UCLouvain

linfo2172

2021

Databases

6.00 credits	30.0 h + 30.0 h	Q2

Teacher(s)	Nijssen Siegfried ;
Language :	English
Place of the course	Louvain-la-Neuve
Main themes	• Data Base Management Systems (objectives, requirements, architecture). • The Relational data model (formal theory, first-order logic, constraints). • Conceptual models (entity-relationship, object role modeling). • Logical database design (normal forms & normalization, ER-To-Relational) • Physical database design and storage (tables and keys, indexes, file structures). • Querying databases (Relational Algebra, Relational Calculus, data structures, query optimization, SQL) • ACID properties (Atomicity, Consistency, Isolation, Durability), Concurrency Control, Recovery techniques. • Programming database applications (JDBC, Database Cursors, Object-Relational Mapping). • Recent or more advanced trends in the database field (object-oriented databases, Big Data, NoSQL, NewSQL)
Learning outcomes	**At the end of this learning unit, the student is able to :** Given the learning outcomes of the "Master in Computer Science and Engineering" program, this course contributes to the development, acquisition and evaluation of the following learning outcomes: • INFO1.1-3 • INFO2.1-4 • INFO4.1-4 • INFO5.1-5 • INFO6.1, INFO6.4 Given the learning outcomes of the "Master [120] in Computer Science" program, this course contributes to the development, acquisition and evaluation of the following learning outcomes: • SINF1.M2 • SINF2.1-4 • SINF4.1-4 • SINF5.1-5 • SINF6.1, SINF6.4 Given the learning outcomes of the "Master [60] in Computer Science" program, this course contributes to the development, acquisition and evaluation of the following learning outcomes: • 1SINF1.M2 1 • 1SINF2.1-4 • 1SINF4.1-4 • 1SINF5.1-5 • 1SINF6.1, 1SINF6.4 Students completing this course successfully will be able to : • explain the scenarios in which using a database is more convenient than programming with data files; • explain the characteristics of the database approach, where they come from and contrast them with current trends in the database field • identify and describe the main functions of a database management system; • categorize conceptual, logical and physical data models based on the concepts they provide to describe the database structure; • understand the main principles and mathematical theory of the relational approach to database management; • design databases using a systematic approach, from a conceptual model through a logical level (i.e., a relational schema) into a physical model (i.e., tables and indexes); • Use SQL (DDL) to implement a relational database schema and distinguish from SQL facilities with respect to the logical vs. physical distinction. • query relational databases using SQL (DML) and contrast SQL with relational theory . • optimize the performance of databases. • understand the benefits and drawbacks of NoSQL databases. • use relational databases either directly or from a conventional programming language.

Evaluation methods	The final grade is determined by 4 projects and an exam that is organized at the end of the semester. The grade is calculated following a 75% / 25% rule (final written exam / participation and grade obtained for projects during the semester). Every project counts equally.
Teaching methods	The objectives are organized along three main axes: • Understand: both the historical context, and recent challenges and developments in the database field; relational theory, why is has been invented and how it fits in practice; implementation techniques and major algorithms for data organization, query and transaction processing. • Design: from conceptual modeling (e.g. Entity-Relationship, UML) down to physical database tuning (e.g. indexes, query plans), through logical database design (e.g. functional dependencies, normal forms, normalization algorithms) and reasoning (relational algebra, views and constraints). • Use: installing and configuring database management systems, creating and tuning databases, using query languages in practice (e.g. SQL), connecting to databases (e.g. call interfaces, ORMs), integrating database systems in software designs. Theory and practice are acquired by students along those three axes as follows: • Theory is dispensed in the traditional way, through lectures during the second quarter. The theoretical course follows Elmasri & Navathe's textbook [EN10]. • Practice is obtained by participating in 4 projects. These projects are either done individually, in groups of 2 or in larger groups. • Both theory and pratical missions are dispensed in English. Even though preference will be given to face-to-face teaching sessions, depending on the health situation and the number of students enrolled, other forms of teaching (online, co-modal or hybrid) may be considered.
Content	• Introduction to the entity-relationship model, • Bases of the relational model: data structures and algebra, • Logic-based relational languages to define and manipulate data, • Critical study of the SQL language, • Query optimization, • Database application programming, • Functions and architecture of database management systems, • Management of concurrent database accesses and associated techniques of recovery after failures, • NoSQL databases: graph databases, key-value stores, document stores, • Overview of other databases: spatio-temporal databases, data warehouses, OLAP
Inline resources	https://moodle.uclouvain.be/course/view.php?id=733
Bibliography	• Ramez Elmasri and Shamkant Navathe Fundamentals of Database Systems. Addison-Wesley Publishing Company, USA, 7th edition, 2010. • Chris J. Date. An Introduction to Database Systems. Pearson Addison-Wesley, Boston, MA, 8 edition, 2004. • Hugh Darwen, An Introduction To Relational Database Theory, 3th Edition, Bookboon, 2009 • Jean-Luc Hainaut, Bases de Données, Concepts, Utilisation et Développement, 2e Edition, Dunod, 2012 • T.M. Connolly and C.E. Begg. Database Systems: A Practical Approach to Design, Implementation, and Management. Number v. 1 in International computer science series. Addison-Wesley, 2005.
Other infos	This course does not have strict prerequisites. Nevertheless, prior knowledge of the following subjects is highly recommended: • SQL • diagrammes d'objets • structures de données the course will take significantly more effort to follow if you don't have this prior knowledge. Examples of courses that offer these topics are: • LSINF1225 or LINFO1225 : Basic knowledge of database management, • LSINF1121 or LINFO1121 : good abilities in programming.
Faculty or entity in charge	INFO

Programmes containing this learning unit (UE)

Program title	Acronym	Credits	Prerequisite	Learning outcomes
Master [120] in Agricultural Bioengineering	BIRA2M	6		
Master [120] in Data Science Engineering	DATE2M	6		
Master [120] in Environmental Bioengineering	BIRE2M	5		
Master [120] in Computer Science and Engineering	INFO2M	6		
Master [120] in Data Science: Information Technology	DATI2M	6		
Master [120] in Biomedical Engineering	GBIO2M	6		
Master [60] in Computer Science	SINF2M1	6		
Master [120] in Forests and Natural Areas Engineering	BIRF2M	6		
Master [120] in Computer Science	SINF2M	6		
Master [120] in Mathematical Engineering	MAP2M	6		
Master [120] in Data Science : Statistic	DATS2M	6		