NIE-PDB | **Advanced Database Systems** | 2024/25 Winter

Assignment **03** – **MapReduce**

Assignment

- Create an **input text file** with sample data from the domain of your individual topic
  - Insert realistic and non-trivial data about at least **10 entities of one type**
  - Put each of these entities on a separate line
    - I.e., assume that **each line of the input file yields one input record**
  - Organize the actual entity attributes in whatever way you are able to easily parse
  - E.g.: `Medvídek 2007 53 100 Trojan Macháček Vilhelmová`
    - Which is supposed to correspond to a pattern `Movie Year Rating Length Actors ...`
- Implement a non-trivial **MapReduce job**
  - Choose from aggregation, grouping, filtering or any other general MapReduce usage pattern
  - Use `WordCount.java` source file as a basis for your own implementation
  - Both the `Map` and `Reduce` functions should be non-trivial, each about 10 lines of code
  - It is not necessary to implement the `Combine` function
- Comment the source file and also **provide a description of the problem** you are solving
- You may also create a shell script that allows for the execution of your entire MapReduce job
  - I.e., compile source files, deploy input file, execute the actual job, retrieve its result, ...
  - However, this script is not supposed to be submitted and serves just for your own convenience
  - Even if you do so, it will not be used for the purpose of homework assessment in any way

Requirements

- You may split your MapReduce job implementation into multiple **Java source files**
  - They all must be located in the submission root directory
  - At least `MapReduce.java` source file with its public `MapReduce` class is required
    - Do not forget that file names in general are expected to correspond to class names
  - This class is expected to represent the main class of the entire MapReduce job
- Do not change the way how **command line arguments** are processed
  - I.e., the only two arguments represent the input and output HDFS locations respectively
- Do not use packages in order to organize your Java source files
- Assume that only the following two libraries will be linked with your project
  - `hadoop-common-3.3.4.jar` and `hadoop-mapreduce-client-core-3.3.4.jar`
- Do not submit your Netbeans (or any other) project directory or Hadoop (or any other) libraries
- Use Java Standard Edition version 7 or newer
- You are free to use your `/user/pdb241_login/` **HDFS home directory** for debugging
  - Homework assessment, however, will take place in a different dedicated HDFS directory

Submission

- **readme.txt**: description of the input data structure and objective of the MapReduce job
- **input.txt**: text file with your sample input data (i.e., only one input file is permitted)
- **MapReduce.java** and possibly other ***.java**: Java source files with your MapReduce implementation
- **output.txt**: expected output of your MapReduce job
  - I.e., submit the result of the execution you performed by yourself

Tools

- **Apache Hadoop** (3.3.4) – https://hadoop.apache.org/
  - Already installed on the NoSQL server

References

- **Hadoop File System Shell Commands**
  - https://hadoop.apache.org/docs/r3.3.4/hadoop-project-dist/hadoop-common/FileSystemShell.html
- **MapReduce Tutorial**
  - https://hadoop.apache.org/docs/r3.3.4/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html
- **MapReduce Commands Guide**
  - https://hadoop.apache.org/docs/r3.3.4/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapredCommands.html
- **Hadoop JavaDoc API Documentation**
  - https://hadoop.apache.org/docs/r3.3.4/api/