

# Laborationsrapport

## 3.1 / MongoDB

*DT162G, Javascriptbaserad utveckling*

Författare: Björn Olsson, [bjol1600@student.miun.se](mailto:bjol1600@student.miun.se)  
Termin 5, år: 3, 2018



**Mittuniversitetet**  
MID SWEDEN UNIVERSITY

Campus Härnösand Universitetsbacken 1, SE-871 88. Campus Sundsvall Holmgatan 10, SE-851 70 Sundsvall.  
Campus Östersund Kunskapens väg 8, SE-831 25 Östersund.  
Phone: +46 (0)771 97 50 00, Fax: +46 (0)771 97 50 01.

## Sammanfattning

Besvara ett antal frågor kopplat till moment 3.1 som handlar om en introduktion till MongoDB.

## Innehållsförteckning

<b>Sammanfattning.....</b>	<b>2</b>
<b>1    NoSQL.....</b>	<b>4</b>
1.1    Fråga 1.....	4
1.2    Fråga 2.....	4
1.3    Fråga 3.....	4
1.4    Fråga 4.....	4
1.4.1    MongoDB.....	5
1.4.2    Apache Ignite.....	5
1.4.3    Amazon DynamoDB.....	5
<b>2    MongoDB.....</b>	<b>7</b>
2.1    Fråga 5.....	7
2.2    Fråga 6.....	7
2.3    Fråga 7.....	7
2.4    Fråga 8.....	8
2.5    Fråga 9.....	8
2.6    Fråga 10.....	8

# 1 NoSQL

## 1.1 Fråga 1

**Förklara begreppet "NoSQL"**

Begreppet NoSQL syftar till "non SQL" eller att den inte använder sig av relationsdatabaser, och är avsedd för dokument som t.ex. JSON-format. Källa: <https://en.wikipedia.org/wiki/NoSQL>

## 1.2 Fråga 2

**Vad är fördelarna med en NoSQL-databas, jämfört med en relationsdatabas?**

NoSQL-databaser används mer och mer i s.k. "big-data" applikationer för att de kan lagra stora mängder data och anses vara enklare designade, enklare horisontell skalbarhet för maskiner vilket enligt <https://en.wikipedia.org/wiki/NoSQL> kan vara ett problem för relationsdatabaser, de anses även vara mer tillgängliga. Strukturerna som används i NoSQL skiljer sig från relationsdatabaser vilket gör dem snabbare i vissa avseenden. Källa: <https://en.wikipedia.org/wiki/NoSQL> De är designade för att vara mer ekonomiskt tillgängliga och därmed användas av lågkostnadsutrustning. <https://greengarageblog.org/7-pros-and-cons-of-nosql>

## 1.3 Fråga 3

**Kan du se några nackdelar?**

Enligt <https://greengarageblog.org/7-pros-and-cons-of-nosql> så har relationsdatabaser växt fram över ren vilket gör dem mycket mer stabila än NoSQL-databaser. Det tas även upp att företag som använder sig av det här vill ju ha tillgång till snabb och effektiv hjälp ifall något skulle gå åt skogen. Bristen på antal personer som kan NoSQL till den grad att de kan bidra med support är fortfarande hög.

## 1.4 Fråga 4

**Välj tre stycken NoSQL-databaser och ge en summerad redogörelse för var och en. Bör vara i storleksordningen 1 till max 2 stycken A4-sidor (inklusive eventuella illustrationer).**

#### 1.4.1 MongoDB



MongoDB är ett gratis dokumentorienterat databasprogram som är både gratis och tillgängligt på flertalet plattformar. MongoDB klassas som ett NoSQL-program och använder sig av documenttyper som liknar JSON med "schemata". MongoDB började utvecklas 2007 som en delkomponent i ett större system. Över tid förändrades företagets inställning till det man utvecklade och till slut blev det ett fristående system där man 2013 bytte namn på företaget till MongoDB Inc.

Exempel på funktioner MongoDB stödjer är exempelvis s.k. "Ad hoc"-queries vilket innebär att sökningar gjorda på fält, omfång/spann och regular expressions. De queries man använder sig av kan även returnera Javascriptfunktioner. Indexering som görs i MongoDB kan vara både primärindexerade och sekundärindexerade. För närvarande finns MongoDB i en Community-version och en Enterprise-version beroende på om man är företag eller ej och släpps under Server Side Public License där den senaste stabila versionen är 4.0.4 som släpptes den 8:e November 2018. (Källa: <https://en.wikipedia.org/wiki/MongoDB>)

#### 1.4.2 Apache Ignite



Apache Ignite är en databas med öppen källkod som är designad för att lagra och beräkna stora mängder data. Ignite-databaser använder sig av RAM som standard och klassas som "in-memory computing"-system. Data lagras i key-value pairs och databaskomponenten scale-ar horisontellt genom att man distribuerar paren över alla noder så att det blir jämnt fördelat, data balanseras därmed också om varje gång en nod läggs till eller tas bort från klustret.

Apache Ignite är skrivet i Java, C# och C++ och dess senaste release är version 2.6.0. och utvecklare av Ignite är Apache Software Foundation. (Källa: [https://en.wikipedia.org/wiki/Apache\\_Ignite](https://en.wikipedia.org/wiki/Apache_Ignite) )

#### 1.4.3 Amazon DynamoDB



DynamoDB är en databas som kommer från Amazon och är på framfart i världen där den bl.a. används av det Svenska företaget KRY (Källa: TV4-nyheterna, 2018-11-20). DynamoDB är en NoSQL-databas som använder sig av key-value pair-lagring i document-datastrukturer, och ingår i Amazon Web Services. DynamoDB släpptes första gången 2012 och är en multiplatforms-lösning som scale-ar automatiskt om användaren så önskar, då sprids datan och trafiken ut över ett antal solid-state hårddiskar vilket tydligen ger tillgång till stabil prestation från serverna. DynamoDB finns tillgänglig för flertalet språk så som: Java, Node.js, C#.NET, PHP och Python.

(Källa: [https://en.wikipedia.org/wiki/Amazon\\_DynamoDB](https://en.wikipedia.org/wiki/Amazon_DynamoDB) )

## 2 MongoDB

### 2.1 Fråga 5

**I MongoDB, om man inte specifikt anger ett värde för "\_id"-fältnamnet kommer ett sådant att skapas automatiskt. Hur är detta uppbyggt för att minska risken att två likadana id'n skapas?**

Som jag förstår det som står på MongoDB:s egen hemsida så skapas ett automatiskt unikt ObjectId innan insättningen i databasen sker. Om man sätter in ett id själv så måste det vara unikt annars kommer det skapa "duplicate key error".

(Källa: <https://docs.mongodb.com/manual/reference/method/db.collection.insert/>)

### 2.2 Fråga 6

**Vilka datatyper finns tillgängliga i MongoDB? Vilka tror du är de som du kommer använda flitigast?**

Det finns 16 st olika datatyper i MongoDB som: "Double, String, Object, Array, Binary data, Undefined, Object Id, Boolean, Date, Null, Regular Expression, Javascript, symbol, Javascript with scope, integer, timestamp, min key och max key. Och de vanligaste borde väl vara string, object, array, integer och boolean skulle jag anta. (Källa: <https://data-flair.training/blogs/mongodb-data-types/> )

### 2.3 Fråga 7

**Namnge och beskriv kortfattat de olika användarroller som finns tillgängliga i MongoDB.**

Som jag förstår det finns det inbyggda "user roles" och "administration roles" och har man "admin"-databasen så förekommer ännu fler.

I "User roles" ingår 2 roller som heter "read" och "readWrite", den förstnämnda ger möjlighet att läsa alla "non-system collections" medans "readWrite" ger möjlighet att både läsa och modifiera samma data i systems.js-collectionen.

I "Administration roles" ingår "dbAdmin", "dbOwner" och "userAdmin". "dbAdmin" ger möjlighet att utföra de administrativa uppgifter man vill såsom ändra indexering, samla in statistik men man kan inte hantera roll- och användare. "dbOwner" är en sammanslagning, kan man säga, av readWrite, dbAdmin och userAdmin och kan utföra vilka administrativa uppgifter som helst. "userAdmin" ger möjlighet att ändra roller och användare i den nuvarande databasen, denna roll kan även tilldela superuser vilket ger möjlighet att utöka privilegierna till "admin database".

(Källa: <https://docs.mongodb.com/manual/core/security-built-in-roles/> )

## 2.4 Fråga 8

**Anta att man använder följande kommando för att lägga in data i en databas:**

```
use companyAB
db.createCollection("companyRecords")
col1 = db.getCollection("companyRecords")
col1.insert({ name: "Jan Jansson", email:
"jan.janssons@jansson.se", phone: "555-129234" })
col1.insert({ name: "Sven Svensson", email:
"sven.svensson@svensson.se", phone: "555-987651" })
```

**Hur skriver man om man vill plocka fram eller skriva ut enbart det dokument som har telefonnummer 555-129234?**

```
db.companyRecords.find({phone: "555-129234"})
```

(Källa: <https://docs.mongodb.com/manual/tutorial/project-fields-from-query-results/> )

## 2.5 Fråga 9

**Hur skriver man för att uppdatera telefonnumret för Sven Svensson till 555-123456, utan att ändra på övriga uppgifter?**

```
db.companyRecords.updateOne({ "name": "Sven Svensson"},
{ $set: { "phone": "555-123456" } });
```

(Källa: <https://docs.mongodb.com/manual/reference/method/db.collection.updateOne/#db.collection.updateOne> )

## 2.6 Fråga 10

**Hur skriver man om man vill exportera ut den collection ovan som heter *companyRecords* från databasen *companyAB* i JSON-format till en extern fil vid namn *records.json*?**

```
Mongoexport --db companyAB --collection companyRe-
cords --out records.json
```

(Källa: <https://docs.mongodb.com/manual/reference/program/mongoexport/> )



