

KillTest

高品質 より良いサービス



練習問題

<https://www.killtest.jp>

1年間無料アップデート

Exam : **CCA175**

Title : CCA Spark and Hadoop
Developer Exam

Version: DEMO

1.Problem Scenario 1:

You have been given MySQL DB with following details.

user=retail_dba

password=cloudera

database=retail_db

table=retail_db.categories

jdbc URL = jdbc:mysql://quickstart:3306/retail_db

Please accomplish following activities.

1. Connect MySQL DB and check the content of the tables.
2. Copy "retaildb.categories" table to hdfs, without specifying directory name.
3. Copy "retaildb.categories" table to hdfs, in a directory name "categories_target".
4. Copy "retaildb.categories" table to hdfs, in a warehouse directory name "categories_warehouse".

Answer:

Solution:

Step 1: Connecting to existing MySQL Database `mysql --user=retail_dba --password=cloudera retail_db`

Step 2: Show all the available tables `show tables;`

Step 3: View/Count data from a table in MySQL `select count(1) from categories;`

Step 4: Check the currently available data in HDFS directory `hdfs dfs -ls`

Step 5: Import Single table (Without specifying directory).

`sqoop import --connect jdbc:mysql://quickstart:3306/retail_db -username=retail_dba -password=cloudera -table=categories`

Note: Please check you dont have space between before or after '=' sign. Sqoop uses the MapReduce framework to copy data from RDBMS to hdfs

Step 6: Read the data from one of the partition, created using above command, `hdfs dfs`

`-cat categories/part-m-00000`

Step 7: Specifying target directory in import command (We are using number of mappers =1, you can change accordingly) `sqoop import -connect jdbc:mysql://quickstart:3306/retail_db -username=retail_dba -password=cloudera -table=categories -target-dir=categories_target --m 1`

Step 8: Check the content in one of the partition file.

`hdfs dfs -cat categories_target/part-m-00000`

Step 9: Specifying parent directory so that you can copy more than one table in a specified target directory.

Command to specify warehouse directory.

`sqoop import --connect jdbc:mysql://quickstart:3306/retail_db --username=retail_dba -password=cloudera -table=categories -warehouse-dir=categories_warehouse --m 1`

2.Problem Scenario 2:

There is a parent organization called "ABC Group Inc", which has two child companies named Tech Inc and MPTech.

Both companies employee information is given in two separate text file as below. Please do the following activity for employee details.

Tech Inc.txt

1, Alok, Hyderabad

2, Krish, Hongkong

3, Jyoti, Mumbai

- 4, Atul, Bangalore
- 5, Ishan, Gurgaon
- MPTech.txt
- 6, John, Newyork
- 7, alp2004, California
- 8, Tellme, Mumbai
- 9, Gagan21, Pune
- 10, Mukesh, Chennai

1.

Which command will you use to check all the available command line options on HDFS and How will you get the Help for individual command.

2. Create a new Empty Directory named Employee using Command line. And also create an empty file named in it Techinc.txt
3. Load both companies Employee data in Employee directory (How to override existing file in HDFS).
4. Merge both the Employees data in a Single tile called MergedEmployee.txt, merged tiles should have new line character at the end of each file content.
5. Upload merged file on HDFS and change the file permission on HDFS merged file, so that owner and group member can read and write, other user can read the file.
6. Write a command to export the individual file as well as entire directory from HDFS to local file System.

Answer:

Solution:

Step 1: Check All Available command `hdfs dfs`

Step 2: Get help on Individual command `hdfs dfs -help get`

Step 3: Create a directory in HDFS using named Employee and create a Dummy file in it called e.g. Techinc.txt `hdfs dfs -mkdir Employee`

Now create an empty file in Employee directory using Hue.

Step 4: Create a directory on Local file System and then Create two files, with the given data in problems.

Step 5: Now we have an existing directory with content in it, now using HDFS command line, overrid this existing Employee directory. While copying these files from local file System to HDFS. `cd`

`/home/cloudera/Desktop/ hdfs dfs -put -f Employee`

Step 6: Check All files in directory copied successfully `hdfs dfs -ls Employee`

Step 7: Now merge all the files in Employee directory, `hdfs dfs -getmerge -nl Employee MergedEmployee.txt`

Step 8: Check the content of the file. `cat MergedEmployee.txt`

Step 9: Copy merged file in Employeeed directory from local file ssytem to HDFS. `hdfs dfs -put MergedEmployee.txt Employee/`

Step 10: Check file copied or not. `hdfs dfs -ls Employee`

Step 11: Change the permission of the merged file on HDFS `hdfs dfs -chmpd 664 Employee/MergedEmployee.txt`

Step 12: Get the file from HDFS to local file system, `hdfs dfs -get Employee Employee_hdfs`

3.Problem Scenario 3: You have been given MySQL DB with following details.

user=retail_dba

password=cloudera

database=retail_db

table=retail_db.categories

jdbc URL = jdbc:mysql://quickstart:3306/retail_db

Please accomplish following activities.

1. Import data from categories table, where category=22 (Data should be stored in categories_subset)
2. Import data from categories table, where category>22 (Data should be stored in categories_subset_2)
3. Import data from categories table, where category between 1 and 22 (Data should be stored in categories_subset_3)
4. While importing categories data change the delimiter to '|' (Data should be stored in categories_subset_5)
5. Importing data from categories table and restrict the import to category_name, category id columns only with delimiter as '|'
6. Add null values in the table using below SQL statement ALTER TABLE categories modify category_department_id int(11); INSERT INTO categories values (eO.NULL.'TESTING');
7. Importing data from categories table (In categories_subset_17 directory) using '|' delimiter and categoryjd between 1 and 61 and encode null values for both string and non string columns.
8. Import entire schema retail_db in a directory categories_subset_all_tables

Answer:

Solution:

Step 1: Import Single table (Subset data) Note: Here the ' is the same you find on - key

```
sqoop import --connect jdbc:mysql://quickstart:3306/retail_db --username=retail_dba  
-password=cloudera -table=categories ~warehouse-dir= categories_subset --where 'category_id'=22  
--m 1
```

Step 2: Check the output partition

```
hdfs dfs -cat categories_subset/categories/part-m-00000
```

Step 3: Change the selection criteria (Subset data)

```
sqoop import --connect jdbc:mysql://quickstart:3306/retail_db --username=retail_dba  
-password=cloudera -table=categories ~warehouse-dir= categories_subset_2 --where 'category_id'\>22  
-m 1
```

Step 4: Check the output partition

```
hdfs dfs -cat categories_subset_2/categories/part-m-00000
```

Step 5: Use between clause (Subset data)

```
sqoop import --connect jdbc:mysql://quickstart:3306/retail_db --username=retail_dba  
-password=cloudera -table=categories ~warehouse-dir=categories_subset_3 --where "'category_id'  
between 1 and 22" --m 1
```

Step 6: Check the output partition

```
hdfs dfs -cat categories_subset_3/categories/part-m-00000
```

Step 7: Changing the delimiter during import.

```
sqoop import --connect jdbc:mysql://quickstart:3306/retail_db --username=retail_dba -password=cloudera  
-table=categories -warehouse-dir=:categories_subset_6 --where "'categoryjd '/' between 1 and 22"  
-fields-terminated-by='|' -m 1
```

Step 8: Check the output partition

```
hdfs dfs -cat categories_subset_6/categories/part-m-00000
```

Step 9: Selecting subset columns

```
sqoop import --connect jdbc:mysql://quickstart:3306/retail_db --username=retail_dba  
-password=cloudera -table=categories --warehouse-dir=categories_subset_col -where "'category id'  
between 1 and 22" -fields-terminated-by=T -columns=category name, category id --m 1
```

Step 10: Check the output partition

```
hdfs dfs -cat categories_subset_col/categories/part-m-00000
```

Step 11: Inserting record with null values (Using mysql) ALTER TABLE categories modify category_department_id int(11); INSERT INTO categories values ^NULL/TESTING'); select" from categories;

Step 12: Encode non string null column

```
sqoop import --connect jdbc:mysql://quickstart:3306/retail_db --username=retail_dba -password=cloudera  
-table=categories --warehouse-dir=categories_subset_17 -where "\"category_id\" between 1 and 61"  
-fields-terminated-by=, |' --null-string-N' -null-non-string=, N' --m 1
```

Step 13: View the content

```
hdfs dfs -cat categories_subset_17/categories/part-m-00000
```

Step 14: Import all the tables from a schema (This step will take little time)

```
sqoop import-all-tables -connect jdbc:mysql://quickstart:3306/retail_db --username=retail_dba  
-password=cloudera -warehouse-dir=categories_si
```

Step 15: View the contents

```
hdfs dfs -ls categories_subset_all_tables
```

Step 16: Cleanup or back to originals.

```
delete from categories where categoryid in (59, 60);
```

```
ALTER TABLE categories modify category_department_id int(11) NOTNULL;
```

```
ALTER TABLE categories modify category_name varchar(45) NOT NULL;
```

```
desc categories;
```

4.Problem Scenario 4: You have been given MySQL DB with following details.

user=retail_dba

password=cloudera

database=retail_db

table=retail_db.categories

jdbc URL = jdbc:mysql://quickstart:3306/retail_db

Please accomplish following activities.

Import Single table categories (Subset data) to hive managed table, where category_id between 1 and 22

Answer:

Solution:

Step 1: Import Single table (Subset data)

```
sqoop import --connect jdbc:mysql://quickstart:3306/retail_db -username=retail_dba -password=cloudera  
-table=categories -where "'category_id' between 1 and 22" --hive-import --m 1
```

Note: Here the ' is the same you find on ~ key

This command will create a managed table and content will be created in the following directory.

/user/hive/warehouse/categories

Step 2: Check whether table is created or not (In Hive)

```
show tables;
```

```
select * from categories;
```

5.Problem Scenario 5: You have been given following mysql database details.

user=retail_dba

password=cloudera

database=retail_db

jdbc URL = jdbc:mysql://quickstart:3306/retail_db

Please accomplish following activities.

1. List all the tables using sqoop command from retail_db
2. Write simple sqoop eval command to check whether you have permission to read database tables or not.
3. Import all the tables as avro files in /user/hive/warehouse/retail cca174.db
4. Import departments table as a text file in /user/cloudera/departments.

Answer:

Solution:

Step 1: List tables using sqoop

```
sqoop list-tables --connect jdbc:mysql://quickstart:3306/retail_db --username retail_dba -password cloudera
```

Step 2: Eval command, just run a count query on one of the table.

```
sqoop eval \  
--connect jdbc:mysql://quickstart:3306/retail_db \  
-username retail_dba \  
-password cloudera \  
--query "select count(1) from ordeMtems"
```

Step 3: Import all the tables as avro file.

```
sqoop import-all-tables \  
--connect jdbc:mysql://quickstart:3306/retail_db \  
-username=retail_dba \  
-password=cloudera \  
-as-avrodatafile \  
-warehouse-dir=/user/hive/warehouse/retail stage.db \  
-ml
```

Step 4: Import departments table as a text file in /user/cloudera/departments

```
sqoop import \  
--connect jdbc:mysql://quickstart:3306/retail_db \  
-username=retail_dba \  
-password=cloudera \  
-table departments \  
-as-textfile \  
-target-dir=/user/cloudera/departments
```

Step 5: Verify the imported data.

```
hdfs dfs -ls /user/cloudera/departments
```

```
hdfs dfs -ls /user/hive/warehouse/retailstage.db
```

```
hdfs dfs -ls /user/hive/warehouse/retail_stage.db/products
```