Title: MongoDB - Aggregation and Indexing

Problem Statement:
Design and develop MongoDB Queries using aggregation and indexing with suitable example using MongoDB

Objectives:
- Understand aggregation & indexing in MongoDB with suitable example.

SW and HW Requirements:
Windows 10 64bit, 8GB RAM,
MongoDB installed on machine.

Theory:

Aggregation:
Aggregation operation process data records and return computed result. Aggregation operations group values from multiple documents together, and can peform a variety of operations on the grouped data to return single result. In SQL count(* and with group is equivalent to MongoDB aggregation.

syntax:
db.collection_name.aggregate(AGGREGATE_OPERATION)

eg db.records.aggregate([{$group: {_id:"$rollno",
num_sources: {$sum: 1}}}])

this will return number of cources enrolled by each student.

Indexing:

Indexes support efficient resolution of queries. Without indexes, MongoDB must scan every document of a collection to select those documents that match the query statement. This scan is highly inefficient and require MongoDB to process a large volume of data.

Indexes are special data structures, that store a small portion of data of the set in easy-to-traverse form.

SYNTAX:

db.collection_name.createIndex({KEY:1})

- key is name of field on which you want to create Index and 1 is ascending order.
-1 for decending order.

queries:

db.records.createIndex({"rollno":1})

db.records.dropIndex({"rollno":1})

db.records.getIndexes()

conclusion:

Successfully designed & implemented aggregation & indexing techniques in MongoDB.