**SQOOP Operations - Lab Manual**

**Aim:**

To understand and perform **data transfer between Hadoop and relational databases** using Apache Sqoop.

**Understanding Apache Sqoop:**

Apache Sqoop is a tool used for **transferring structured data** between **RDBMS (MySQL, PostgreSQL, etc.)** and **Hadoop (HDFS, Hive, HBase, etc.)** efficiently.

**1. Checking Sqoop Installation**

**Command:**

sqoop version

**2. Listing Databases in MySQL**

**Command:**

sqoop list-databases --connect jdbc:mysql://localhost:3306/ --username root --password root

**3. Listing Tables in a Database**

**Command:**

sqoop list-tables --connect jdbc:mysql://localhost:3306/employees --username root --password root

**4. Importing a Table from MySQL to HDFS**

**Command:**

sqoop import --connect jdbc:mysql://localhost:3306/employees --username root --password root \

--table employee\_details --m 1 --target-dir /sqoop\_output/employee\_details

**5. Importing a Table into Hive**

**Command:**

sqoop import --connect jdbc:mysql://localhost:3306/employees --username root --password root \

--table employee\_details --hive-import --hive-database employees\_hive

**6. Importing Specific Columns from MySQL to HDFS**

**Command:**

sqoop import --connect jdbc:mysql://localhost:3306/employees --username root --password root \

--table employee\_details --columns "employee\_id,name,salary" --target-dir /sqoop\_output/employees\_selected

**7. Importing Data with a Condition (Filtering Rows)**

**Command:**

sqoop import --connect jdbc:mysql://localhost:3306/employees --username root --password root \

--table employee\_details --where "salary > 50000" --target-dir /sqoop\_output/high\_salary

**8. Exporting Data from HDFS to MySQL**

**Command:**

sqoop export --connect jdbc:mysql://localhost:3306/employees --username root --password root \

--table new\_employee\_details --export-dir /sqoop\_output/new\_data

**9. Incremental Data Import (Appending New Data Only)**

**Command:**

sqoop import --connect jdbc:mysql://localhost:3306/employees --username root --password root \

--table employee\_details --incremental append --check-column employee\_id --last-value 100

**10. Importing Data in Parallel Using Multiple Mappers**

**Command:**

sqoop import --connect jdbc:mysql://localhost:3306/employees --username root --password root \

--table employee\_details --num-mappers 4 --target-dir /sqoop\_output/employees\_parallel

**Example Output (for Table Import in HDFS):**

INFO mapreduce.ImportJobBase: Retrieved 100 records.

INFO mapreduce.ImportJobBase: Data successfully stored in HDFS at /sqoop\_output/employee\_details

**Example Output:**

pgsql

CopyEdit

INFO sqoop.Sqoop: Running Sqoop version 1.4.7

INFO manager.SqlManager: Successfully connected to database

INFO tool.ImportTool: Importing data to HDFS location: /sqoop\_output/employee\_details

INFO mapreduce.ImportJobBase: Retrieved 100 records.

INFO mapreduce.ImportJobBase: Data successfully stored in HDFS at /sqoop\_output/employee\_details

INFO tool.ImportTool: Completed Successfully

**Final Result:**

Apache Sqoop successfully **transfers structured data between relational databases and Hadoop**, enabling **efficient big data processing**. These commands help in importing, exporting, filtering, and optimizing data movement in a **Hadoop-based data pipeline**.