

UE Bases de données avancées



Niveau d'étude
Bac +4



ECTS
3 crédits



Crédits ECTS
Echange
3.0



Composante
UFR IM2AG
(informatique,
mathématiques
et
mathématiques
appliquées)



Période de
l'année
Automne (sept.
à dec./janv.)

- > **Date de début des cours:** 8 sept. 2016
- > **Langue(s) d'enseignement:** Français
- > **Méthodes d'enseignement:** En présence
- > **Forme d'enseignement :** Cours magistral
- > **Ouvert aux étudiants en échange:** Oui
- > **Crédits ECTS Echange:** 3.0
- > **Code d'export Apogée:** GBIE7U11

Présentation

Description

Cette UE forme les étudiants à la configuration et l'optimisation d'un système de gestion de bases de données afin de concilier cohérence des données et efficacité d'accès aux données.

Les différentes propriétés d'un SGBD traditionnel sont étudiées et mises en pratique au sein d'une architecture logicielle multi-tiers et/ou micro-services.

Ces propriétés sont aussi analysées dans un cadre de système de gestion de données No-SQL afin d'identifier la meilleure solution en fonction de cas d'utilisation.

Le cours repose sur un enseignement théorique et pratique basé sur différents cas d'utilisation.

Heures d'enseignement

| | | |
|----|----|-----|
| TD | TD | 9h |
| CM | CM | 18h |
| TP | TP | 9h |

Pré-requis recommandés

- Concevoir un modèle de données UML et relationnel.
- Concevoir et normaliser une base de donnée relationnelle.
- Implémenter une base de donnée relationnelle en SQL.
- Transcrire un besoin fonctionnel en requêtes relationnelles algébriques et SQL
- Manipuler une base de données via un langage de haut niveau en suivant une architecture client-serveur 3-tiers
- Maitriser le langage Java

Période : Semestre 7

Compétences visées

- Analyser les exigences d'une application sur un plan transactionnel et d'optimisations de requêtes.
 - Implémenter une stratégie transactionnelle en considérant la cohérence des données et l'efficacité d'accès aux données.
 - Configurer un SGBD et optimiser un modèle de données pour garantir un plan d'exécution optimal des requêtes essentielles à un système d'information.
 - Identifier le type d'architecture de gestion de données en adéquation avec les besoins d'un système de gestion de données (SQL vs. NO-SQL, API vs. GraphQL).
 - Mettre en œuvre des architectures multi-tiers et micro-services d'accès aux données.
-

Bibliographie

- Bases de données et systèmes relationnels de Claude Delobel, Michel Adiba
 - Database System concepts de Henry F. Korth, Abraham Silberschartz
 - Cours de bases de données - Aspects systèmes de Philippe Rigaux
- <https://www.baeldung.com/category/spring/spring-boot>

Infos pratiques

Contacts

Responsable pédagogique

Fabrice Jouanot

✉ Fabrice.Jouanot@univ-grenoble-alpes.fr

Lieu(x) ville

> Grenoble

Campus

> Grenoble - Domaine universitaire