1. **What database models do you know?**

* Hierarchical (tree)
* Network / graph
* Relational (table)
* Object-oriented

1. **Which are the main functions performed by a Relational Database Management System (RDBMS)?**

* Represent a bunch of tables together with the relationships between them.
* Rely on a strong mathematical foundation: the relational algebra.
* Relational Database Management Systems (RDBMS) manage data stored in tables.

1. **Define what is "table" in database terms.**

Database tables consist of data, arranged in rows and columns. All rows have the same structure. Columns have name and type (number, string, date, image, or other).

1. **Explain the difference between a primary and a foreign key.**

* Primary key is a column of the table that uniquely identifies its rows (usually it is a number). Two records (rows) are different if and only if their primary keys are different. The primary key can be composed by several columns (composite primary key).
* The foreign key is an identifier of a record located in another table (usually it’s primary key).By using relationships we avoid repeating data in the database.

1. **Explain the different kinds of relationships between tables in relational databases.**

Relationships have multiplicity:

* One-to-many – e.g. country / towns (A single record in the first table has many corresponding records in the second table)
* Many-to-many – e.g. student / course (Records in the first table have many corresponding records in the second one and vice versa)
* One-to-one – e.g. example human / student (A single record in a table corresponds to a single record in the other table)

1. **When is a certain database schema normalized?**

* Normalization of the relational schema removes repeating data.
* Non-normalized schemas can contain many data repetitions

1. **What are database integrity constraints and when are they used?**

* Integrity constraints ensure data integrity in the database tables. Enforce data rules which cannot be violated.
* Primary key constraint ensures that the primary key of a table has unique value for each table row.
* Unique key constraint ensures that all values in a certain column (or a group of columns) are unique.
* Foreign key constraint ensures that the value in given column is a key from another table.
* Check constraint ensures that values in a certain column meet some predefined condition.

1. **Point out the pros and cons of using indexes in a database.**

Indices speed up searching of values in a certain column or group of columns usually implemented as B-trees. Indices can be built-in the table (clustered) or stored externally (non-clustered). Adding and deleting records in indexed tables is slower! Indices should be used for big tables only.

1. **What's the main purpose of the SQL language?**

Structured Query Language (SQL) is a special-purpose programming language designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream management system (RDSMS).

1. **What are transactions used for?**

Transactions guarantee the consistency and the integrity of the database. Transactions are a sequence of database operations which are executed as a single unit:

* Either all of them execute successfully
* Or none of them is executed at all

**Example:**

* A bank transfer from one account into another (withdrawal + deposit).
* If either the withdrawal or the deposit fails the entire operation should be cancelled.

1. **What is a NoSQL database?**

NoSQL (non-relational) databases use document-based model (non-relational) , schema-free document storage. Still support CRUD operations (Create, Read, Update, Delete). Still support indexing and querying, still supports concurrency and transactions. They are highly optimized for append / retrieve. They gave great performance and scalability.

1. **Explain the classical non-relational data models.**

Relational databases:

* Data stored as table rows
* Relationships between related rows
* Single entity spans multiple tables
* RDBMS systems are very mature, rock solid

NoSQL databases:

* Data stored as documents
* Single entity (document) is a single record
* Documents do not have a fixed structure

1. **Give few examples of NoSQL databases and their pros and cons.**

**Redis** - Ultra-fast in-memory data structures server

**MongoDB** - Mature and powerful JSON-document database

**CouchDB** - JSON-based document database with REST API

**Cassandra** - Distributed wide-column database