**HADOOP**

하둡은 배치성으로 데이터를 저장하고 처리하는데 적합한 시스템이다. 이러한 저장기능을 맞는 시스템이 HDFS이다. 하지만, 무결성 때문에 HADOOP 플랫폼에서 처리가 어려워 HIVE와 Spark 등 방식으로 처리한다.

**HIVE**

|  |  |
| --- | --- |
|  | 분산처리를 담당하는 것이 Mapreduce가 필요한데 Hive를 이용하면 엑셀처럼 sum을 사용할 수 있다. 즉, HIVE를 이용하면 Mapper와 Reducer 역할을 손쉽게 구현할 수 있다. |

|  |  |
| --- | --- |
|  |  |

* 하둡 파일 시스템 HDFS (Hadoop Distributed File System) 으로부터 데이터를 불러오고 저장
* Apache Hive는 Hadoop을 기반으로 동작하는 SQL Query Engine이라고 간단하게 요약, 즉 Apache Hive를 구성하는 모듈에서 Metadata store는 Hive QL에서 사용하는 데이터베이스 테이블과 HDFS 상의 파일을 서로 연결해주는 역할로 보통 연결하는 작업은 Hive QL로 처리하지만, 그 정보는 어딘가에 저장돼 있어야 하는데 그 저장소가 바로 Metadata store
* HDFS는 TEXTFILE형태로 되어 있고, HIVE는 METASTORE\_DB를 통해서 HDFS의 자료정보를 불러오고 SCHEMA를 통해서 DB-TABLE 형태로 만들어 MAP + REDUCER 작업을 한다.

**BEELINE**

* HiveServer2에 접속하여 command shell 을 수행할 수 있도록 도와주는 client

**ORC 파일**

* ORC 파일 포맷은 하이브의 처리 속도를 높이고 컬럼 기반으로 데이터를 저장

**[실습 1] hdoop계정의 로컬파일 employee.csv 데이터를 hive로 hdfs에 저장하기**

1. **Local에서 csv 파일 생성 및 저장**

파일내용

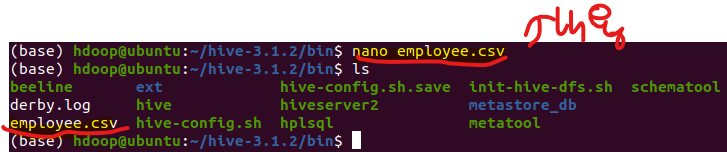
1201,Gopal,45000,Technical manager

1202,Manisha,45000,Proof reader

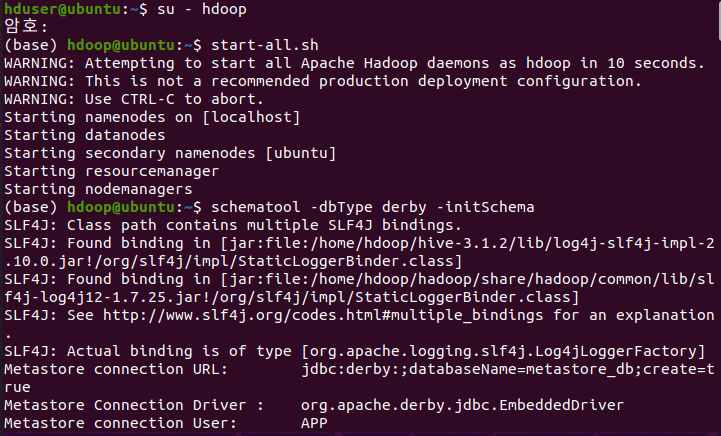
1203,Masthanvali,40000,Technical writer

1204,Kiran,40000,Hr Admin

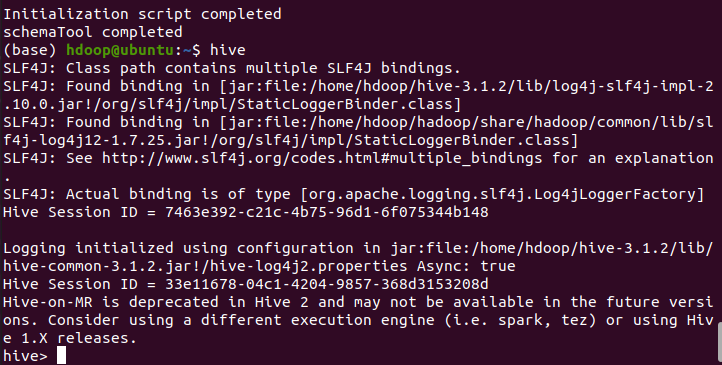
1205,Kranthi,30000,Op Admin



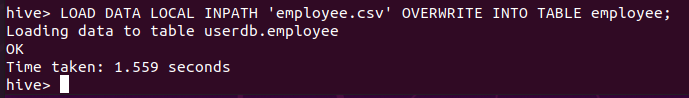
1. **HIVE 실행**



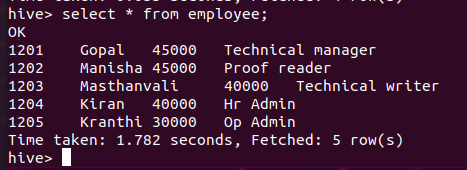
( -- 중략 -- )



1. **HIVE를 통해서 로커파일을 HDFS 파일에 저장**



1. **HIVE로 저장된 HDFS 파일 불러오기**

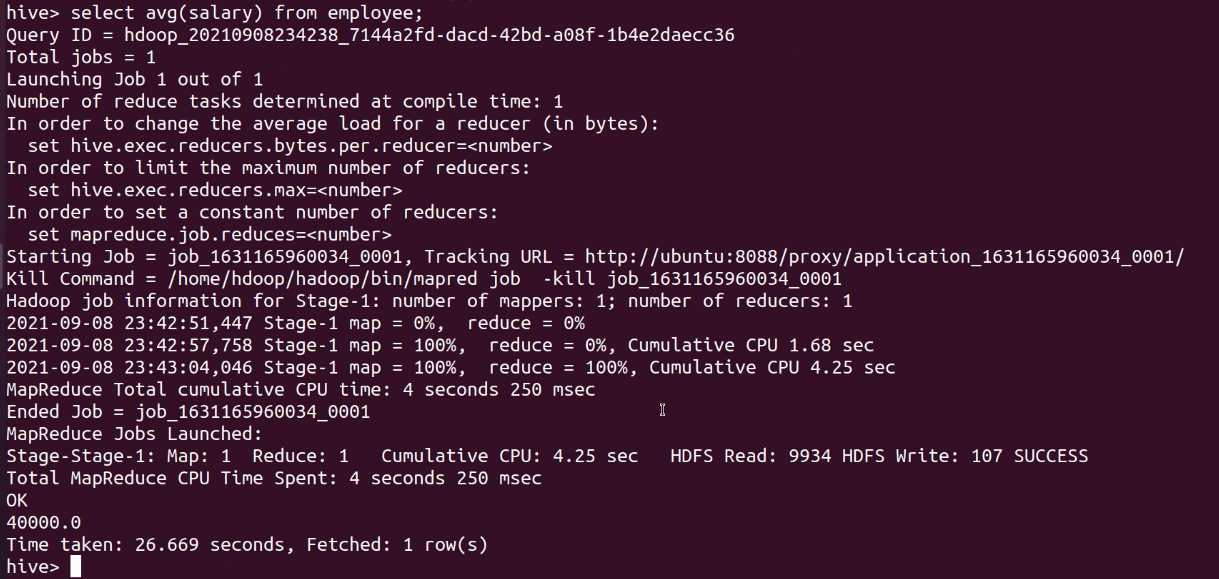


1. **리눅스 커널에서도 HDFS 파일 저장여부를 확인**

|  |
| --- |
|  |

**[2] HIVE의 주요기능 MAPREDUCER 작업을 수행**

**- HIVE QL을 통해서 맵리듀스(정렬분류) 작업을 하여 급여 평균 구히기**

****

**[3] 파이썬을 통해서 ORC 테이블 생성과 BEELINE으로 해당 테이블 확인하는 작업**

**- 크롤링 작업을 수행하고, 해당 파일을 TEXTFILE 테이블로 저장**

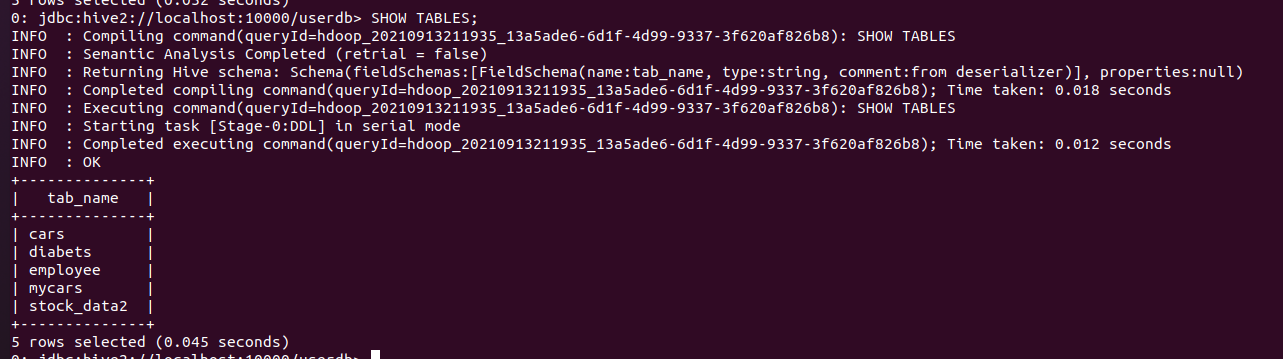
**- BEELINE에서 테이블을 확인하고 ORC TABLE을 생성**

**- TEXTFILE 테이블 내용을 ORC TABLE로 복사이동**

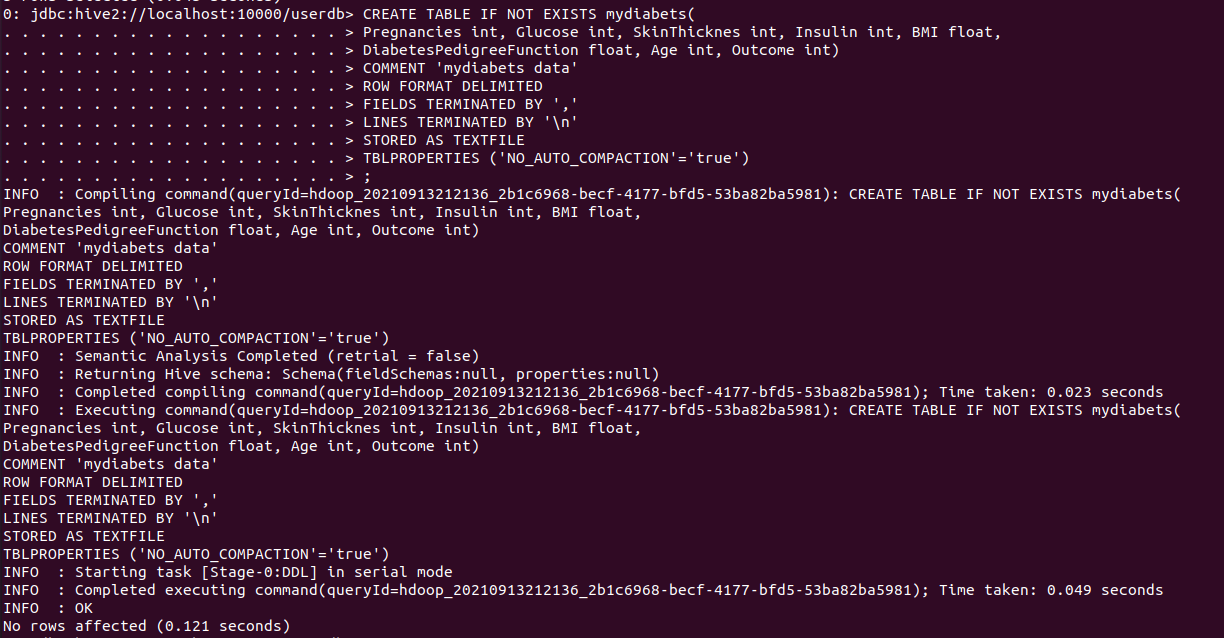
코드 실행 (크롤링 및 테이블형성 그리고 파일저장) **(codefile: hive-test-01.ipynb)**



EXTERNAL TABLE 생성



ORC TABLE 생성



EXTERNAL TABLE MOVE TO ORC TABLE

