful in software development beyond the only Java programming language, regardless of the language and tools used.

It is organized in "sprints", with several tasks per sprint. The course follows a progression in three incremental phases, taking all students on a journey towards mastering software development better.

1. The first phase is about imperative programming and programming tools, setting up a common foundation across students coming from backgrounds.
2. The second phase is about discovering object-oriented programming, understanding the core concepts of objects and classes. The focus is on the object graph and object-oriented execution.
3. The third phase is about introducing polymorphism and inheritance, discovering the power of object-oriented programming and tools.

---

## Objectifs

---

## Heures d'enseignement

Programmation orientée objet / Object oriented programming - CMTD

Cours magistral - Travaux dirigés

44h

---

## Pré-requis recommandés

Aucun

None.

**Période :** Semestre 5

Évaluation initiale / Session principale - Épreuves

Libellé	Nature de l'enseignement	Type d'évaluation	Nature de l'épreuve	Durée (en minutes)	Nombre d'épreuves	Coefficient de l'épreuve	Remarques
						60/100	

---

## Bibliographie

- B.Meyer, Object Oriented Software Construction, Second Édition. Prentice Hall, 1997
- G.Masini, A.Napoli, D.Colnet, D.Léonard, K.Tombre, Les Langages à objets, InterÉditions 1989
- Bruce Eckel, Thinking in Java, 4th Édition Prentice-Hall PTR, 2006.

## Infos pratiques

---

### Lieu(x) ville

- > Grenoble

---

### Campus

- > Grenoble - Saint-Martin d'Hères