



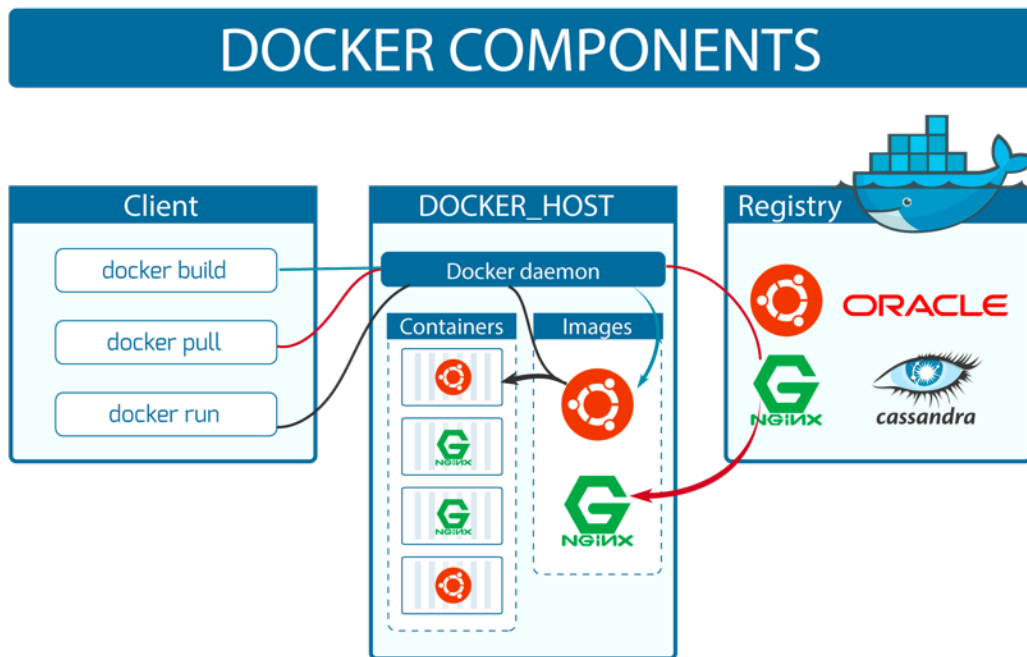
Подготовка рабочего места

Установка и настройка Hive

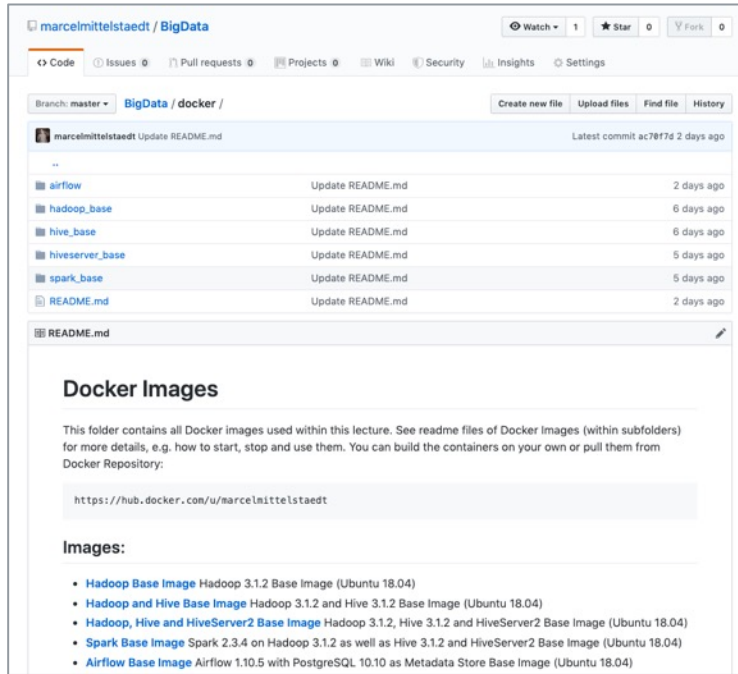
Вариант 4 (Практический экзамен)

Docker

Чтобы ускорить процесс и не тратить время на установку и настройку Hive и других инструментов, мы будем использовать контейнер docker, который уже подготовлен.

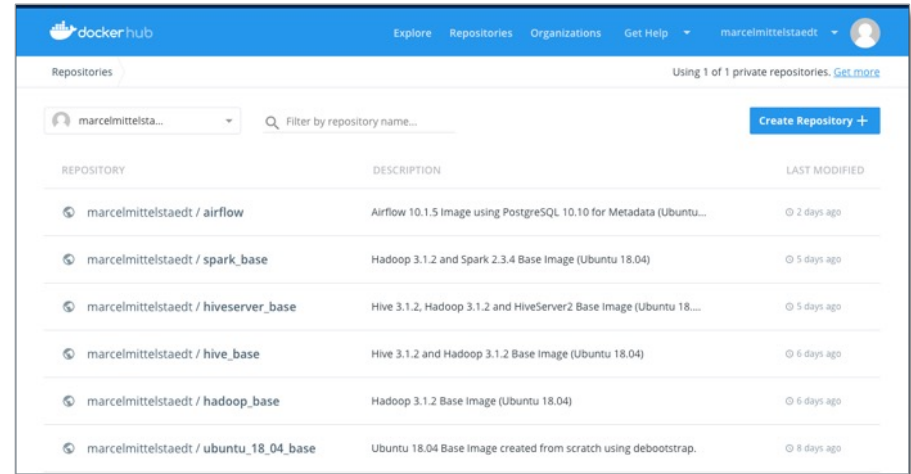


Docker Images/Dockerfiles



The screenshot shows the GitHub repository page for `marcelmittelstaedt / BigData`. The `docker` folder is selected, showing a list of files and their commit history. The `README.md` file is open, displaying the title **Docker Images** and a description: "This folder contains all Docker images used within this lecture. See readme files of Docker Images (within subfolders) for more details, e.g. how to start, stop and use them. You can build the containers on your own or pull them from Docker Repository:". Below the text is a URL: `https://hub.docker.com/u/marcelmittelstaedt`. A section titled **Images:** lists the following Docker images:

- **Hadoop Base Image** Hadoop 3.1.2 Base Image (Ubuntu 18.04)
- **Hadoop and Hive Base Image** Hadoop 3.1.2 and Hive 3.1.2 Base Image (Ubuntu 18.04)
- **Hadoop, Hive and HiveServer2 Base Image** Hadoop 3.1.2, Hive 3.1.2 and HiveServer2 Base Image (Ubuntu 18.04)
- **Spark Base Image** Spark 2.3.4 on Hadoop 3.1.2 as well as Hive 3.1.2 and HiveServer2 Base Image (Ubuntu 18.04)
- **Airflow Base Image** Airflow 1.10.5 with PostgreSQL 10.10 as Metadata Store Base Image (Ubuntu 18.04)



The screenshot shows the Docker Hub page for the user `marcelmittelstaedt`. It displays a list of repositories with the following columns: **REPOSITORY**, **DESCRIPTION**, and **LAST MODIFIED**.

REPOSITORY	DESCRIPTION	LAST MODIFIED
<code>marcelmittelstaedt / airflow</code>	Airflow 10.1.5 Image using PostgreSQL 10.10 for Metadata (Ubuntu...)	2 days ago
<code>marcelmittelstaedt / spark_base</code>	Hadoop 3.1.2 and Spark 2.3.4 Base Image (Ubuntu 18.04)	5 days ago
<code>marcelmittelstaedt / hiveserver_base</code>	Hive 3.1.2, Hadoop 3.1.2 and HiveServer2 Base Image (Ubuntu 18...)	5 days ago
<code>marcelmittelstaedt / hive_base</code>	Hive 3.1.2 and Hadoop 3.1.2 Base Image (Ubuntu 18.04)	6 days ago
<code>marcelmittelstaedt / hadoop_base</code>	Hadoop 3.1.2 Base Image (Ubuntu 18.04)	6 days ago
<code>marcelmittelstaedt / ubuntu_18_04_base</code>	Ubuntu 18.04 Base Image created from scratch using debootstrap.	8 days ago

<https://hub.docker.com/u/marcelmittelstaedt>

<https://github.com/marcelmittelstaedt/BigData/tree/master/docker>

Настройка Контейнера Docker

3. Установка и настройка docker

```
sudo apt-get update  
sudo apt-get install docker.io  
sudo usermod -aG docker $USER  
# exit and login again
```

4. Монтирование **Hadoop** с образом **Hive**

```
docker pull marcelmittelstaedt/hive_base:latest
```

5. Запуск контейнера из смонтированного образа:

```
docker run -dit --name hive_base_container -p 8088:8088 -p 9870:9870 -p 9864:  
9864 marcelmittelstaedt/hive_base:latest
```

Настройка Контейнера Dockerr

6. Show Running Container:

```
docker ps -a
```

```
marcel.mittelstaedt@big-data:~$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
c821a0e1bdcf	marcelmittelstaedt/hive_base:latest	"/startup.sh"	6 minutes ago	Up 6 minutes	0.0.0.0:8088->8088/tcp, 0.0.0.0:9870->9870/tcp	hive_base_container

```
marcel.mittelstaedt@big-data:~$
```

7. Вывести на экран журналы контейнера (дождитесь завершения):

```
docker logs hive_base_container
```

```
[...]  
Stopping namenodes on [localhost]  
Stopping datanodes  
Stopping secondary namenodes [c821a0e1bdcf] Stopping nodemanagers  
Stopping resourcemanager  
Container Startup finished.
```

Настройка Контейнера Docker

8. Войти в оболочку контейнера:

```
hans.wurst@big-data:~$ docker exec -it hive_base_container bash  
root@c821a0e1bdcf:/#
```

9. Переключиться на пользователя hadoop:

```
root@c821a0e1bdcf:/# sudo su hadoop  
hadoop@c821a0e1bdcf:/# cd hadoop@c821a0e1bdcf:~$
```

10. Запустите DFS и YARN:

```
start-all.sh
```

Проверка Hive

11. Проверьте, прошла ли установка и настройка Hive успешно. Запуск Hive

```
hive
```

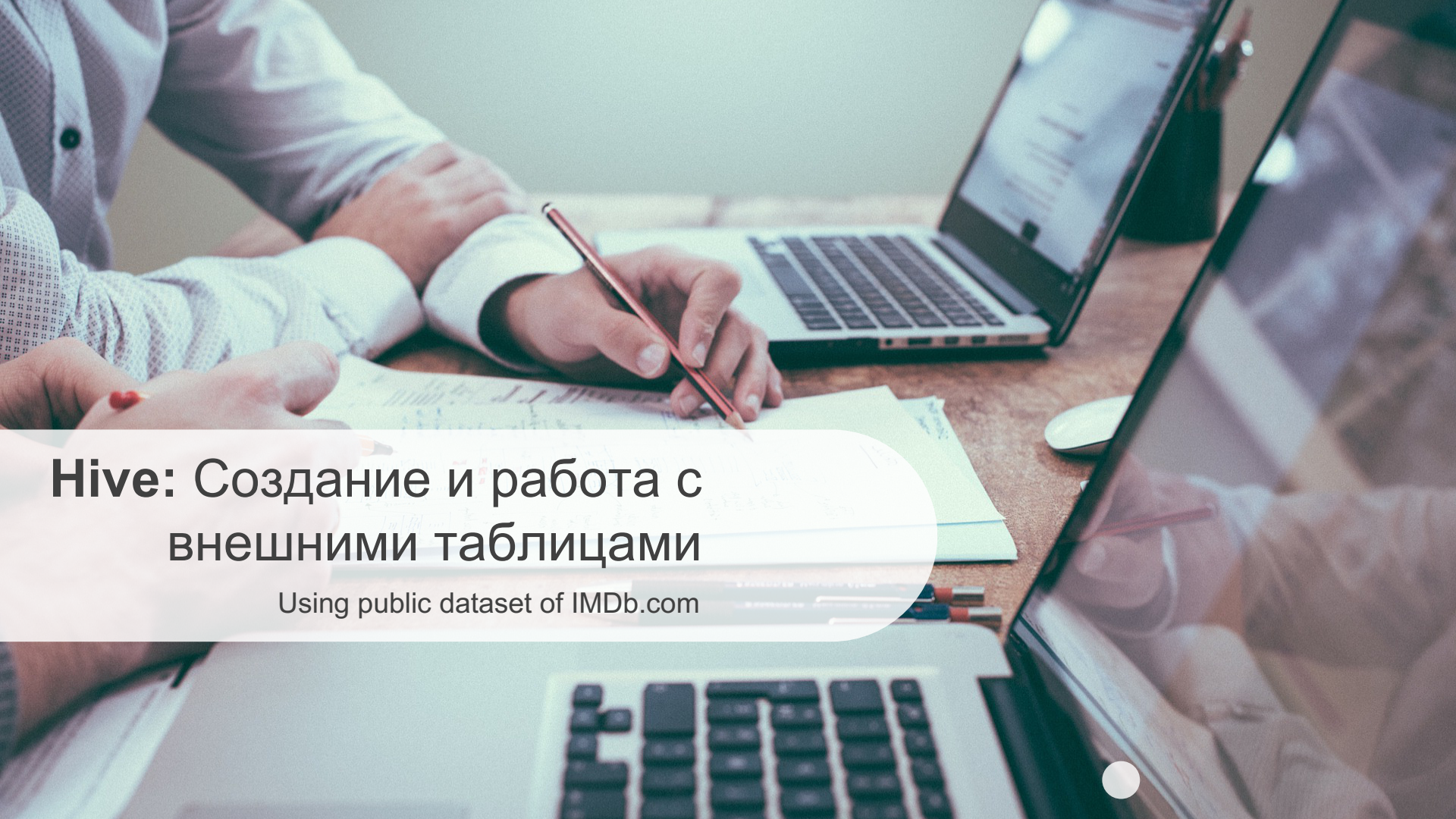


```
SLF4J: Class path contains multiple SLF4J bindings.  
SLF4J: Found binding in [jar:file:/home/hadoop/hive/lib/log4j-slf4j-impl-2.10.0.jar!/org/slf4j/  
impl/StaticLoggerBinder.class]  
SLF4J: Found binding in [jar:file:/home/hadoop/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7  
.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]  
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.  
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]  
Hive Session ID = c120d0b1-9025-43db-96e4-48ccfb875f1a  
  
Logging initialized using configuration in jar:file:/home/hadoop/hive/lib/hive-common-3.1.0.jar  
!/hive-log4j2.properties Async: true  
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider us  
ing a different execution engine (i.e. spark, tez) or using Hive 1.X releases.  
Hive Session ID = fdd6f06a-d4e4-48e6-8971-997e8a0a8e2c  
hive>
```

Установка и настройка Hive

12. Выполнить SQL Запрос:

```
hive> show databases;  
OK  
default  
Time taken: 0.083 seconds, Fetched: 1 row(s) hive>
```

Hive: Создание и работа с внешними таблицами

Using public dataset of IMDb.com



Получение Данных IMDb И Перемещение Их В HDFS

1. Get IMDb Data (<https://www.imdb.com/interfaces/>):

```
wget https://datasets.imdbws.com/title.basics.tsv.gz  
wget https://datasets.imdbws.com/title.ratings.tsv.gz
```

2. Распаковывание IMDb Данных:

```
gunzip title.basics.tsv.gz  
gunzip title.ratings.tsv.gz
```

3. Создание Каталогов HDFS Для Данных IMDb:

```
hadoop fs -mkdir /user/hadoop/imdb  
hadoop fs -mkdir /user/hadoop/imdb/title_basics  
hadoop fs -mkdir /user/hadoop/imdb/title_ratings
```

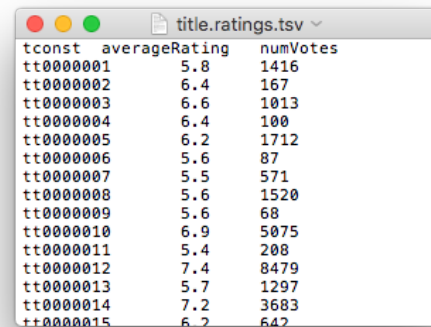
Создание Внешних Таблиц В Hive

4. Перенос файлов данных IMDb в HDFS:

```
hadoop fs -put title.basics.tsv /user/hadoop/imdb/title_basics/title.basics.tsv
hadoop fs -put title.ratings.tsv /user/hadoop/imdb/title_ratings/title.ratings.tsv
```

5. Создание Внешней таблицы **title_ratings** (file *title.ratings.tsv*) в Hive:

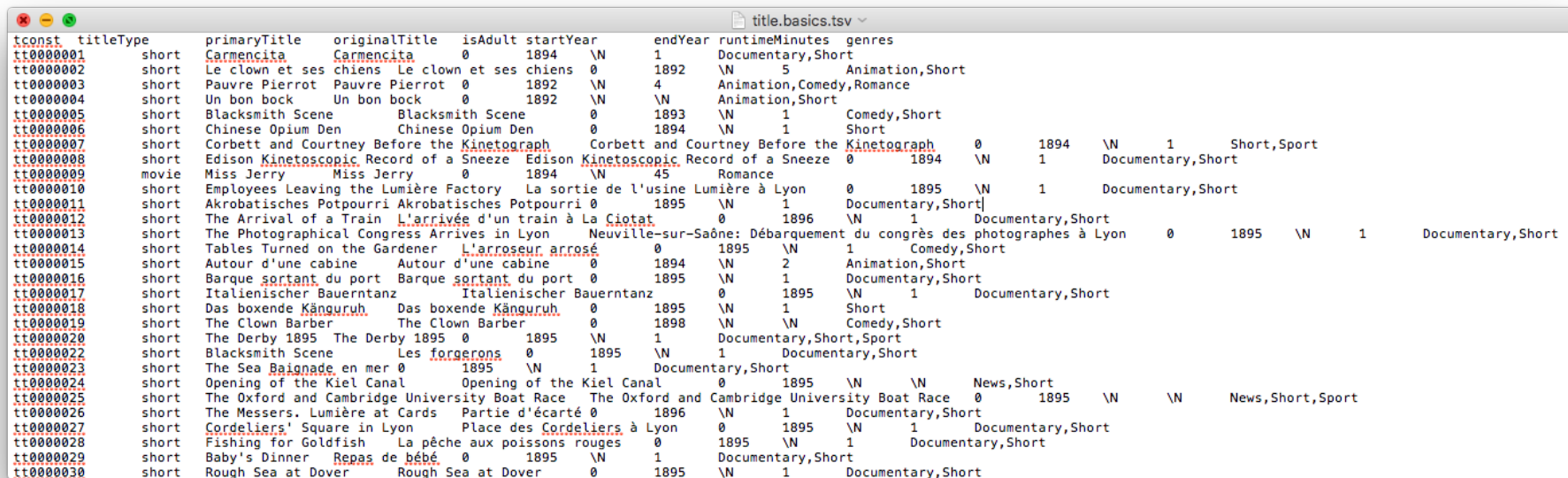
```
hive > CREATE EXTERNAL TABLE IF NOT EXISTS title_ratings(
    tconst STRING,
    average_rating DECIMAL(2,1),
    num_votes BIGINT
) COMMENT 'IMDb Ratings'
ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t' STORED AS
TEXTFILE LOCATION '/user/hadoop/imdb/title_ratings'
TBLPROPERTIES ('skip.header.line.count'='1');
```



tconst	averageRating	numVotes
tt0000001	5.8	1416
tt0000002	6.4	167
tt0000003	6.6	1013
tt0000004	6.4	100
tt0000005	6.2	1712
tt0000006	5.6	87
tt0000007	5.5	571
tt0000008	5.6	1520
tt0000009	5.6	68
tt0000010	6.9	5075
tt0000011	5.4	208
tt0000012	7.4	8479
tt0000013	5.7	1297
tt0000014	7.2	3683
tt0000015	6.2	642

Создание Внешних Таблиц В Hive

6. Создать Внешнюю таблицу **title_basics** для файла **title.basics.tsv** в Hive:



id	titleType	primaryTitle	originalTitle	isAdult	startYear	endYear	runtimeMinutes	genres
tt0000001	short	Carmencita	Carmencita	0	1894	\N	1	Documentary,Short
tt0000002	short	Le clown et ses chiens	Le clown et ses chiens	0	1892	\N	5	Animation,Short
tt0000003	short	Pauvre Pierrot	Pauvre Pierrot	0	1892	\N	4	Animation,Comedy,Romance
tt0000004	short	Un bon bock	Un bon bock	0	1892	\N	\N	Animation,Short
tt0000005	short	Blacksmith Scene	Blacksmith Scene	0	1893	\N	1	Comedy,Short
tt0000006	short	Chinese Opium Den	Chinese Opium Den	0	1894	\N	1	Short
tt0000007	short	Corbett and Courtney Before the Kinetograph	Corbett and Courtney Before the Kinetograph	0	1894	\N	1	Short,Sport
tt0000008	short	Edison Kinetoscopic Record of a Sneeze	Edison Kinetoscopic Record of a Sneeze	0	1894	\N	1	Documentary,Short
tt0000009	movie	Miss Jerry	Miss Jerry	0	1894	\N	45	Romance
tt0000010	short	Employees Leaving the Lumière Factory	La sortie de l'usine Lumière à Lyon	0	1895	\N	1	Documentary,Short
tt0000011	short	Akrobatisches Potpourri	Akrobatisches Potpourri	0	1895	\N	1	Documentary,Short
tt0000012	short	The Arrival of a Train	L'arrivée d'un train à La Ciotat	0	1896	\N	1	Documentary,Short
tt0000013	short	The Photographical Congress Arrives in Lyon	Neuville-sur-Saône: Débarquement du congrès des photographes à Lyon	0	1895	\N	1	Documentary,Short
tt0000014	short	Tables Turned on the Gardener	L'arroseur arrosé	0	1895	\N	1	Comedy,Short
tt0000015	short	Autour d'une cabine	Autour d'une cabine	0	1894	\N	2	Animation,Short
tt0000016	short	Barque sortant du port	Barque sortant du port	0	1895	\N	1	Documentary,Short
tt0000017	short	Italienischer Bauerntanz	Italienischer Bauerntanz	0	1895	\N	1	Documentary,Short
tt0000018	short	Das boxende Känguruh	Das boxende Känguruh	0	1895	\N	1	Short
tt0000019	short	The Clown Barber	The Clown Barber	0	1898	\N	\N	Comedy,Short
tt0000020	short	The Derby 1895	The Derby 1895	0	1895	\N	1	Documentary,Short,Sport
tt0000022	short	Blacksmith Scene	Les forgerons	0	1895	\N	1	Documentary,Short
tt0000023	short	The Sea Baignade en mer		0	1895	\N	1	Documentary,Short
tt0000024	short	Opening of the Kiel Canal	Opening of the Kiel Canal	0	1895	\N	\N	News,Short
tt0000025	short	The Oxford and Cambridge University Boat Race	The Oxford and Cambridge University Boat Race	0	1895	\N	\N	News,Short,Sport
tt0000026	short	The Messers. Lumière at Cards	Partie d'écarté	0	1896	\N	1	Documentary,Short
tt0000027	short	Cordeliers' Square in Lyon	Place des Cordeliers à Lyon	0	1895	\N	1	Documentary,Short
tt0000028	short	Fishing for Goldfish	La pêche aux poissons rouges	0	1895	\N	1	Documentary,Short
tt0000029	short	Baby's Dinner	Repas de bébé	0	1895	\N	1	Documentary,Short
tt0000030	short	Rough Sea at Dover	Rough Sea at Dover	0	1895	\N	1	Documentary,Short

Создание Внешних Таблиц В Hive

6. Создание Внешней таблицы **title_basics** для файла **title.basics.tsv** в Hive:

```
hive > CREATE EXTERNAL TABLE IF NOT EXISTS title_basics (  
    tconst STRING,  
    title_type STRING,  
    primary_title STRING,  
    original_title STRING,  
    is_adult DECIMAL(1,0),  
    start_year DECIMAL(4,0),  
    end_year STRING,  
    runtime_minutes INT,  
    genres STRING  
) COMMENT 'IMDb Movies' ROW FORMAT DELIMITED FIELDS TERMINATED BY  
'\t' STORED AS TEXTFILE LOCATION '/user/hadoop/imdb/title_basics'  
TBLPROPERTIES ('skip.header.line.count'='1');
```

Создание Внешних Таблиц В Hive

7. Запрос таблицы `title_basics` в Hive с использованием SQL (HiveQL):

```
hive> select * from title_basics limit 3;
OK
tt0000001 short Carmencita Carmencita 0 1894 NULL 1 Documentary,Short
tt0000002 short Le clown et ses chiens Le clown et ses chiens 0 1892 NULL 5 Animation,Short
tt0000003 short Pauvre Pierrot Pauvre Pierrot 0 1892 NULL 4 Animation,Comedy,Romance
Time taken: 0.139 seconds, Fetched: 5 row(s)
hive>
```

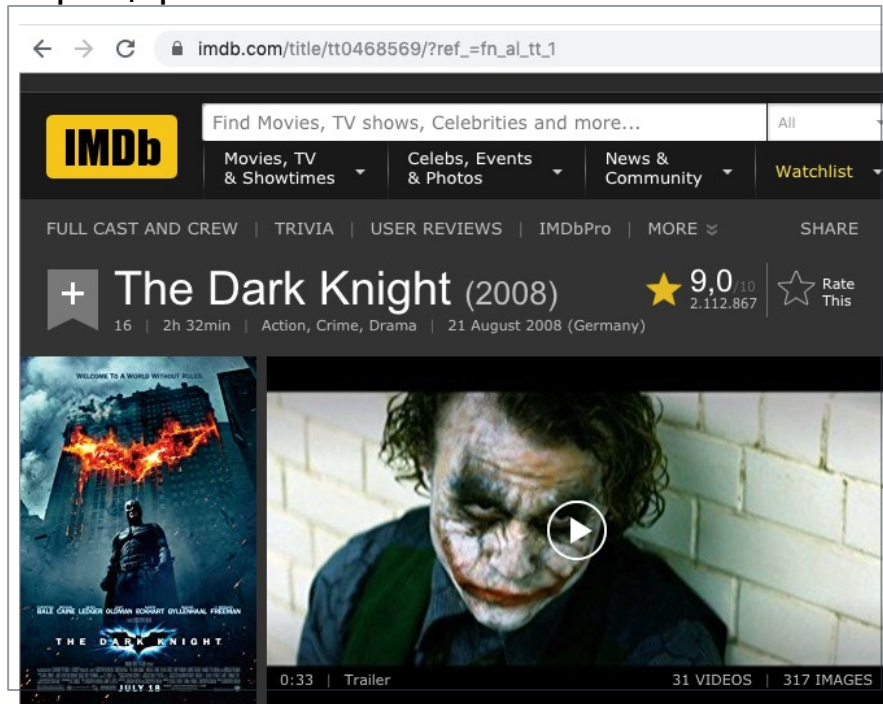
8. Запрос таблицы `title_ratings` в Hive с использованием SQL (HiveQL):

```
hive> select * from title_ratings limit 3;
OK
tt0000001 5.6 1540
tt0000002 6.1 186
tt0000003 6.5 1199
Time taken: 0.119 seconds, Fetched: 3 row(s)
hive>
```


Создание Внешних Таблиц В Hive

9. Запустите сложный запрос, который запускает задание MapReduce на Yarn, например, получите рейтинг фильма „Темный рыцарь“:

```
SELECT
  *
FROM
  title_basics b
  JOIN title_ratings r ON (b.tconst=r.tconst)
WHERE
  original_title = 'The Dark Knight'
  AND title_type='movie';
```




Создание Внешних Таблиц В Hive

9. Выполненный запрос

```
hive> SELECT * FROM title_basics b JOIN title_ratings r ON (b.tconst=r.tconst) WHERE original_title =  
'The Dark Knight' and title_type='movie';  
[...]  
Starting Job = job_1613323804972_0003, Tracking URL = http://154172c92bc7:8088/proxy/application_1613323804972_0003/  
Kill Command = /home/hadoop/hadoop/bin/mapred job -kill job_1613323804972_0003  
Hadoop job information for Stage-3: number of mappers: 3; number of reducers: 0  
2021-02-14 17:37:59,326 Stage-3 map = 0%, reduce = 0%  
2021-02-14 17:38:20,617 Stage-3 map = 50%, reduce = 0%, Cumulative CPU 35.59 sec  
2021-02-14 17:38:21,656 Stage-3 map = 67%, reduce = 0%, Cumulative CPU 52.65 sec  
2021-02-14 17:38:22,718 Stage-3 map = 100%, reduce = 0%, Cumulative CPU 56.84 sec  
MapReduce Total cumulative CPU time: 56 seconds 840 msec  
Ended Job = job_1613323804972_0003  
MapReduce Jobs Launched:  
Stage-Stage-3: Map: 3 Cumulative CPU: 56.84 sec HDFS Read: 650217476 HDFS Write: 376 SUCCESS  
Total MapReduce CPU Time Spent: 56 seconds 840 msec  
OK  
tt0468569 movie The Dark Knight The Dark Knight 0 2008 NULL 152 Action,Crime,Drama tt0468569 9.0 2111245  
Time taken: 53.094 seconds, Fetched: 1 row(s)  
hive>
```


Создание Внешних Таблиц В Hive

9. Проверьте результат в YARN (<http://XXX.XXX.XXX.XXX:8088/cluster/>):



Cluster

About
Nodes
Node Labels
Applications

NEW
NEW SAVING
SUBMITTED
ACCEPTED
RUNNING
FINISHED
FAILED
KILLED

Scheduler

Tools

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved
5	0	1	4	4	11 GB	16 GB	0 B	4	8	0

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes	Shutdown Nodes
1	0	0	0	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation	Maximum Cluster Application Priority
Capacity Scheduler	[memory-mb (unit=Mi), vcores]	<memory:1024, vCores:1>	<memory:8192, vCores:4>	0

Show 20 entries

ID	User	Name	Application Type	Queue	Application Priority	StartTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCoers	Allocated Memory MB	Reserved CPU VCoers	Reserved Memory MB	% of Queue	% of Cluster	Progress	Tracking UI	Blacklisted Nodes
application_1613323804972_0005	hadoop	SELECT * FROM title_bas...title_type='movie' (Stage-3)	MAPREDUCE	default	0	Sun Feb 14 18:41:12 +0100 2021	N/A	RUNNING	UNDEFINED	4	4	11264	0	0	68.8	68.8	<div></div>	ApplicationMaster	0

Showing 1 to 1 of 1 entries



ЗАДАНИЕ

Hive: Создание и работа с внешними таблицами



HDFS и Hive QL упражнения - IMDB

1. Выполните все действия на слайдах выше
2. Скачать <https://datasets.imdbws.com/name.basics.tsv.gz>
3. Создать каталог HDFS `/user/hadoop/imdb/name_basics/` для файла **name.basics.tsv**
4. Создайте внешнюю Hive таблицу **name_basics** для **name.basics.tsv**
5. Используйте HiveQL, чтобы ответить на следующие вопросы:
 - a) Сколько фильмов и сериалов находится в наборе данных IMDB?
 - b) Кто самый молодой актер/сценарист/... в наборе данных?
 - c) Создайте список (`tconst`, `original_title`, `start_year`, `average_rating`, `num_votes`), который состоит из:
 - фильм вышел в 2010 году или позднее;
 - фильм имеет средний рейтинг, равный или превышающий 8,1
 - проголосовали более 100 000 раз
 - d) Сколько фильмов находится в списке c)?

HDFS и Hive QL упражнения - IMDB

5. Используйте **HiveQL**, чтобы ответить на следующие вопросы::

е) Мы хотим знать, какие годы были великими для кинематографа.

Создайте список с одной строкой в год и соответствующим количеством фильмов, которые:

- имеют средний рейтинг выше 8;
- были проголосованы более 100 000 раз в порядке убывания количества фильмов.