1. **Centralised User Name and Password For mongodb:**

Suppose, you want a centralised user name and password, (The API follows this concept)

Then, you have to do the following:

**mongo**

(to start the mongo shell)

**use admin**

(to use mongodb in admin mode)

**db.createUser(  
  {  
    user: "admin",  
    pwd: "admin123",  
    roles: [ { role: "root", db: "admin" } ]  
  }  
)**

**use serverApplicationDB**

(to create a Db named serverApplicationDB)

1. **Make Mongodb Listen To Remote Connection Request:**

Open /etc/mongod.conf using vim editor.

**# Listen to local interface only. Comment out to listen on all interfaces.**

**bind\_ip=127.0.0.1**

Comment out bind\_ip line.

So,

It must look like this:

**# Listen to local interface only. Comment out to listen on all interfaces.**

**#bind\_ip=127.0.0.1**

1. **To set unique keys (primary key type feature) for collection (table equivalent of database)**

**To create a collection:**

db.createCollection(“uniqueIDCard4”)

To make id field (say, uniqueIDCard4 will have a field named id) acts as primary key( So, that records with already existing id value cannot be inserted) we have to create a index

**Db.uniqueIDCard4.createIndex({id:1},{unique:true})**

id could be 1 or -1. (ascending or descending)

**Note:**

However, we cannot ensure that primary key’s value cannot be null.

A huge drawback in mongodb. We need additional checking mechanism before insertion.