

Advanced Management of Data

Exam topics

Exam topics

- Three-Level Architecture, Data Independence
- Conceptual Data Modeling (EER-Model using UML)
 - Practical Exercise: Creation of an ER model for a given task
- Concepts of the relational model
 - relation, attribute, domain, primary key, foreign key, integrity constraints, views, null
 - Practical Exercise: Deriving relations for a given ER model
 - Relational Algebra: Operations selection, projection, cartesian product, Join
 - SQL: Realize (and understand) simple queries involving one or more tables

Exam topics

- Extensions of SQL
 - What led to the introduction of PL/SQL, and SQL/PSM?
 - What are cursors, exceptions, stored procedures (no syntactic details)?
 - How is a procedure different from a function?
 - What are triggers?
 - Discuss the differences between BEFORE, AFTER, INSTEAD OF, ROW-level Statement-level triggers? Give examples to illustrate your answers
 - Discuss the advantages and disadvantages of database triggers
 - Practical Exercise: You don't need to program SQL extensions, but may be able to describe the results of given code examples

Exam topics

- Discuss the general characteristics of advanced database applications
- Discuss why the weaknesses of the relational data model and relational DBMSs may make them unsuitable for advanced database applications
- What is the Non First Normal Form?
- What are Nest, Unnest, and intersection join operations?
- Discuss some extensions of ORDBMS to overcome limitations of RDBMS, Name Advantages & Disadvantages of ORDBMS

(you don't need to know syntactic details, but know existing concepts)

Exam topics

- What is a distributed Database?
- What are the main reasons for and potential advantages of distributed databases?
- What additional functions does a DDBMS have over a centralized DBMS?
- Which types of Distributed DBS exist?
- Discuss concepts of Distributed DB (Availability, Reliability, Scalability, Partition Tolerance, Autonomy, Replication, Allocation, Transparency)
- Discuss the architecture of a DDBMS. Within the context of a centralized DBMS, briefly explain new components introduced by the distribution of data.

Exam topics

- What is a fragment of a relation? What are the main types of fragments? Why is fragmentation a useful concept in distributed database design?
- Why is data replication useful in DDBMSs? What typical units of data are replicated?
- What is meant by data allocation in distributed database design? What typical units of data are distributed over sites?
- How is a horizontal partitioning of a relation specified? How can a relation be put back together from a complete horizontal partitioning?
- How is a vertical partitioning of a relation specified? How can a relation be put back together from a complete vertical partitioning?

(You should have basic knowledge of SQL, in order to derive fragments from a table)

Exam topics

- Discuss application of traditional relational systems and NOSQL systems
- CAP Theorem, Eventual consistency, Sharding, Schemaless data, Semistructured Data (motivation, representation)
- Categories of NOSQL Systems
 - some essential properties / concepts of these systems (by example)
- What is consistent Hashing?

(again, no syntactic details have to be known of presented NOSQL systems)