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

Relational and NoSQL.

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

Data definition (create, drop etc.), Data management (insert, update etc.), SQL language support and optionally Transaction management support.

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

Table contains data in structured form. Table columns define the “properties” or the types of data that a single entry can have. Rows represent different entries each having the same base structure as all others in that table.

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

The primary key is unique identifier for each row of a given table. Foreign key in a given table represents the primary key of another table, providing means to connect the data between the two tables.

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

The relationship between tables can be one-to-many where single row in one table can have multiple corresponding entries in the other table.

Many-to-many relationship means that entries from both tables can have multiple corresponding records. This is achieved with intermediary table which keeps record of the relationships.

One-to-one relationship is used to represent inheritance between tables. It splits the data in two or more tables with each row in one table corresponding to a single row in the other table.

1. **When is a certain database schema normalized? What are the advantages of normalized databases?**

The database schema is normalized when one or more of the Normal Forms are achieved. Generally they define rules for the design of the database such as the repetitions of data are minimized. The more normalized the database is the less repeating data it contains. Usually this comes at the cost of performance.

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

Integrity constraints define rules that provide data integrity in the database by validating data stored in columns or even relationships between tables.

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

Indexes increase the search speed in specified columns. They are usually used in databases with large size to increase performance. On the other hand they usually slow down adding and deleting operations and are not recommended on tables with a lot insertions and deletions, for columns other than auto increment primary key.

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

The main purpose of SQL is to provide standardized language for declaring the general operations performed over a database.

1. **What are transactions used for? Give an example.**

Transactions are used to prevent simultaneous access to data that can result in undetermined state for this data. They also provide locking mechanism allowing for several operations to be carried out in sequence without the possibility for any other operation interfering with their execution. For example a transaction is bank account withdraw. If two withdraws happen simultaneously, the transaction will carry out for one of them with validation then the actual withdraw operation and update of the internal state. After this the other transaction will perform all those actions. This prevents circumventing some of the validation logic.

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

NoSQL database store data not in tabular and relational form but in some other form that doesn’t have predefined structure or schema.

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

Document – set of documents;

Key-value – pairs of keys and values;

Hierarchical key-values – same as above but stored in hierarchy

Wide-column – key-value pairs with schema

Object – set of oop-style object

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

Windows registry is example for hierarchical key-value pairs. Tweeter and Facebook use NoSQL data bases. MongoDB is the most widespread NoSQL DB.

The main pros of the NoSQL databases is the performance, high access speed and great scalability. The cons are they don’t have fixed structure and are less reliable than Relational databases, therefore are not good for sensitive data.