



**INSTITUTO POLITÉCNICO
NACIONAL**

**ESCUELA SUPERIOR DE
CÓMPUTO**

**INGENIERÍA EN SISTEMAS
COMPUTACIONALES**



UNIDAD DE APRENDIZAJE

Sistemas Distribuidos

GRUPO

7CV2

Práctica 2. Procesamiento de tablas agregadas mediante MapReduce

ALUMNO

Díaz Ortiz Brandon Aldair

PROFESOR

Carlos Pineda Guerrero

Tabla de contenido

DESARROLLO	3
CREACIÓN Y CARGA DE LA BASE DE DATOS	3
1. Crear una máquina virtual con Ubuntu en Azure, 4 GB de memoria RAM, 2 CPU virtuales y disco SSD estándar.	3
2. Instalar el servidor MySQL en la máquina virtual.	4
3. Crear la base de datos "practica_olap":	4
4. Descargar el archivo "practica_olap.sql" de la plataforma Moodle.	5
5. Descargar el archivo "sales_data.csv" de la plataforma Moodle.	5
6. Crear las tablas ejecutando el script "practica_olap.sql" en la línea de comandos:.....	5
7. Ejecutar el monitor de mysql incluyendo la opción "local-infile" para habilitar la carga de datos:	6
8. Cargar los datos a la tabla "sales_data", ejecutando los siguientes comandos en el monitor de mysql:.....	6
10. Cargar las tablas de dimensiones y la fact table a partir de los datos de la tabla "sales_data".	7
11. Ejecutar el monitor de mysql:	8
12. Ejecutar la siguiente instrucción SELECT para saber qué directorio puede escribir MySQL:	9
13. Ejecutar la siguiente instrucción SELECT para generar un archivo CSV con id_contry, id_category, id_product y sales:.....	9
INSTALACIÓN Y EJECUCIÓN DE APACHE HADOOP	10
14. Ahora vamos a descargar e instalar Apache Hadoop. Ejecutar la siguiente instrucción en la máquina virtual:	10
15. Descomprimir y desempacar el archivo descargado:	10
16. Apache Hadoop requiere Java, entonces necesitamos instalar el JDK en la máquina virtual. Instalar el openjdk 16 o mayor.....	11
17. Ahora vamos a editar el archivo "hadoop-3.4.0/etc/hadoop/hadoop-env.sh":.....	11
18. Quitar el comentario a "export JAVA_HOME" y asignarle "/usr" (el directorio que contiene el directorio "bin" que a su vez contiene los programas "java" y "javac"):	12
19. Ahora vamos a crear una aplicación Hadoop que obtenga una tabla agregada a partir del archivo "country_category_product.csv"......	12
20. Ahora se cargará el archivo resultante del proceso MapReduce, a la tabla agregada "country_category_product" de la base de datos "practica_olap":	15
CONCLUSIONES	18

Desarrollo

Creación y carga de la base de datos

1. Crear una máquina virtual con Ubuntu en Azure, 4 GB de memoria RAM, 2 CPU virtuales y disco SSD estándar.

El nombre de la máquina virtual deberá ser: "P2-" concatenando el número de boleta del alumno o alumna y la palabra "-dss" por ejemplo, si el número de boleta es 12345678, entonces la máquina virtual deberá llamarse: P2-12345678-dss

Incluir en el documento PDF la captura de la última pantalla, correspondiente a la creación de la máquina virtual.

The screenshot displays the Microsoft Azure portal interface. At the top, the header shows the Microsoft Azure logo, a search bar, and the user's profile (bdiazo1800@alumno.ip...). The main content area is titled "CreateVm-canonical.0001-com-ubuntu-server-focal-2-20241107193917 | Información general". A green checkmark icon indicates that the deployment is complete. The text "Se completó la implementación" is prominently displayed. Below this, details of the implementation are shown, including the name of the implementation, the subscription (Azure for Students), the resource group (practica2), the start time (7/11/2024, 7:43:48 p.m.), and the correlation ID. A section titled "Pasos siguientes" (Next steps) lists recommended actions: "Configurar el apagado automático" (Recommended), "Supervisar el estado, el rendimiento y las dependencias de red de la máquina virtual" (Recommended), and "Ejecutar un script dentro de la máquina virtual" (Recommended). At the bottom of this section, there are buttons for "Ir al recurso" (Go to resource) and "Crear otra VM" (Create another VM). On the right side of the screen, there are several informational cards: "Cost Management" (Obtenga una notificación para permanecer dentro del presupuesto y evitar cargos inesperados en su factura), "Microsoft Defender for Cloud" (Proteja sus aplicaciones e infraestructura), "Tutoriales gratuitos de Microsoft" (Comience a aprender hoy), and "Trabajar con un experto" (Los expertos de Azure son asociados proveedores de servicios que pueden ayudar a administrar sus recursos en Azure y ser la primera línea de soporte técnico).

2. Instalar el servidor MySQL en la máquina virtual.

```
ubuntu@P2-2022630588-dss:~$ sudo apt install mysql-server -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libbcgi-fast-perl libbcgi-pm-perl libencode-locale-perl libevent-core-2.1-7 libevent-pthreads-2.1-7 libfcgi-perl libhtml-parser-perl libhtml-tagset-perl libhtml-template-perl
  libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libmecab2 libtimedate-perl liburi-perl mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0
  mysql-client-core-8.0 mysql-common mysql-server mysql-server-8.0 mysql-server-core-8.0
Suggested packages:
  libdata-dump-perl libipc-sharedcache-perl libwww-perl mailx tinycsa
The following NEW packages will be installed:
  libbcgi-fast-perl libbcgi-pm-perl libencode-locale-perl libevent-core-2.1-7 libevent-pthreads-2.1-7 libfcgi-perl libhtml-parser-perl libhtml-tagset-perl libhtml-template-perl
  libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libmecab2 libtimedate-perl liburi-perl mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0
  mysql-client-core-8.0 mysql-common mysql-server mysql-server-8.0 mysql-server-core-8.0
0 upgraded, 25 newly installed, 0 to remove and 0 not upgraded.
Need to get 36.9 MB of archives.
After this operation, 318 MB of additional disk space will be used.
Get:1 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 mysql-common all 5.8+1.0.5ubuntu2 [7496 B]
Get:2 http://azure.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-core-8.0 amd64 8.0.39-0ubuntu0.20.04.1 [5088 kB]
Get:3 http://azure.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-8.0 amd64 8.0.39-0ubuntu0.20.04.1 [22.0 kB]
Get:4 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libevent-core-2.1-7 amd64 2.1.11-stable-1 [89.1 kB]
Get:5 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libevent-pthreads-2.1-7 amd64 2.1.11-stable-1 [7372 B]
Get:6 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libmecab2 amd64 0.996-10build1 [233 kB]
Get:7 http://azure.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-server-core-8.0 amd64 8.0.39-0ubuntu0.20.04.1 [22.8 MB]
Get:8 http://azure.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-server-8.0 amd64 8.0.39-0ubuntu0.20.04.1 [1326 kB]
Get:9 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libhtml-tagset-perl all 3.20-4 [12.5 kB]
Get:10 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 liburi-perl all 1.76-2 [77.5 kB]
Get:11 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libhtml-parser-perl amd64 3.72-5 [86.3 kB]
Get:12 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libbcgi-pm-perl all 4.46-1 [186 kB]
Get:13 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libfcgi-perl amd64 0.79-1 [33.1 kB]
Get:14 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libbcgi-fast-perl all 1:2.15-1 [10.5 kB]
Get:15 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libencode-locale-perl all 1.05-1 [12.3 kB]
Get:16 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libhtml-template-perl all 2.97-1 [59.0 kB]
Get:17 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libtimedate-perl all 2.3200-1 [34.0 kB]
Get:18 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libhttp-date-perl all 6.05-1 [9920 B]
Get:19 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libio-html-perl all 1.001-1 [14.9 kB]
Get:20 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 liblwp-mediatypes-perl all 6.04-1 [19.5 kB]
Get:21 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 libhttp-message-perl all 6.22-1 [76.1 kB]
Get:22 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 mecab-utils amd64 0.996-10build1 [4912 B]
Get:23 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 mecab-ipadic all 2.7.0-20070801+main-2.1 [6714 kB]
Get:24 http://azure.archive.ubuntu.com/ubuntu focal/main amd64 mecab-ipadic-utf8 all 2.7.0-20070801+main-2.1 [4380 B]
Get:25 http://azure.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-server all 8.0.39-0ubuntu0.20.04.1 [9480 B]
Fetched 36.9 MB in 43.1 MB/s)
Preconfiguring packages ...
Selecting previously unselected package mysql-common.
(Reading database ... 59063 files and directories currently installed.)
Preparing to unpack .../0-mysql-common_5.8+1.0.5ubuntu2_all.deb ...
```

3. Crear la base de datos "practica_olap":

mysql -u root -p

create database practica_olap;

```
ubuntu@P2-2022630588-dss:~$ sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.39-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.



Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.



mysql> create database practica_olap;
Query OK, 1 row affected (0.02 sec)

mysql>
```

4. Descargar el archivo "practica_olap.sql" de la plataforma Moodle.

Nombre	
	practica_olap.sql
	sales_data.csv

5. Descargar el archivo "sales_data.csv" de la plataforma Moodle.

Nombre	
	practica_olap.sql
	sales_data.csv

6. Crear las tablas ejecutando el script "practica_olap.sql" en la línea de comandos:

```
mysql -u root -p practica_olap < practica_olap.sql
```

```
ubuntu@P2-2022630588-dss:~$ sudo mysql -u root -p practica_olap < practica_olap.sql
Enter password:
ubuntu@P2-2022630588-dss:~$
```

7. Ejecutar el monitor de mysql incluyendo la opción "local-infile" para habilitar la carga de datos:

mysql --local-infile=1 -u root -p practica_olap

```
ubuntu@P2-2022630588-dss:~$ mysql --local-infile=1 -u root -p practica_olap
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 21
Server version: 8.0.39-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

8. Cargar los datos a la tabla "sales_data", ejecutando los siguientes comandos en el monitor de mysql:

```
SET GLOBAL local_infile = 'ON';
load data local infile '/Users/brandondiaz/Desktop/SD_practica2/sales_data.csv' into
table sales_data
fields terminated by ',' enclosed by '"' lines terminated by '\n'
ignore 1 lines
(sales,@order_date,product,customer,country,region,employee,category,weekday,mo
nth,quarter,semester)
SET order_date=STR_TO_DATE(@order_date,'%Y-%m-%d');
Donde "ruta-archivo-sales_data.csv" es la ruta absoluta del archivo "sales_data.csv",
descargado de la plataforma Moodle.
```

```
mysql> SET GLOBAL local_infile = 'ON';
Query OK, 0 rows affected (0.00 sec)

mysql> LOAD DATA LOCAL INFILE '/home/ubuntu/sales_data.csv'
-> INTO TABLE sales_data
-> FIELDS TERMINATED BY ',' ENCLOSED BY '"'
-> LINES TERMINATED BY '\n'
-> IGNORE 1 LINES
-> (sales, @order_date, product, customer, country, region, employee, category, weekday, month, quarter, semester)
-> SET order_date = STR_TO_DATE(@order_date, '%Y-%m-%d');
Query OK, 10000 rows affected (0.48 sec)
Records: 10000 Deleted: 0 Skipped: 0 Warnings: 0
```

10. Cargar las tablas de dimensiones y la fact table a partir de los datos de la tabla "sales_data".

```
mysql> INSERT INTO country (country, id_region)
-> SELECT DISTINCT s.country, r.id_region
-> FROM sales_data s
-> JOIN region r ON s.region = r.region
-> WHERE s.country IS NOT NULL;
weekday IS NOT NULL;

-- Fecha de pedido
INSERT INTO order_date (order_date, id_weekday, id_month)
SELECT DISTINCT s.order_date, wd.id_weekday, m.id_month
FROM sales_data s
JOIN weekday wd ON s.weekday = wd.weekday
JOIN month m ON s.month = m.month;

-- Producto
INSERT INTO product (product)
SELECT DISTINCT product FROM sales_data WHERE product IS NOT NULL;

-- Categoría
INSERT INTO category (category)
SELECT DISTINCT category FROM sales_data WHERE category IS NOT NULL;
Query OK, 16 rows affected (0.03 sec)
Records: 16 Duplicates: 0 Warnings: 0

mysql>
mysql> -- Cliente
mysql> INSERT INTO customer (customer)
-> SELECT DISTINCT customer FROM sales_data WHERE customer IS NOT NULL;
Query OK, 15 rows affected (0.03 sec)
Records: 15 Duplicates: 0 Warnings: 0

mysql>
mysql> -- Empleado
mysql> INSERT INTO employee (employee)
-> SELECT DISTINCT employee FROM sales_data WHERE employee IS NOT NULL;
Query OK, 10 rows affected (0.02 sec)
Records: 10 Duplicates: 0 Warnings: 0

mysql>
mysql> -- Semestre
mysql> INSERT INTO semester (semester)
-> SELECT DISTINCT semester FROM sales_data WHERE semester IS NOT NULL;
Query OK, 2 rows affected (0.02 sec)
Records: 2 Duplicates: 0 Warnings: 0

mysql>
mysql> -- Trimestre
mysql> INSERT INTO quarter (quarter, id_semester)
-> SELECT DISTINCT s.quarter, se.id_semester
-> FROM sales_data s

mysql>
mysql> -- Trimestre
mysql> INSERT INTO quarter (quarter, id_semester)
-> SELECT DISTINCT s.quarter, se.id_semester
-> FROM sales_data s
Query OK, 4 rows affected (0.03 sec)
Records: 4 Duplicates: 0 Warnings: 0

mysql>
mysql> -- Mes
mysql> INSERT INTO month (month, id_quarter)
-> SELECT DISTINCT s.month, q.id_quarter
-> FROM sales_data s
-> JOIN quarter q ON s.quarter = q.quarter;
Query OK, 12 rows affected (0.02 sec)
Records: 12 Duplicates: 0 Warnings: 0

mysql>
mysql> -- Día de la semana
mysql> INSERT INTO weekday (weekday)
-> SELECT DISTINCT weekday FROM sales_data WHERE weekday IS NOT NULL;
Query OK, 7 rows affected (0.02 sec)
Records: 7 Duplicates: 0 Warnings: 0

mysql>
mysql> -- Fecha de pedido
mysql> INSERT INTO order_date (order_date, id_weekday, id_month)
-> SELECT DISTINCT s.order_date, wd.id_weekday, m.id_month
-> FROM sales_data s
-> JOIN weekday wd ON s.weekday = wd.weekday
-> JOIN month m ON s.month = m.month;
Query OK, 1571 rows affected (0.16 sec)
Records: 1571 Duplicates: 0 Warnings: 0

mysql>
mysql> -- Producto
mysql> INSERT INTO product (product)
-> SELECT DISTINCT product FROM sales_data WHERE product IS NOT NULL;
Query OK, 51 rows affected (0.04 sec)
Records: 51 Duplicates: 0 Warnings: 0

mysql>
mysql> -- Categoría
mysql> INSERT INTO category (category)
-> SELECT DISTINCT category FROM sales_data WHERE category IS NOT NULL;
Query OK, 24 rows affected (0.02 sec)
Records: 24 Duplicates: 0 Warnings: 0

mysql>
```

```
mysql> DELETE FROM product;
Query OK, 51 rows affected (0.01 sec)

mysql> ALTER TABLE product AUTO_INCREMENT = 1;
Query OK, 0 rows affected (0.03 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> INSERT INTO product (product, id_category)
-> SELECT DISTINCT s.product, c.id_category
-> FROM sales_data s
-> JOIN category c ON s.category = c.category
-> WHERE s.product IS NOT NULL;
Query OK, 51 rows affected (0.03 sec)
Records: 51 Duplicates: 0 Warnings: 0

mysql> select * from product;
+-----+-----+-----+
| id_product | product                | id_category |
+-----+-----+-----+
| 1 | Yoga Mat                | 1 |
| 2 | Notebook Professional   | 2 |
| 3 | Fitness Tracker         | 3 |
| 4 | Men's Sneakers          | 4 |
| 5 | Smart Watch             | 3 |
| 6 | Tablet Alpha            | 5 |
| 7 | Desktop Alpha           | 2 |
| 8 | Streaming Media Player  | 6 |
| 9 | Desktop Beta            | 2 |
| 10 | Sculpture               | 7 |
```



```
mysql> INSERT INTO fact_table (sales, id_order_date, id_product, id_customer, id_country, id_employee)
-> SELECT
->     s.sales,
->     od.id_order_date,
->     p.id_product,
->     c.id_customer,
->     co.id_country,
->     e.id_employee
-> FROM sales_data s
-> JOIN order_date od ON s.order_date = od.order_date
-> JOIN product p ON s.product = p.product
-> JOIN customer c ON s.customer = c.customer
-> JOIN country co ON s.country = co.country
-> JOIN employee e ON s.employee = e.employee;
Query OK, 10000 rows affected (0.98 sec)
Records: 10000 Duplicates: 0 Warnings: 0

mysql>
```

11. Ejecutar el monitor de mysql:

mysql -u root -p practica_olap

```
bye
ubuntu@P2-2022630588-dss:~$ mysql -u root -p practica_olap
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 23
Server version: 8.0.39-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```


12. Ejecutar la siguiente instrucción SELECT para saber qué directorio puede escribir MySQL:

```
SELECT @@secure_file_priv;
```

```
mysql> SELECT @@secure_file_priv;
+-----+
| @@secure_file_priv |
+-----+
| /var/lib/mysql-files/ |
+-----+
1 row in set (0.01 sec)

mysql> █
```

13. Ejecutar la siguiente instrucción SELECT para generar un archivo CSV con id_contry, id_category, id_product y sales:

```
SELECT a.id_country,b.id_category,a.id_product,a.sales
INTO OUTFILE '/var/lib/mysql-files/country_category_product.csv'
FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
LINES TERMINATED BY '\n'
FROM fact_table a,product b
WHERE a.id_product=b.id_product;
```

En este caso "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads" es el directorio que se obtuvo al ejecutar la instrucción SELECT en el paso 12 en Windows.

```
mysql> SELECT a.id_country, b.id_category, a.id_product, a.sales
-> INTO OUTFILE '/var/lib/mysql-files/country_category_product.csv'
-> FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
-> LINES TERMINATED BY '\n'
-> FROM fact_table a
[ -> JOIN product b ON a.id_product = b.id_product;
Query OK, 10000 rows affected (0.02 sec)

mysql> █
```

Instalación y ejecución de Apache Hadoop

14. Ahora vamos a descargar e instalar Apache Hadoop. Ejecutar la siguiente instrucción en la máquina virtual:

wget https://dlcdn.apache.org/hadoop/common/hadoop-3.4.0/hadoop-3.4.0-aarch64.tar.gz

```
ubuntu@P2-2022630588-dss:~$ wget https://dlcdn.apache.org/hadoop/common/hadoop-3.4.0/hadoop-3.4.0-aarch64.tar.gz
--2024-11-08 05:04:00-- https://dlcdn.apache.org/hadoop/common/hadoop-3.4.0/hadoop-3.4.0-aarch64.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42:1644::
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 965757299 (921M) [application/x-gzip]
Saving to: 'hadoop-3.4.0-aarch64.tar.gz'

hadoop-3.4.0-aarch64.tar.gz      100%[=====] 921.02M  88.8MB/s   in 6.2s

2024-11-08 05:04:06 (148 MB/s) - 'hadoop-3.4.0-aarch64.tar.gz' saved [965757299/965757299]

ubuntu@P2-2022630588-dss:~$
```

15. Descomprimir y desempacar el archivo descargado:

gunzip hadoop-3.4.0-aarch64.tar.gz
tar xvf hadoop-3.4.0-aarch64.tar

```
hadoop-3.4.0/libexec/shellprofile.d/hadoop-s3guard.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-gridmix.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-https.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-distcp.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-azure-datalake.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-streaming.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-extras.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-yarn.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-hdfs.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-rumen.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-azure.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-archive-logs.sh
hadoop-3.4.0/libexec/shellprofile.d/hadoop-aws.sh
hadoop-3.4.0/libexec/hadoop-layout.sh.example
hadoop-3.4.0/libexec/yarn-config.cmd
hadoop-3.4.0/README.txt
hadoop-3.4.0/lib/
hadoop-3.4.0/lib/native/
hadoop-3.4.0/lib/native/libhdfs.so.0.0.0
hadoop-3.4.0/lib/native/libhadoop.a
hadoop-3.4.0/lib/native/libhdfspp.so.0.1.0
hadoop-3.4.0/lib/native/libnativetask.so
hadoop-3.4.0/lib/native/libhdfs.so
hadoop-3.4.0/lib/native/libhadoop.so.1.0.0
hadoop-3.4.0/lib/native/libhdfspp.so
hadoop-3.4.0/lib/native/libhdfs.a
hadoop-3.4.0/lib/native/libhadooppipes.a
hadoop-3.4.0/lib/native/examples/
hadoop-3.4.0/lib/native/examples/wordcount-part
hadoop-3.4.0/lib/native/examples/wordcount-nopipe
hadoop-3.4.0/lib/native/examples/pipes-sort
hadoop-3.4.0/lib/native/examples/wordcount-simple
hadoop-3.4.0/lib/native/libhadooputils.a
hadoop-3.4.0/lib/native/libhdfspp.a
hadoop-3.4.0/lib/native/libnativetask.so.1.0.0
hadoop-3.4.0/lib/native/libhadoop.so
hadoop-3.4.0/lib/native/libnativetask.a
hadoop-3.4.0/LICENSE.txt
hadoop-3.4.0/NOTICE.txt
hadoop-3.4.0/bin/
hadoop-3.4.0/bin/test-container-executor
hadoop-3.4.0/bin/hdfs.cmd
hadoop-3.4.0/bin/yarn.cmd
hadoop-3.4.0/bin/mapred
hadoop-3.4.0/bin/container-executor
hadoop-3.4.0/bin/mapred.cmd
hadoop-3.4.0/bin/hdfs
hadoop-3.4.0/bin/yarn
hadoop-3.4.0/bin/hadoop.cmd
hadoop-3.4.0/bin/oom-listener
hadoop-3.4.0/bin/hadoop
ubuntu@P2-2022630588-dss:~$
```

16. Apache Hadoop requiere Java, entonces necesitamos instalar el JDK en la máquina virtual. Instalar el openjdk 16 o mayor.

```
ubuntu@2-2022639586-dss:~$ sudo apt install openjdk-16-jdk -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  adwaita-icon-theme at-spi2-core ca-certificates-java fontconfig fontconfig-config fonts-dejavu-core fonts-dejavu-extra gtk-update-icon-cache hicolor-icon-theme
  humanity-icon-theme java-common libatk-bridge2.0-0 libatk-wrapper-java libatk-wrapper-java-jni libatk1.0-0 libatk1.0-data libatspi2.0-0 libbavahi-client3 libbavahi-common-data
  libbavahi-common3 libcairo-gobject2 libcairo2 libcups2 libdatrie1 libdrm-amdgpu1 libdrm-intel1 libdrm-nouveau2 libdrm-radeon1 libfontconfig1 libfontenc1 libgail-common libgall18
  libgbk-plibuf2.0-0 libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgl7 libgl1 libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgraphite2-3 libgtk2.0-0
  libgtk2.0-bin libgtk2.0-common libharfbuzz0b libice-dev libice6 libjbig0 libjpeg-turbo8 libjpeg8 liblcms2-2 liblvm2 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0
  libpciaccess0 libpcsc-lite1 libplxman-1-0 libpthread-stubs0-dev librsvg2-2 librsvg2-common libsensors-config libsensors5 libsm-dev libsm6 libthai-data libthai0 libtiff5 libvulkan1
  libwayland-client0 libwebp6 libx11-dev libx11-xcb1 libxau-dev libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0 libxcb-present0 libxcb-randr0 libxcb-render0 libxcb-shape0
  libxcb-shm0 libxcb-sync1 libxcb-xfixes0 libxcb1-dev libxcomposite1 libxcursor1 libxdamage1 libxdmcp-dev libxfixes3 libxft2 libx16 libxinerama1 libxkbfile1 libxmu6 libxpm4
  libxrandr2 libxrender1 libxshmfence1 libxt-dev libxt6 libxtst6 libxv1 libxxf86dga1 libxxf86vm1 mesa-vulkan-drivers openjdk-16-jdk-headless openjdk-16-jre openjdk-16-jre-headless
  ubuntu-mono x11-common x11-utils x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
  default-jre cups-common gvfs libice-doc liblcm2-utils pcsd librsvg2-bin lm-sensors libsm-doc libx11-doc libxcb-doc libxt-doc openjdk-16-demo openjdk-16-source visualvm
  libnss-mdns fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei fonts-indic mesa-utils
The following NEW packages will be installed:
  adwaita-icon-theme at-spi2-core ca-certificates-java fontconfig fontconfig-config fonts-dejavu-core fonts-dejavu-extra gtk-update-icon-cache hicolor-icon-theme
  humanity-icon-theme java-common libatk-bridge2.0-0 libatk-wrapper-java libatk-wrapper-java-jni libatk1.0-0 libatk1.0-data libatspi2.0-0 libbavahi-client3 libbavahi-common-data
  libbavahi-common3 libcairo-gobject2 libcairo2 libcups2 libdatrie1 libdrm-amdgpu1 libdrm-intel1 libdrm-nouveau2 libdrm-radeon1 libfontconfig1 libfontenc1 libgail-common libgall18
  libgbk-plibuf2.0-0 libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgl7 libgl1 libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgraphite2-3 libgtk2.0-0
  libgtk2.0-bin libgtk2.0-common libharfbuzz0b libice-dev libice6 libjbig0 libjpeg-turbo8 libjpeg8 liblcms2-2 liblvm2 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0
  libpciaccess0 libpcsc-lite1 libplxman-1-0 libpthread-stubs0-dev librsvg2-2 librsvg2-common libsensors-config libsensors5 libsm-dev libsm6 libthai-data libthai0 libtiff5 libvulkan1
  libwayland-client0 libwebp6 libx11-dev libx11-xcb1 libxau-dev libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0 libxcb-present0 libxcb-randr0 libxcb-render0 libxcb-shape0
  libxcb-shm0 libxcb-sync1 libxcb-xfixes0 libxcb1-dev libxcomposite1 libxcursor1 libxdamage1 libxdmcp-dev libxfixes3 libxft2 libx16 libxinerama1 libxkbfile1 libxmu6 libxpm4
  libxrandr2 libxrender1 libxshmfence1 libxt-dev libxt6 libxtst6 libxv1 libxxf86dga1 libxxf86vm1 mesa-vulkan-drivers openjdk-16-jdk openjdk-16-jdk-headless openjdk-16-jre
  openjdk-16-jre-headless ubuntu-mono x11-common x11-utils x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
```

17. Ahora vamos a editar el archivo "hadoop-3.4.0/etc/hadoop/hadoop-env.sh":

nano hadoop-3.4.0/etc/hadoop/hadoop-env.sh

```
##
## Licensed to the Apache Software Foundation (ASF) under one
## or more contributor license agreements. See the NOTICE file
## distributed with this work for additional information
## regarding copyright ownership. The ASF licenses this file
## to you under the Apache License, Version 2.0 (the
## "License"); you may not use this file except in compliance
## with the License. You may obtain a copy of the License at
##
## http://www.apache.org/licenses/LICENSE-2.0
##
## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.
##
## Set Hadoop-specific environment variables here.
##
## THIS FILE ACTS AS THE MASTER FILE FOR ALL HADOOP PROJECTS.
## SETTINGS HERE WILL BE READ BY ALL HADOOP COMMANDS. THEREFORE,
## ONE CAN USE THIS FILE TO SET YARN, HDFS, AND MAPREDUCE
## CONFIGURATION OPTIONS INSTEAD OF xxx-env.sh.
##
## Precedence rules:
##
## {yarn-env.sh|hdfs-env.sh} > hadoop-env.sh > hard-coded defaults
##
## {YARN_xyz|HDFS_xyz} > HADOOP_xyz > hard-coded defaults
##
## Many of the options here are built from the perspective that users
## may want to provide OVERWRITING values on the command line.
## For example:
##
## JAVA_HOME=/usr/java/testing hdfs dfs -ls
##
## Therefore, the vast majority (BUT NOT ALL!) of these defaults
## are configured for substitution and not append. If append
## is preferable, modify this file accordingly.
##
## Generic settings for HADOOP
##
## Technically, the only required environment variable is JAVA_HOME.
## All others are optional. However, the defaults are probably not
## preferred. Many sites configure these options outside of Hadoop,
## such as in /etc/profile.d
##
hadoop-3.4.0/etc/hadoop/hadoop-env.sh" 434L, 16786C
```

18. Quitar el comentario a "export JAVA_HOME" y asignarle "/usr" (el directorio que contiene el directorio "bin" que a su vez contiene los programas "java" y "javac"):

```
export JAVA_HOME=/usr
```

```
# The java implementation to use. By default, this environment
# variable is REQUIRED on ALL platforms except OS X!
export JAVA_HOME=/usr
```

19. Ahora vamos a crear una aplicación Hadoop que obtenga una tabla agregada a partir del archivo "country_category_product.csv".

19.1 Definimos las siguientes variables de entorno:

```
export JAVA_HOME=/usr
export HADOOP_HOME=/home/usuario/hadoop-3.4.0
export PATH=$PATH:$JAVA_HOME/bin:$HADOOP_HOME/bin
```

Donde usuario es el usuario actual de la máquina virtual.

```
ubuntu@P2-2022630588-dss:~$ export JAVA_HOME=/usr
ubuntu@P2-2022630588-dss:~$ export HADOOP_HOME=/home/ubuntu/hadoop-3.4.0
ubuntu@P2-2022630588-dss:~$ export PATH=$PATH:$JAVA_HOME/bin:$HADOOP_HOME/bin
```

19.2 Creamos un directorio "prueba", y dentro del directorio creamos las clases "AggregationMapper.java", "AggregationReducer.java" y "AggregationDriver.java" que vimos en clase.

```
mkdir prueba
```

```
ubuntu@P2-2022630588-dss:~$ mkdir prueba
```

```
nano prueba/AggregationMapper.java
nano prueba/AggregationReducer.java
nano prueba/AggregationDriver.java
```

```
ubuntu@P2-2022630588-dss:~$ vim prueba/AggregationMapper.java
ubuntu@P2-2022630588-dss:~$ vim prueba/AggregationReducer.java
ubuntu@P2-2022630588-dss:~$ vim prueba/AggregationDriver.java
ubuntu@P2-2022630588-dss:~$
```

```
ubuntu@P2-2022630588-dss:~$ cat prueba/AggregationDriver.java
// Generado por ChatGPT, 2024.
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
public class AggregationDriver {
    public static void main(String[] args) throws Exception {
        if (args.length != 2) {
            System.err.println("Usage: AggregationDriver <input path> <output path>");
            System.exit(-1);
        }
        Configuration conf = new Configuration();
        Job job = Job.getInstance(conf, "Data Aggregation");
        job.setJarByClass(AggregationDriver.class);
        job.setMapperClass(AggregationMapper.class);
        job.setReducerClass(AggregationReducer.class);
        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(DoubleWritable.class);
        FileInputFormat.addInputPath(job, new Path(args[0]));
        FileOutputFormat.setOutputPath(job, new Path(args[1]));
        System.exit(job.waitForCompletion(true) ? 0 : 1);
    }
}
ubuntu@P2-2022630588-dss:~$ cat prueba/AggregationMapper.java
// Generado por ChatGPT, 2024
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.Mapper;
import java.io.IOException;
public class AggregationMapper extends Mapper<LongWritable, Text, Text, DoubleWritable> {
    private Text dimensionKey = new Text();
    private DoubleWritable metricValue = new DoubleWritable();
    @Override
    public void map(LongWritable key, Text value, Context context) throws IOException, InterruptedException {
        String[] parts = value.toString().split(",");
        if (parts.length == 4) {
            String d1 = parts[0].trim();
            String d2 = parts[1].trim();
            String d3 = parts[2].trim();
            double m1 = Double.parseDouble(parts[3].trim());
            dimensionKey.set(d1 + "," + d2 + "," + d3);
            metricValue.set(m1);
            context.write(dimensionKey, metricValue);
        }
    }
}
}
```

```
ubuntu@P2-2022630588-dss:~$ cat prueba/AggregationReducer.java
// Generado por ChatGPT, 2024
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.Reducer;
import java.io.IOException;
public class AggregationReducer extends Reducer<Text, DoubleWritable, Text, DoubleWritable> {
    private DoubleWritable result = new DoubleWritable();
    @Override
    public void reduce(Text key, Iterable<DoubleWritable> values, Context context) throws IOException, InterruptedException {
        double sum = 0.0;
        for (DoubleWritable val : values) {
            sum += val.get();
        }
        result.set(sum);
        context.write(key, result);
    }
}
ubuntu@P2-2022630588-dss:~$
```

19.3 Compilamos las clases y creamos un archivo .jar (aplicación Hadoop):

```
javac -classpath ` $HADOOP_HOME/bin/hadoop classpath ` -d prueba prueba/*.java  
jar -cvf Aggregation.jar -C prueba .
```

```
ubuntu@P2-2022630588-dss:~$ javac -classpath ` $HADOOP_HOME/bin/hadoop classpath ` -d prueba prueba/*.java  
ubuntu@P2-2022630588-dss:~$ jar -cvf Aggregation.jar -C prueba .  
added manifest  
adding: AggregationDriver.class(in = 1558) (out= 855)(deflated 45%)  
adding: AggregationReducer.class(in = 1722) (out= 721)(deflated 58%)  
adding: AggregationMapper.class(in = 2427) (out= 992)(deflated 59%)  
adding: AggregationReducer.java(in = 604) (out= 306)(deflated 49%)  
adding: AggregationMapper.java(in = 885) (out= 402)(deflated 54%)  
adding: AggregationDriver.java(in = 1119) (out= 446)(deflated 60%)  
ubuntu@P2-2022630588-dss:~$
```

19.4 Creamos un directorio donde colocaremos todos los archivos de entrada para nuestra aplicación Hadoop:

```
mkdir prueba/input
```

```
adding: AggregationDriver.java(in = 1119) (out= 446)(deflated 60%)  
ubuntu@P2-2022630588-dss:~$ mkdir prueba/input  
ubuntu@P2-2022630588-dss:~$
```

19.5 Copiamos el archivo "country_category_product.csv" (obtenido en el paso 13) al directorio "prueba/input".

```
ubuntu@P2-2022630588-dss:~$ sudo cp /var/lib/mysql-files/country_category_product.csv prueba/input/  
ubuntu@P2-2022630588-dss:~$
```


19.6 Ejecutamos el job de Hadoop (si el directorio prueba/output ya existe, es necesario eliminarlo antes de ejecutar el job de Hadoop):

hadoop jar Aggregation.jar AggregationDriver prueba/input prueba/output
Entonces podemos ver el resultado del proceso en el directorio "prueba/output", en este caso el resultado es la tabla agregada.

```
ubuntu@P2-2022630588-dss:~$ hadoop jar Aggregation.jar AggregationDriver prueba/input prueba/output
2024-11-08 05:27:20,168 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2024-11-08 05:27:20,562 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2024-11-08 05:27:20,708 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2024-11-08 05:27:20,709 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2024-11-08 05:27:20,819 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
2024-11-08 05:27:20,945 INFO input.FileInputFormat: Total input files to process : 1
2024-11-08 05:27:20,998 INFO mapreduce.JobSubmitter: number of splits:1
2024-11-08 05:27:21,198 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local1908075018_0001
2024-11-08 05:27:21,199 INFO mapreduce.JobSubmitter: Executing with tokens: []
2024-11-08 05:27:21,411 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2024-11-08 05:27:21,412 INFO mapreduce.Job: Running job: job_local1908075018_0001
2024-11-08 05:27:21,418 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2024-11-08 05:27:21,422 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting to FileOutputCommitterFactory
2024-11-08 05:27:21,424 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2024-11-08 05:27:21,424 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2024-11-08 05:27:21,426 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
2024-11-08 05:27:21,485 INFO mapred.LocalJobRunner: Waiting for map tasks
2024-11-08 05:27:21,485 INFO mapred.LocalJobRunner: Starting task: attempt_local1908075018_0001_m_000000_0
2024-11-08 05:27:21,507 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting to FileOutputCommitterFactory
2024-11-08 05:27:21,507 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2024-11-08 05:27:21,507 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2024-11-08 05:27:21,531 INFO mapred.Task: Using ResourceCalculatorProcessTree : [ ]
2024-11-08 05:27:21,535 INFO mapred.MapTask: Processing split: file:/home/ubuntu/prueba/input/country_category_product.csv:0-147286
2024-11-08 05:27:21,598 INFO mapred.MapTask: (EQUATOR) 0 kvt 26214396(104857584)
2024-11-08 05:27:21,598 INFO mapred.MapTask: mapreduce.task.io.sort.ab: 100
2024-11-08 05:27:21,599 INFO mapred.MapTask: soft limit at 83886080
2024-11-08 05:27:21,599 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
2024-11-08 05:27:21,599 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
2024-11-08 05:27:21,602 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
2024-11-08 05:27:21,729 INFO mapred.LocalJobRunner:
2024-11-08 05:27:21,733 INFO mapred.MapTask: Starting flush of map output
2024-11-08 05:27:21,733 INFO mapred.MapTask: Spilling map output
2024-11-08 05:27:21,733 INFO mapred.MapTask: bufstart = 0; bufend = 158421; bufvoid = 104857600
2024-11-08 05:27:21,733 INFO mapred.MapTask: kvstart = 26214396(104857584); kvend = 26174400(104697600); length = 39997/6553600
2024-11-08 05:27:21,817 INFO mapred.MapTask: Finished spill 0
2024-11-08 05:27:21,837 INFO mapred.Task: Task:attempt_local1908075018_0001_m_000000_0 is done. And is in the process of committing
2024-11-08 05:27:21,839 INFO mapred.LocalJobRunner: map
2024-11-08 05:27:21,840 INFO mapred.Task: Task 'attempt_local1908075018_0001_m_000000_0' done.
2024-11-08 05:27:21,852 INFO mapred.Task: Final Counters for attempt_local1908075018_0001_m_000000_0: Counters: 17
  File System Counters
    FILE: Number of bytes read=152349
    FILE: Number of bytes written=897575
    FILE: Number of read operations=0
    FILE: Number of large read operations=0
    FILE: Number of write operations=0
  Map-Reduce Framework
```

20. Ahora se cargará el archivo resultante del proceso MapReduce, a la tabla agregada "country_category_product" de la base de datos "practica_olap":

20.1 Podemos ver que el archivo resultante se llama "part-r-00000" y se encuentra en el directorio "prueba/output". Si desplegamos este archivo veremos que consta de parejas clave-valor separadas por tabulador, por tanto, para poder cargar el archivo a la tabla "country_category_product" debemos cambiar el tabulador por coma (para que sea un archivo CSV), entonces ejecutamos el siguiente comando en la línea de comandos:

sed -i 's/\t/,/g' prueba/output/part-r-00000

```
ubuntu@P2-2022630588-dss:~$ sed -i 's/\t/,/g' prueba/output/part-r-00000
ubuntu@P2-2022630588-dss:~$
```


20.2 Ejecutar el monitor de MySQL incluyendo la opción "local-infile" para habilitar la carga de datos:

mysql --local-infile=1 -u root -p practica_olap

```
ubuntu@P2-2022630588-dss:~$ mysql --local-infile=1 -u root -p practica_olap
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 24
Server version: 8.0.39-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

20.3 Cargar el archivo "part-r-00000", ejecutando los siguientes comandos en el monitor de mysql:

load data local infile 'ruta-absoluta-del-archivo-part-r-00000' into table
country_category_product
fields terminated by ',' enclosed by '"' lines terminated by '\n'
(id_country,id_category,id_product,sales);
Donde "ruta-absoluta-del-archivo-part-r-00000" es la ruta absoluta del archivo part-r-00000.

```
ubuntu@P2-2022630588-dss:~$ mysql --local-infile=1 -u root -p practica_olap
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 24
Server version: 8.0.39-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> LOAD DATA LOCAL INFILE '/home/ubuntu/prueba/output/part-r-00000'
-> INTO TABLE country_category_product
-> FIELDS TERMINATED BY ',' ENCLOSED BY '"' LINES TERMINATED BY '\n'
-> (id_country, id_category, id_product, sales);
Query OK, 816 rows affected, 369 warnings (0.05 sec)
Records: 816 Deleted: 0 Skipped: 0 Warnings: 369

mysql>
```

```
mysql> select * from country_category_product limit 100;
```

id_country	id_category	id_product	sales
1	1	1	7187.64
1	1	11	5791.53
1	10	17	4492.64
1	10	32	7046.43
1	11	19	4471.83
1	11	43	6772.27
1	12	20	7562.84
1	12	30	3522.75
1	13	21	6034.70
1	13	38	9237.73
1	14	22	3958.90
1	15	23	6834.43
1	15	29	4870.26
1	16	24	8542.78
1	16	47	6422.87
1	17	25	6825.29
1	17	31	5292.59
1	18	26	4617.44
1	18	48	11771.10
1	19	27	7298.61
1	19	49	4738.59
1	2	18	4666.79
1	2	2	5489.95
1	2	7	4225.27
1	2	9	8323.50
1	20	33	5760.87
1	20	51	5488.89
1	21	34	5963.80
1	21	39	4578.07
1	21	42	4219.06
1	22	37	6687.66
1	22	41	6457.82
1	23	40	8803.36
1	23	46	6492.03
1	24	50	6391.06
1	3	3	7964.37
1	3	5	5758.16
1	4	13	3750.32
1	4	4	7051.32
1	5	12	5691.43
1	5	16	8909.14
1	5	44	3292.52
1	5	6	4170.47
1	6	36	5102.52
1	6	8	5701.25
1	7	10	3122.80
1	7	35	4201.85

Conclusiones

En esta práctica, se llevó a cabo el procesamiento de datos mediante la creación de una arquitectura OLAP en un entorno de sistemas distribuidos, implementando tanto MySQL para la gestión de datos como Hadoop para el procesamiento de grandes volúmenes de información. La experiencia fue enriquecedora, ya que permitió aplicar conceptos de bases de datos relacionales y su integración con herramientas de procesamiento distribuido como MapReduce.

Primero, se configuró una máquina virtual en Azure, en la cual se instaló MySQL y se cargaron datos desde un archivo CSV para crear una base de datos OLAP estructurada en tablas de dimensiones y una tabla de hechos. Posteriormente, se configuró y ejecutó Apache Hadoop, lo cual permitió realizar operaciones de agregación en los datos mediante una aplicación MapReduce. La transferencia de datos entre MySQL y Hadoop, así como el ajuste de formatos para compatibilidad, ayudaron a comprender los desafíos de manejar datos en un entorno distribuido.

Esta práctica fortaleció las habilidades en administración de bases de datos, manipulación de grandes volúmenes de datos y en el uso de entornos distribuidos para el procesamiento eficiente de información. En conclusión, la combinación de MySQL y Hadoop resultó ser una solución efectiva para implementar una arquitectura de procesamiento de datos en sistemas distribuidos, brindando una perspectiva clara de los beneficios y retos en la administración de sistemas de datos escalables.