Lab 3 Column-oriented Databases

Objetivos

Os objetivos deste trabalho são:

- Compreender os fundamentos das bases de dados em que as tabelas são armazenadas por colunas.
- Instalar e utilizar uma solução de código aberto.
- Desenvolver soluções para diversos casos de uso.

Nota prévia

Este módulo deverá ser preferencialmente desenvolvido em Linux. Caso pretenda usar Windows verifique as notas sobre compatibilidade do software que irá usar.
Submeta o código/resultados/relatórios no elearning. Utilize uma pasta (1, 2, ..5) para cada exercício, compactadas num único ficheiro.
Bom trabalho!

3.1 Cassandra – Instalação e exploração por linha de comandos

Cassandra é uma *column-oriented database* inicialmente desenvolvida pelo Facebook e atualmente suportada pela "Apache Software Foundation". É um projeto de código aberto, com licença "Apache License 2.0".

a) Instale a Apache Cassandra no seu computador pessoal (http://cassandra.apache.org/) e execute o servidor (`cassandra -f`).

b) Estude o funcionamento do sistema testando os comandos mais usados, através de linha de comandos (programa cliente `cqlsh`).

   Consulte os slides disponibilizados para a disciplina e sítios web com documentação sobre Cassandra. Alguns exemplos:

   - *http://cassandra.apache.org/doc/latest/*
   - *https://www.tutorialspoint.com/cassandra/*

   Deve estudar conceitos e funcionalidades tais como:

   - Criação, descrição e utilização de Keyspace
   - Criação e descrição de Tabelas
   - Escrita, Leitura, Edição, Remoção (CRUD)
   - Column Values - utilização de *nested tuples*, *collections, etc.*
   - Time-to-live e Timestamp

c) Produza um relatório (`CBD_L301_<NMEC>.TXT`) com todas as iterações com o `cqlsh`. Comente algumas das operações.

3.2 Cassandra – Sistema de Partilha de Vídeos

Pretende-se implementar uma base de dados que suporte um sistema de partilha de vídeos utilizando a linguagem CQL (Cassandra Query Language).

a) Desenvolva e implemente o modelo de dados do sistema de partilha de vídeos tendo em atenção as especificidades da base de dados Cassandra e respeitando os seguintes requisitos:

1. *Gestão de utilizadores: entre outros, registar o username, nome, email, selo temporal de registo na plataforma;*
2. *Gestão de vídeos: entre outros, registar o autor da partilha, nome do vídeo, descrição, uma ou mais etiquetas (tag) e selo temporal de upload/partilha;*
3. *Gestão de comentários: realizados para um vídeo em determinado momento temporal e tem como autor um utilizador;*
4. *Gestão de vídeo followers: permitir o registo de utilizadores que seguem determinado vídeo. Num sistema de informação, permitirá por exemplo notificar todos os followers de um novo comentário inserido para o vídeo;*
5. *Registo de eventos: por vídeo e utilizador e podem ser do tipo play/pause/stop, incluindo o registo temporal do evento e o momento (temporal) em que ocorre no vídeo. Por exemplo, o utilizador XPTO fez play no vídeo YGZ às 2:35:54 do dia 2 de outubro de 2017, ao 300 segundo de tempo do vídeo;*
6. *Rating de vídeos: valor inteiro de 1-5, por vídeo e não necessita de registo do autor.*
7. *Permitir a pesquisa de todos os vídeos de determinado autor;*
8. *Permitir a pesquisa de comentários por utilizador, ordenado inversamente pela data;*
9. *Permitir a pesquisa de comentários por vídeos, ordenado inversamente pela data;*
10. *Permitir a pesquisa do rating médio de um vídeo e quantas vezes foi votado;*

b) Introduza dados em todas as tabelas criadas (10-30 inserções por entidade). O conteúdo das mesmas deve fazer sentido, nomeadamente garantir que temos dados que permitam responder às diversas pesquisas solicitadas. De seguida, e para cada tabela, efetue uma query que retorne todos os registos em formato JSON. Deve submeter a script com os CQL DML *statements* criados, assim com o resultado de cada query num ficheiro ".json".

c) Se ainda não o fez, implemente as pesquisas definidas da alínea *7* a *10*;

d) Implemente as seguintes pesquisas.
<u>Notas importantes</u>*: a) Deve evitar utilizar o "allow filtering" nas consultas implementadas; b) Algumas alíneas/queries não são possíveis de realizar em Cassandra; identifique-as e justifique, baseando-se nos conceitos do modelo de dados de Cassandra.*

1. *Os últimos 3 comentários introduzidos para um vídeo;*
2. *Lista das tags de determinado vídeo;*

3. *Todos os vídeos com a tag Aveiro;*
4. *Os últimos 5 eventos de determinado vídeo realizados por um utilizador;*
5. *Vídeos partilhados por determinado utilizador (maria1987, por exemplo) num determinado período de tempo (Agosto de 2017, por exemplo);*
6. *Os últimos 10 vídeos, ordenado inversamente pela data da partilhada;*
7. *Todos os seguidores (followers) de determinado vídeo;*
8. *Todos os comentários (dos vídeos) que determinado utilizador está a seguir (following);*
9. *Os 5 vídeos com maior rating;*
10. *Uma query que retorne todos os vídeos e que mostre claramente a forma pela qual estão ordenados;*
11. *Lista com as Tags existentes e o número de vídeos catalogados com cada uma delas;*
12. .. 13. *Descreva 2 perguntas adicionais à base dados (alíneas 12 a 13), significativamente distintas das anteriores, e apresente igualmente a solução de pesquisa para cada questão.*

3.3 Cassandra – Driver (Java)

Para este exercício deverá utilizar a base de dados do sistema de partilha de vídeos, mas agora acedendo programaticamente através de um driver JAVA. Para instalação do driver no seu ambiente de trabalho (Java IDE), deve seguir as recomendações disponíveis. Por exemplo, em:

https://www.tutorialspoint.com/cassandra/cassandra_installation.htm
https://docs.datastax.com/en/driver-matrix/doc/driver_matrix/javaDrivers.html
https://github.com/datastax/java-driver

a) Desenvolva uma pequena aplicação Java que demonstre a inserção, edição e pesquisa de registos na base de dados.

b) Reimplemente em Java, quatro *queries* à sua escolha do exercício 3.2.

3.4 Base de Dados com Temática Livre

Este exercício tem como objetivo o desenvolvimento de uma pequena base de dados que tire partido do modelo de dados de Cassandra e cuja temática é livre. No entanto, deve respeitar os seguintes requisitos:

a) Um keyspace com, pelo menos, 4 tabelas;

b) Inserção de uma média de 12 registos por tabela;

c) Utilização das seguintes estruturas de dados (todas):

- *Set, list, map*

d) Definição de, pelo menos, 2 índices secundários;

e) Utilização de, pelo menos, 5 updates e 5 deletes de dados:

- *utilize operações não triviais sobre as estruturas de dados (da alínea c))*

f) Criação de 10 queries expressivas do seu domínio de conhecimento da cláusula

SELECT:

- *use WHERE, ORDER BY, LIMIT, etc.*

3.5 Cassandra – Restaurant Database (Opcional)

Tenha como referência a base de dados de restaurantes do guião Lab2.

a) Construa uma réplica dessa base de dados utilizando agora Cassandra. O modelo de dados deve ter em consideração as especificidades deste tipo da base de dados.

b) Carregue a base de dados com o conteúdo do ficheiro "restaurantes.json".

c) Implemente as consultas 1-5, 7, 8, 12, 14, 15, 17, 19, 27e 30 do exercício 2.2, utilizando o set de instruções DML (Data Management Language) de CQL. O resultado das consultas deve ser apresentado/guardado em formato JSON. Note que algumas destas consultas poderão requerer um processamento complementar do lado do cliente.