Latihan Hadoop, Pig & Hive

Mata Kuliah: Big Data dan Data Lakehouse (A)

Dosen Pengampu: Fuad Dary Rosyadi, S.Kom., M.Kom.

Nama	NRP
Nabiel Nizar Anwari	5027231087

Excercise Hadoop, PIG, and Hive menggunakan dataset movielens 100k

Instalasi Hadoop + Tools Disini

Soal 1: Apache Hadoop (HDFS)

Prerequisite

```
wget https://files.grouplens.org/datasets/movielens/ml-100k.zip
unzip ml-100k.zip
```

1. Buat direktori movielens di HDFS.

```
hdfs dfs -mkdir /movielens
```

2. Upload file u.data ke direktori tersebut.

```
hdfs dfs -put /root/ml-100k/u.data /movielens/
```

3. Tampilkan 10 baris pertama dari file.

```
hdfs dfs -cat /movielens/u.data | head -n 10
```

4. Hitung ukuran file di HDFS.

```
hdfs dfs -ls -h /movielens/u.data
```

image1 image2 image3

Soal 2: Apache Pig

1. Load file u.data ke Pig.

```
pig

raw_data = LOAD '/movielens/u.data' USING PigStorage('\t')
           AS (user_id:int, item_id:int, rating:int, timestamp:long);
```

2. Hitung rata-rata rating per item_id (film).

```
grouped = GROUP raw_data BY item_id;

avg_rating = FOREACH grouped GENERATE
             group AS item_id,
             AVG(raw_data.rating) AS avg_rating;
```

3. Ambil hanya film yang memiliki rating rata-rata ≥ 4.0.

```
fav_movies = FILTER avg_rating BY avg_rating >= 4.0;
```

4. Simpan hasil akhir ke output/film_favorit.

```
STORE fav_movies INTO '/output/film_favorit' USING PigStorage('\t');
```

Menampilkan Hasil:

```
quit

hdfs dfs -cat /output/film_favorit/part-*
```

image1 image2

Soal 3: Apache Hive

1. Buat database movielens.

```
hive

CREATE DATABASE movielens;
USE movielens;
```

2. Buat tabel ratings (user_id INT, item_id INT, rating INT, timestamp BIGINT).

```
CREATE TABLE ratings (
  `user_id` INT,
  `item_id` INT,
  `rating` INT,
  `timestamp` BIGINT
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY '\t';
```

3. Load data dari file u.data.

```
LOAD DATA INPATH '/movielens/u.data' INTO TABLE ratings;
```

4. Hitung rata-rata rating setiap film.

```
SELECT item_id, AVG(rating) AS avg_rating
FROM ratings
GROUP BY item_id;
```

5. Ambil 10 film dengan rata-rata rating tertinggi.

```
SELECT item_id, AVG(rating) AS avg_rating
FROM ratings
GROUP BY item_id
ORDER BY avg_rating DESC
LIMIT 10;
```

image1 image 2 image 3 image 4 image 5

> Hamdalah