

Multi-Model DBs (ArangoDB) oppgaver

Bakgrunnsinfo

- Følg dokumentet for oppsett og import til ArangoDB og sett opp en lokal versjon av ArangoDB:
<http://www.uio.no/studier/emner/matnat/ifi/INF3100/v17/undervisningsmateriale/arangodb-setup.pdf>
- For dokumentasjon på AQL (spørrespråket) se her:
<https://docs.arangodb.com/3.1/AQL/Fundamentals/>

Document-store oppgaver (spøringer)

(bruk collection: **buildings_csv**)

1. Finn alle bygninger med fare for oversvømmelse (FLOOD) og returner koordinatene for disse byggene, samt risikoen for oversvømmelse.

HINT: «FLOOD» er definert med tall.

2. Finn alle bygninger med fare for storm (STORM) og returner koordinatene for disse byggene, samt risikoen for storm.

HINT: «STORM» er definert med tall.

Document-store oppgaver (spøringer)

(bruk collection: **buildings_csv**)

3. Kombiner de to første oppgavene for å finne faren for storm og oversvømmelse for alle bygg. Dersom det er fare for oversvømmelse er øker vi risikoen med 1. Dersom faren for storm er mer en 0, øker vi risikoen med 1.

Returner koordinatene for bygningene og risikoen.

HINT: Bruk «LET» for å lage variabler.

4. Beregn risikoen for oversvømmelse per sted. Dette beregnes med å se på antall bygg som er utsatt for oversvømmelse, delt på totalt antall bygg på dette stedet og ganget med 100:

$$\text{EIF} = (\text{AntByggiFare} / \text{TotAntallBygg}) * 100$$

Returner Stedsnavn (KOMM_NAVN), stedskode (KOMM) og EIF (Kalkulert risiko)

HINT: Bruk «collect» for å gruppere på sted, «Aggregate» for å summere opp og «LET» for å kalkulere EIF. I dokumentasjonen finner du dette under «High level Operations» og «collect».

Graf oppgaver (spørringer) – tips

- Når vi går gjennom grafer kan vi si hvilke retning vi ønsker å sjekke. Dette er «Outbound» og «Inbound».
- Når vi går gjennom nodene med FOR definerer vi hvor mange steg vi ønsker å gå på denne måten:
1..4
Dette betyr at vi vil sjekke fra første node etter startnoden og frem til det fjerde steget bort fra start noden.
- Syntaksen for å gå gjennom en graf er som dette:
FOR node (kan hete det du vil) IN [1..4] OUTBOUND [node id] GRAPH [navn på grafen]
FILTER osv.
- Se dokumentasjon for graf spørringer her: <https://docs.arangodb.com/3.1/AQL/Graphs/>

Graf oppgaver (spørringer) – del 1

Med utgangspunkt i oppgavene for Neo4J

(<http://www.uio.no/studier/emner/matnat/ifi/INF3100/v17/undervisningsmateriale/graph-dbs---neo4j.pdf>),

gjør det samme i ArangoDB:

- Opprett en ny collection «Nodes» og en edge-collection «node_edges»
- Du kan enten opprette nodene og kantene gjennom admin panelet, eller du kan lage 2 json filer med dataen i. Dersom du lager json filer må du passe på at du kun har ett json objekt per linje. I tillegg må hvert objekt ha en variabel «_key» som er unik og forteller hvilke objekt det er.

Så student 1 blir for eks:

```
{_key: «s1», studentID: "1", lastName: "Doe", firstName: "Ana", middleName: «Maria"}
```

Deretter kan du følge import oppskriften i dokumentet for oppsett og import til ArangoDB

5. Gjør de samme oppgavene som du gjorde for Neo4J

(<http://www.uio.no/studier/emner/matnat/ifi/INF3100/v17/undervisningsmateriale/graph-dbs---neo4j.pdf>)

Graf oppgaver (spørringer) – del 2

Dette er de samme dataene som i document-store delen av oppgaven. Gjør nå de samme oppgavene (spørringer 1-3), men med en graf representasjon.

Lag en ny graf under «GRAPHS» i admin panelet. Du må gi den et navn, samt fortelle hvilke «edge_definition» som gjelder (building_edge) og hvilke «from/to collection» som gjelder (buildings)

6. Finn alle bygninger med fare for oversvømmelse (FLOOD) og returner koordinatene for disse byggene, samt risikoen for oversvømmelse.
HINT: «FLOOD» er definert med tall.
7. Finn alle bygninger med fare for storm (STORM) og returner koordinatene for disse byggene, samt risikoen for storm.
HINT: «STORM» er definert med tall.
8. Kombiner de to første oppgavene for å finne faren for storm og oversvømmelse for alle bygg. Dersom det er fare for oversvømmelse er dette +1 i risiko. Dersom faren for storm er mer en 0, øker vi risikoen med 1.
Returner koordinatene for bygningene og risikoen.
HINT: Bruk «LET» for å lage variabler.

Graf oppgaver (spørringer) – del 2 – hints

- Et par hint:
 - Hver node som er en bygning har denne verdien «PREFIX: Building»
 - Noder med kolonne verdier har denne verdien «COLUMN: [kolonne navn]» og verdien ligger lagret i denne noden med verdien «value: [*verdien*]»
 - I alle tilfellene under ønsker vi å sjekke «OUTBOUND»
 - WKT (Posisjonen for bygget) har verdien: «literalValue: WKT» og selve punktet ligger i «value»