MA-DS5 NoSQL Databases & Cloud Computing (Winter Term 2021/22)
Dr. Vinu Venugopal & M.Sc. Mauro D. L. Tosi & M.Sc. Jingjing XU

Exercise sheet #3 Due on Dec. 16, 2021

For this exercise sheet, please consider the Yago dataset (`yago2.tsv`) available on the course homepage on `moodle.uni.lu`. The Yago dataset contains facts in the form of triples and each triple has three parts: *subject*, *predicate* and *object*. For example, '$\langle Barack\_Obama\rangle\langle isCitizenOf\rangle\langle United\_States\rangle$' is a fact in Yago. Here, *Barack_Obama* and *United_States* are the subject and the object respectively, and *isCitizenOf* is the predicate connecting the subject and the object.

Pig Latin & Hive

Processing YAGO dataset **8 Points**

Problem 1. Find the top three frequently occurring *predicates* in the Yago dataset using:

(i) operators available in the Pig Latin scripting language;
4 Points

(ii) operators available in HiveQL.
4 Points

Grouping and Joining **10 Points**

Problem 2. Identify all the given-names (corresponding to `hasGivenName` predicate) of persons who are associated with more than one `livesIn` predicates from the Yago dataset using:

(i) the relational operations (joins, grouping, etc.) available in the Pig Latin scripting language;
5 Points

(ii) the relational operations which are available in HiveQL.
5 Points

MongoDB

Aggregation Pipeline and Map-Reduce in MongoDB **12 Points**

Problem 3. Perform the following two queries using *Aggregation Pipeline* & *Map-Reduce* :

(i) all the subjects (`x`) and objects (`y` and `z`) matching the pattern: `?x <hasGivenName> ?y. ?x <livesIn> ?z.`, from the Yago dataset.

For example, if these are your inputs: { `<a> <hasGivenName> <b>, <c> <hasGivenName> <d>, <a> <livesIn> <Luxembourg>, <c> <isCitizenOf> <China>` }. Then, the expect output is: `(<a>, <b>, <Luxembourg>)`
4 Points

(ii) the family name of all the persons who are citizens of more than two countries. You may use the predicates: `<isCitizenOf> and <hasFamilyName>`, for mining the pattern.
4 Points

Consider again the `yago2.tsv` for this problem. For case (i), you may use `$lookup` along with other pipeline operators to construct the query. Please refer to `Ex3_P3_hint.txt` file on the moodle for hints. For case (ii), you may follow the Reduce-Side join example illustrated in the lecture to perform the join operation.

- Compare the runtime obtained in both these cases (i.e, Problem-3 (i) and (ii)). **2 Point**
- Consider creating indexes on various fields of the collection and observe how the runtime changes. **2 Point**