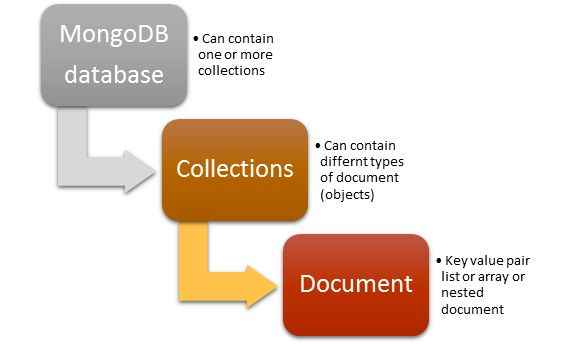
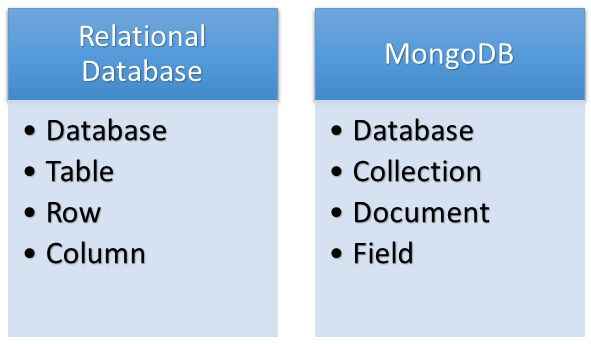
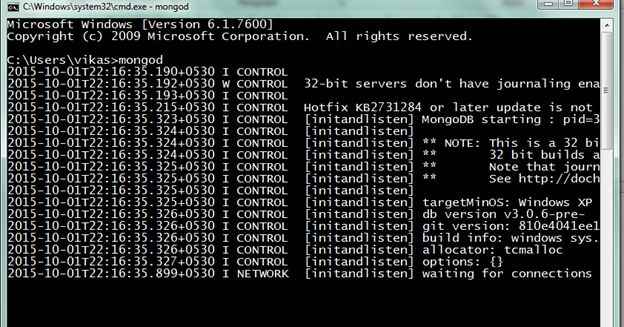
**What is MongoDB?**  
MongoDB is an open source document oriented database. MongoDB falls in the category of the NoSQL – Database which means it do not follow fixed schema structure like in relational databases.  
  
MongoDB cannot replace Relational databases but it should be viewed as an alternative to it.  
  
MongoDB can be installed on Windows, Linux and MAC, so it is a cross platform database. It do not support joins but it can represent rich, hierarchical data structures. And the best feature like the most is that it is easily scalable and can give high performance.   
  
The MongoDB database consists of a set of databases in which each database contains multiple **collections**. MongoDB is schema-less that means every collection can contain different types of object. Every object is also called **document** which is represented as a JSON (JavaScript Object Notation) structure: a list of key-value pairs. The value can be of three types: a primitive value, an array of documents or again a list of key-value pairs.   
  
  
  
Let’s see how RDBMS and MongoDB differ:  
  


**Document-Oriented Data-Model**  
But before we move into the next step of creating our own database let’s see what exactly document oriented data-model is:  
  
A typical document in MongoDB looks something like this:

1. {
2. \_id: ObjectID('4bd9e8e17cefd644108961bb'),
3. name: 'Vivek',
4. **class**: '12th',
5. subjects: ['physics', 'chemistry', 'math', 'english', 'computer'],
6. address: {
7. house\_no: '12B',
8. block: 'B',
9. sector: 12,
10. city: 'noida',
11. },
12. grade: [{
13. exam: 'unit test 1',
14. score: '60%'
15. }, {
16. exam: 'unit test 2',
17. score: '70%'
18. }
20. ]
21. }

Above document contains information of a student in the key-value pair. It contains unique \_id for the record, name and its value, class and its value, subjects and its value is in the form of array, address contains its value in the form of another in-document and grade contains its value in the form of arrays of documents.

If we have to represent the same record in Relational world then we would require at least three tables. One to store basic information like \_id, name, class, address and another to store subjects and another one to store grades, etc. But here we stored the whole relational information in one complete document this is how we managed the deficiency of joins and constraints in MongoDB. In MongoDB we do not have joins but it’s up to us the developers how we are designing our schema to manage relations.

**Mongo Server**  


**Mongo Client**



Commands

* **show dbs :**Will show the databases in your system.
* **show collections :**Will show the collections in a db.
* **db.help() :** Will show the help on db methods.
* **db.mycoll.help() :** Will show the help on collections methods.

