**Introduction to databases homework**

1. Hierarchical(tree), Network / graph, Relational (table), Object Oriented, Document models
2. RDBMS system can create, alter and delete tables and relationships between tables (database schema). They can add, search, change, delete and retrieve data from the data in the tables. The management system supports the SQL language.
3. Table consists of rows and columns. Columns specify the schema of the table – name of the colum and data type. Rows specify every entry in the database (the data) and should implement the schema (the columns).
4. Primary key is a column of the table (usually int) that identifies each row in it. Foreign key is a column that identifies a record (row) in another table (usually it’s primary key).
5. One-to-many – e.g. a firm can have many shops but a shop can be in only one firm. Implemented with one foreign key in the shop table.

Many-to-many – e.g. a course can have many students and a student can participate in many courses. Usually implemented through another database which contains two foreign keys about the course and student.

One-to-one – used when we want to have inheritance. E.g. human – student. The human would have some properties and the student will extend them. The primary key is a foreign key in the same time.

1. A database is normalized when there is no repetition of data in it. We normalize it by extracting repeated data into another table and making relation through a foreign key.

The advantage of a normalized database is that we have a smaller database and the queries are faster

1. Integrity constrain is a condition that a certain column should meet. For example the primary key should be unique in the current table.
2. Indexes speed up searching in the database however it slows down adding and removing elements. It should only be used in big databases.
3. Through SQL we can easily manipulate the database however and whenever we want. We can make scripts for creating a certain database or adding data to it, so we don’t have to do it by ourselves every time.
4. Transactions are a sequence of database operations which is executed as a single unit. That means that either every operation is performed or neither one of them is. It is used for security reasons. For example when we add something to the database and the internet stops we wouldn’t want to have something in half in the database we would prefer to have nothing and this is where transactions come in handy.
5. NoSQL database is a non-relational database. For example a document model database (like mongodb) where everything is stored in a file and we have no relations.
6. Document model - set of documents for example json strings.

Key-value model – set of key – value pairs

Hierarchical key-value - hierarchy of key – value pairs

Wide – column model – key – value with schema

Object model - set of OOP-style objects.

1. Pros – NoSQL databases are much more scalable then relational databases. They use less storage and can be runned on slower and cheaper servers. They are much more flexible – you can easily change your database structure while in production while with a relational database its much harder and you should consider carefully every step otherwise you can loose your whole data.

Cons – NoSQL databases are fairly new. Most of them don’t stand behind big companies so their administration is not the best. Most of them are also in pre-production period or beta versions with some key features still not implemented. Relational databases are much more mature and have all important functionalities and people tend to choose the more secure way.