1. Hierarchical, network, relational and object-oriented are the ones I know about.
2. The main functions performed by a RDBMS are managing data stored in tables, which consist of:

- creating, altering, deleting tables and relationships between them

- adding, changing, deleting, searching and retrieving of data stored in tables

- supporting SQL language

- transaction management, which is optional

1. Table consists of rows and columns, which are filled with data.
2. The foreign key can be a part of the data in another table, whereas the primary key can`t.
3. There are three kinds of relationships between tables:

* One-to-many – a specific movie(one) – containing a lot of genres(many) :

The Shawshank Redemption – crime/drama

* One-to-one – a company/firm – it`s name – there can NOT be two companies with the same name – one company responds to one name
* Many-to-many – meals – ingredients – each meal contains some type of ingredients, which are met in other meals too

1. A certain database is normalized when all the repeated information or data is removed from the current table and stored in another table. Then by using foreign key in the old table the value is connected to the primary key from the new table. The advantages are smaller databases(memory-wise) and better utility(for example if you want to change something you change the value in one table, instead of searching for every occurrence of the value you want to change).
2. Databases integrity constrains ensure data integrity in the data tables. Primary key constrains ensure the primary key has a unique value for each row. Unique key constrains ensure that the value added will not be repeated. Foreign key constrains ensure that there will not be foreign keys pointing to non-existent key. Check constrains ensure values fit a certain query/condition.
3. Using indexes helps you to search faster. The con of this is it makes deleting and adding records in indexed tables is slower.
4. The main purpose of the SQL language is to define and manipulate data.
5. Transactions are used for manipulating a certain database, while having the ability if the transaction fails to rollback any changes made. This keeps the integrity of the database. Example can be the money transfer from one bank account to another.
6. NoSQL database uses the document model. It has no schema, but still it supports most operations that relational model has. It has great performance and scalability.
7. **Wide-column stores** such as Cassandra and HBase are optimized for queries over large datasets, and store columns of data together, instead of rows.Document databases pair each key with a complex data structure known as a document. Documents can contain many different key-value pairs, or key-array pairs, or even nested documents.
8. Some of the pros are :

* Scaling
* Free or minimum cost versus the expensive RDBM systems
* Flexible data models

And some of the cons are :

* Support – most NoSQL systems are open source projects, which leaves you to the community and its experience
* Schemas – for NoSQL databases they are typically dynamic. For some databases (e.g., wide-column stores), it is somewhat more challenging to add new fields dynamically.
* Support Transactions - In certain circumstances and at certain levels (e.g., document level vs. database level)