* Theory:

• What is NOSQL

- NoSQL database is a non-relational database management system that does not require a fixed schema. It avoids joins & is easy to scale.
- The major purpose of using a NOSQL is for distributed data stores with homogeneous data storage needs.
- NOSQL is used for big data & realtime web apps for example companies like twitter, facebook & Google collect terabytes of user data every single day.

• What is MongoDB.

- MongoDB is document database with scalability & flexibility that you want with querying & indexing that you need.

• Differentiate bet^{n} NOSQL & RDBMS

NOSQL	RDBMS
- This is relatively new & expirts in NOSQL are less as this database is evolving day by day	- users know RDBMS well as it is old & many organizat^{n} use this database for proper format of data.

- NoSQL is non-relational database model used to store & retrieve data	- RDBMS is relational database management system based on relational database model.
- NoSQL database operate without predifined schema & they do not need to define database structure for storing & manging data.	- Relational DB req. a schema to enable storing of data & it should be defined before adding data to database.
- Nosql databases are non-relational database models that have more complex structure than tables.	- Relational DB models are table based meaning they store data in structured formed using rows & columns.
- NoSQL databases are able to scale on diff servers without having to join rows from multiple servers & Best suited for modern applications that rely on more complex data structure means unstructed data.	- Relational models aren't designed to be able to scale acro multiple machines across multiple data centers. Best suited for storing & querying structured relational data.

- To Comeout from error in MongoDB use ctrl+c
- To stop mongodb service use ctrl+z

- differentiate bet^n MySQL & MongoDB.

MySQL	MongoDB.
- MySQL represents data in tables & rows (schema)	- Mongodb represents data as JSON documents.
- MySQL requires you to define the tables & columns before you to store & every row in table must have same columns.	- In MongoDB you don't need to define the schema. Instead you just drop in documents & don't even need to have same fields.
- MySQL use structured query language (SQL)	- MongoDB uses javascript as query language.
- It is written in C & C++	- It is written in C, C++ & Java.
- once the schema is defined it cannot be changed, +	- It's schema can change which means it supports dynamic schema.
- MySQL uses table, row, column & joins	- MongoDB uses collection, document, field, embeded document, linking.

Assignment 10

Problem statement:
CRUD operations in MongoDB

Theory

- What is Collection -
A Collection is a grouping of MongoDB documents. Documents with a Collection can have different fields. A Collection is the equivalent of a table in a relational database System.

- What is documents & fields
In MongoDB, the data records are stored as BSON documents. Here, BSON stands for Binary representation of JSON documents, although BSON contains more data types as compared to JSON. The documents is created using field-value pairs or key value pairs & the value of field can be of any BSON type.
field names are string.
Doument's have the following restrictions on field name:
The field name-id is reserved for use as a primary key; it's value must be unique in the collection, is immutable & may be any type other than an array.

- Explain dynamic Schema Concept.
MongoDB supports dynamic Schemas.

In other words, we need not to define the schema of before the insertion of data. We can change the schema of the database dynamically.

- To create Correction Command is
  > db. Create Correction ("name of Correction");

- To insert documents in the Collection we use,
  > db. name Correction. insert ({"attribute1": "value1", "attribute": "value", "class": "TY"});

- To display documents property in a single line
  > db. name of Correction. find(). pretty()

- To update document -
  > db. name of Correction. updat ({"empid": 4}, {$set: {"empid": 10}})

- To remove the document from the Collection
  we use,

  > db. name of Correction. remove ({"rollno": 2