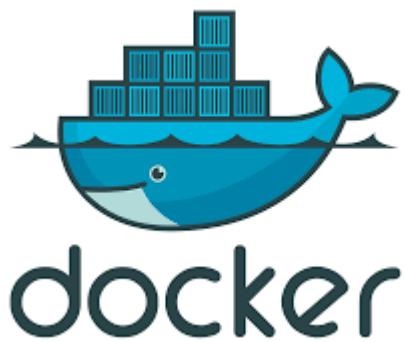




Rapport du projet



Belhouchet Maram
2BD2

Plan

I. Introduction

II. Hadoop sur Docker

Installation de Docker

Installation d'Apache Hadoop

Utilisation de Docker avec Hadoop

Exécution de tâches MapReduce

Captures d'écran du travail réalisé

III. Spark dans Docker

Installation de Spark

Téléchargement de la dataset pour les analyses

Manipulation des données avec Spark

IV. Erreurs et Problèmes Rencontrés

Gestion des conteneurs Docker Hadoop

Erreurs Courantes

Solution pour l'erreur de safemode

I.Introduction:

Ce projet explore l'utilisation de Docker pour créer des environnements conteneurisés, intégrant Apache Hadoop et Spark pour le traitement distribué de données massives. Nous présentons les étapes d'installation, la configuration des clusters, et la réalisation d'analyses de données à l'aide de MapReduce et de Spark. Des solutions aux erreurs rencontrées sont également fournies, offrant ainsi une initiation pratique au Big Data distribué.

II-Hadoop sur docker:

1/installation de docker :

sudo apt update

sudo apt install apt-transport-https ca-certificates curl software-properties-common

description:Cette commande installe les paquets requis pour permettre à Apt d'utiliser des référentiels HTTPS, de gérer les certificats SSL, de télécharger des fichiers via curl, et de gérer les référentiels de logiciels à l'aide de add-apt-repository.

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -

description:Cette commande télécharge la clé GPG officielle de Docker à partir de https://download.docker.com/linux/ubuntu/gpg, la passe à apt-key add - pour l'ajouter à votre trousseau de clés. Cela garantit l'authenticité des paquets Docker téléchargés.

sudo add-apt-repository "deb [arch=amd64]

https://download.docker.com/linux/ubuntu focal stable"

description:Cette commande ajoute le référentiel Docker à la liste des sources de paquets gérées par Apt. Le référentiel est configuré pour prendre en charge l'architecture amd64 (64 bits) et est spécifique à la version focal d'Ubuntu. La branche stable est spécifiée pour installer la version stable de Docker.

sudo apt update

apt-cache policy docker-ce

description:En exécutant cette commande, vous obtiendrez une sortie qui affiche les détails de la politique de paquet pour Docker CE, y compris les versions disponibles, leurs priorités et les dépôts à partir desquels elles sont obtenues. Cela peut être utile pour vérifier quelle version de Docker CE est installée sur votre système et à partir de quel référentiel elle provient.

sudo apt install docker-ce

description:En exécutant cette commande avec les privilèges appropriés, Apt télécharge et installera Docker CE sur votre système. Une fois l'installation terminée, Docker sera prêt à être utilisé.

```
7->7077/tcp, 0.0.0.0:8088->8088/tcp, ::::8088->8088/tcp, 0.0.0.0:9870->9870/tcp, ::::9870->9870/tcp, 0.0.0.0:16010->16010/tcp, ::::16010->16010/tcp, ::::16011->16011/tcp hadoop-master
naram@maram-VirtualBox: ~ docker --version
Docker version 26.0.0, build 2ae903e
naram@maram-VirtualBox: ~
```

2/J'ai installé Apache Hadoop version3.3.6

3/avec l'image de docker uploadée sur dockerhub j'ai la téléchargé :

docker pull liliafxi/hadoop-cluster:latest

4/ créer un réseau qui permettra de relier les trois conteneurs :

```
docker network create --driver=bridge hadoop
```

5/ Créer et lancer les trois conteneurs (les instructions -p permettent de faire un mapping entre les ports de la machine hôte et ceux du conteneur):

```
sudo docker run -itd --net=hadoop -p 9870:9870 -p 8088:8088 -p 7077:7077 -p  
16010:16010 --name hadoop-master --hostname hadoop-master  
liliafaxi/hadoop-cluster:latest
```

```
sudo docker run -itd -p 8040:8042 --net=hadoop --name hadoop-worker1 --hostname  
hadoop-worker1 liliafaxi/hadoop-cluster:latest
```

```
sudo docker run -itd -p 8041:8042 --net=hadoop --name hadoop-worker2 --hostname  
hadoop-worker2 liliafaxi/hadoop-cluster:latest
```

5/vérifier les 3 conteneurs avec docker ps :

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
4a54b372688d	liliafaxi/hadoop-cluster:latest	"sh -c 'service ssh ..."	3 days ago	Up 17 minutes	0.0.0.0:8041->8042/tcp, ::::8041->8042/tcp
e65a0141823b	liliafaxi/hadoop-cluster:latest	"sh -c 'service ssh ..."	3 days ago	Up 17 minutes	0.0.0.0:8040->8042/tcp, ::::8040->8042/tcp
65201398a25c	liliafaxi/hadoop-cluster:latest	"sh -c 'service ssh ..."	3 days ago	Up 18 minutes	0.0.0.0:7077->7077/tcp, ::::7077->7077/tcp, 0.0.0.0:8088->8088/tcp, ::::8088->8088/tcp, 0.0.0.0:9870->9870/tcp, ::::9870->9870/tcp, 0.0.0.0:16010->16010/tcp, ::::16010->16010/tcp

6/exécuter hadoop master , lancer hadoop lancer le job mapreduce après création de mapper et reducer et les mettre dans un répertoire j'ai la nommée mapred et puis le copier vers le master voila les commandes et les captures d'écran sur mon travail:

1/Créer un répertoire dans HDFS, appelé *input*:

```
hdfs dfs -mkdir -p input
```

2/Commencer par décompresser le fichier sur mon machine, puis par le charger dans le conteneur *hadoop-master* avec la commande suivante:

```
docker cp purchases.txt hadoop-master:/root/purchases.txt
```

7/changer directory vers mapred dans lequel j'ai mis les files mapper et reducer:

```
cd mapred
```

6/lancer le mapreduce sur docker :

```
hadoop jar $HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-3.3.6.jar -file  
mapper.py -mapper "python3 mapper.py" -file reducer.py -reducer "python3 reducer.py"  
-input /myinput/purchases.txt -output output33
```

7/captures sur mon travail :

```
maram [En fonction] - Oracle VM VirtualBox
Activities Terminal 19:56 21 آفريل root@hadoop-master: /mapred
maram@maram-VirtualBox: $ sudo docker start hadoop-master
hadoop-master
maram@maram-VirtualBox: $ sudo docker start hadoop-worker1
hadoop-worker1
maram@maram-VirtualBox: $ sudo docker start hadoop-worker2
hadoop-worker2
maram@maram-VirtualBox: $ docker exec -it hadoop-master bash
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.45/containers/hadoop-master/json": dial unix /var/run/docker.sock: connect: permission denied
maram@maram-VirtualBox: $ sudo docker exec -it hadoop-master bash
root@hadoop-master:~# ./start-hadoop.sh

Starting namenodes on [hadoop-master]
hadoop-master: Warning: Permanently added 'hadoop-master' (ED25519) to the list of known hosts.
hadoop-master: WARNING: HADOOP_NAMENODE_OPTS has been replaced by HDFS_NAMENODE_OPTS. Using value of HADOOP_NAMENODE_OPTS.
hadoop-master: namenode is running as process 165. Stop it first and ensure /tmp/hadoop-root-namenode.pid file is empty before retry.
Starting datanodes
WARNING: HADOOP_SECURE_DN_LOG_DIR has been replaced by HADOOP_SECURE_LOG_DIR. Using value of HADOOP_SECURE_DN_LOG_DIR.
hadoop-worker1: Warning: Permanently added 'hadoop-worker1' (ED25519) to the list of known hosts.
hadoop-worker2: Warning: Permanently added 'hadoop-worker2' (ED25519) to the list of known hosts.
hadoop-worker2: WARNING: HADOOP_SECURE_DN_LOG_DIR has been replaced by HADOOP_SECURE_LOG_DIR. Using value of HADOOP_SECURE_DN_LOG_DIR.
hadoop-worker2: WARNING: HADOOP_DATANODE_OPTS has been replaced by HDFS_DATANODE_OPTS. Using value of HADOOP_DATANODE_OPTS.
hadoop-worker1: Warning: Permanently added 'hadoop-worker1' (ED25519) to the list of known hosts.
hadoop-worker1: WARNING: HADOOP_SECURE_DN_LOG_DIR has been replaced by HADOOP_SECURE_LOG_DIR. Using value of HADOOP_SECURE_DN_LOG_DIR.
hadoop-worker1: WARNING: HADOOP_DATANODE_OPTS has been replaced by HDFS_DATANODE_OPTS. Using value of HADOOP_DATANODE_OPTS.
hadoop-worker1: datanode is running as process 73. Stop it first and ensure /tmp/hadoop-root-datanode.pid file is empty before retry.
hadoop-worker2: datanode is running as process 74. Stop it first and ensure /tmp/hadoop-root-datanode.pid file is empty before retry.
Starting secondary namenodes [hadoop-master]
hadoop-master: Warning: Permanently added 'hadoop-master' (ED25519) to the list of known hosts.
hadoop-master: WARNING: HADOOP_SECONDARYNAMENODE_OPTS has been replaced by HDFS_SECONDARYNAMENODE_OPTS. Using value of HADOOP_SECONDARYNAMENODE_OPTS.
hadoop-master: secondarynamenode is running as process 347. Stop it first and ensure /tmp/hadoop-root-secondarynamenode.pid file is empty before retry.
```

```
maram [En fonction] - Oracle VM VirtualBox
Activities Terminal 19:56 21 آفريل root@hadoop-master: /mapred
root@hadoop-master: /mapred
Map input records=4138476
Map output records=4138476
Map output bytes=52880666
Map output materialized bytes=61157630
Input split bytes=198
Combine input records=0
Combine output records=0
Reduce input groups=18
Reduce shuffle bytes=61157630
Reduce input records=4138476
Reduce output records=1
Spilled Records=8276952
Shuffled Maps =2
Failed Shuffles=0
Merged Map outputs=2
GC time elapsed (ms)=2647
CPU time spent (ms)=39660
Physical memory (bytes) snapshot=967933952
Virtual memory (bytes) snapshot=7614590976
Total committed heap usage (bytes)=759693312
Peak Map Physical memory (bytes)=331931648
Peak Map Virtual memory (bytes)=2551861248
Peak Reduce Physical memory (bytes)=337698816
Peak Reduce Virtual memory (bytes)=2553884672
Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
File Input Format Counters
Bytes Read=211317020
File Output Format Counters
Bytes Written=31
2024-04-21 18:42:36,843 INFO streaming.StreamJob: Output directory: output33
root@hadoop-master: /mapred#
```

III-Spark dans Docker:

1/installation de spark:

L'image que j'ai utilisé contient spark sans installation mais je peux l'installer par :

docker pull docker-spark

avec docker-spark est l'image docker.

2/télécharger la dataset pour lancer les analyses :

sudo docker cp purchases.txt hadoop-master:/home/maram/Downloads/sal.csv

description:moi j'ai choisi un dataset sur les salaires des software professional et puis j'ai copié du local au hadoop-master avec la commande :

sudo docker cp purchases.txt hadoop-master:/home/maram/Downloads/sal.csv

3/hdfs dfs –put sal.csv input

description: La commande `hdfs dfs -put sal.csv input` est utilisée pour copier le fichier `sal.csv` du système de fichiers local vers le système de fichiers Hadoop (HDFS), dans le répertoire spécifié `input`.

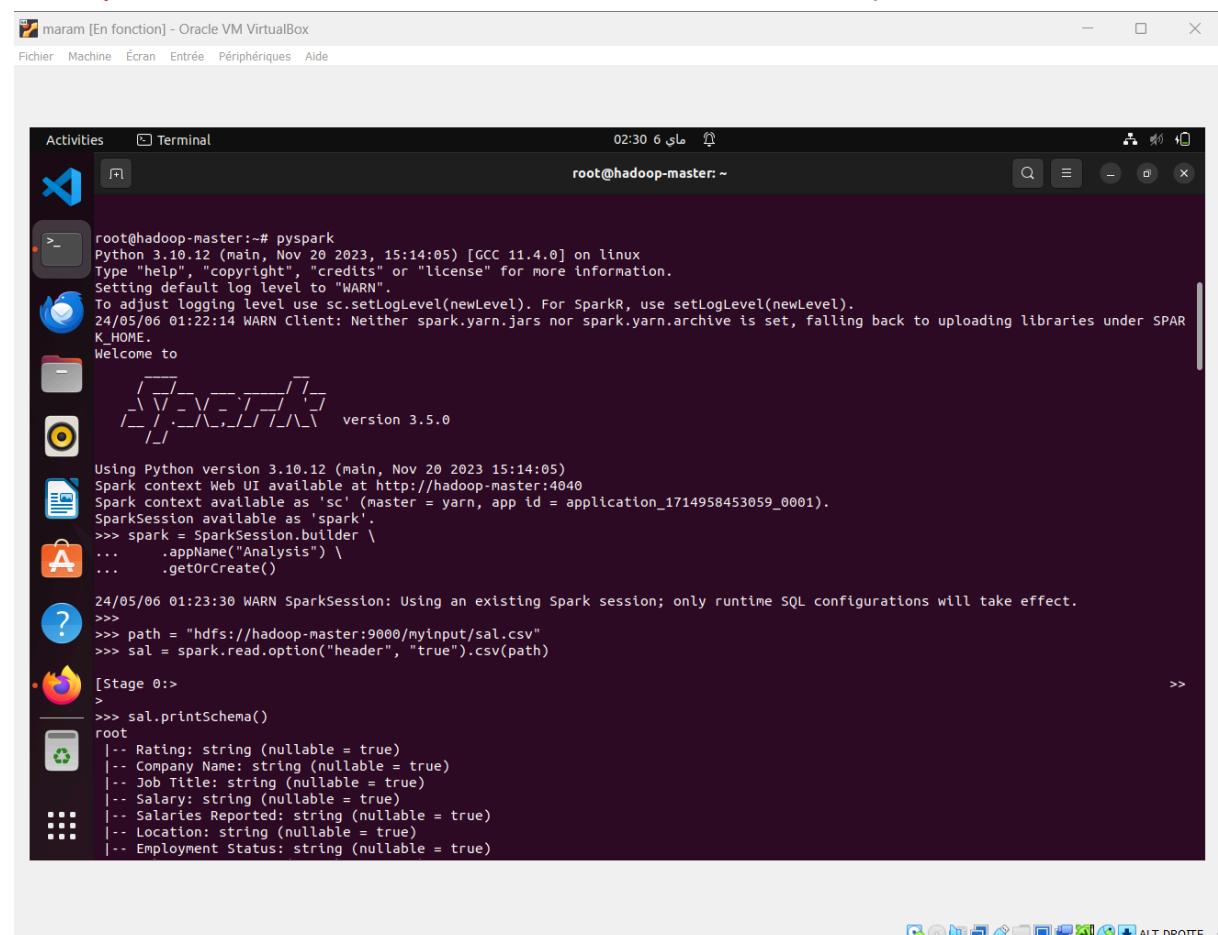
4/hadoop fs -put sal.csv myinput/

description:

La commande `hadoop fs -put sal.csv myinput/` est utilisée pour copier le fichier `sal.csv` du système de fichiers local vers le système de fichiers Hadoop (HDFS), dans le répertoire spécifié `myinput/`.

5/spark-shell

description: exécuter cette commande pour commencer les analyses



```
root@maram [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide

Activities Terminal 02:30 6 مارس 2023 root@hadoop-master: ~

root@hadoop-master:~# pyspark
Python 3.10.12 (main, Nov 20 2023, 15:14:05) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
24/05/06 01:22:14 WARN Client: Neither spark.yarn.jars nor spark.yarn.archive is set, falling back to uploading libraries under SPARK_HOME.
Welcome to
    ____/ \
   / \_ \_ \_ \_ \_ \_ \_ \
  / \_ \_ \_ \_ \_ \_ \_ \_ \
 / \_ \_ \_ \_ \_ \_ \_ \_ \
/ \_ \_ \_ \_ \_ \_ \_ \_ \
Using Python version 3.10.12 (main, Nov 20 2023 15:14:05)
Spark context Web UI available at http://hadoop-master:4040
Spark context available as 'sc' (master = yarn, app id = application_1714958453059_0001).
SparkSession available as 'spark'.
>>> spark = SparkSession.builder \
...     .appName("Analysis") \
...     .getOrCreate()
24/05/06 01:23:30 WARN SparkSession: Using an existing Spark session; only runtime SQL configurations will take effect.
>>>
>>> path = "hdfs://hadoop-master:9000/myinput/sal.csv"
>>> sal = spark.read.option("header", "true").csv(path)
[Stage 0:>
>
>>> sal.printSchema()
root
 |-- Rating: string (nullable = true)
 |-- Company Name: string (nullable = true)
 |-- Job Title: string (nullable = true)
 |-- Salary: string (nullable = true)
 |-- Salaries Reported: string (nullable = true)
 |-- Location: string (nullable = true)
 |-- Employment Status: string (nullable = true)
```

```
6/val path = "hdfs://hadoop-master:9000/myinput/sal.csv"
```

```
val sal = spark.read.option("header", "true").csv(path)
```

description:Ces lignes de code Scala sont utilisées dans un environnement Spark pour lire un fichier CSV à partir du système de fichiers Hadoop (HDFS) et créer un DataFrame à l'aide de Spark.

7/ lancer les analyses sur scala et pyspark (j'ai mis cela dans le fichier code spark)

Code SPARK

1/ code scala:

1. Charger le fichier CSV et afficher le schéma des données :

```
val path = "hdfs://hadoop-master:9000/myinput/sal.csv"
```

```
val sal = spark.read.option("header", "true").csv(path)
```

```
sal.printSchema()
```

description:Cette commande charge le fichier CSV depuis HDFS dans un DataFrame Spark appelé `sal` et affiche le schéma des données, montrant les noms des colonnes et leurs types.

2. Afficher les premières lignes du DataFrame :

sal.show()

description:Cela affiche les premières lignes du DataFrame sal.

```
maram [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
Activities Terminal 02:30 6 જાનુઆરી root@hadoop-master: ~
>_
>>> sal.printSchema()
root
|-- Rating: string (nullable = true)
|-- Company Name: string (nullable = true)
|-- Job Title: string (nullable = true)
|-- Salary: string (nullable = true)
|-- Salaries Reported: string (nullable = true)
|-- Location: string (nullable = true)
|-- Employment Status: string (nullable = true)
|-- Job Roles: string (nullable = true)
>>> sal.show()
+-----+-----+-----+-----+-----+-----+
|Rating| Company Name| Job Title| Salary|Salaries Reported| Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+-----+
| 3.8| Saska...| Android Developer| 400000| 3|Bangalore| Full Time| Android|
| 4.5|Advanced Millenni...| Android Developer| 400000| 3|Bangalore| Full Time| Android|
| 4.0| Unacademy| Android Developer| 1000000| 3|Bangalore| Full Time| Android|
| 3.8| SnapBizz Cloudtech| Android Developer| 300000| 3|Bangalore| Full Time| Android|
| 4.4|Appoids Tech Solu...| Android Developer| 600000| 3|Bangalore| Full Time| Android|
| 4.2| Freelancer| Android Developer| 100000| 3|Bangalore| Full Time| Android|
| 3.7| SQUARE N CUBE| Android Developer| 192000| 3|Bangalore| Full Time| Android|
| 3.1|Samsung R&D Insti...| Android Developer| 400000| 3|Bangalore| Full Time| Android|
| 3.7|DXMinds Technologies| Android Developer| 300000| 3|Bangalore| Full Time| Android|
| 3.6|Endeavour Software...| Android Developer| 600000| 3|Bangalore| Full Time| Android|
| 3.6| Craft Silicon| Android Developer| 300000| 3|Bangalore| Full Time| Android|
| 3.9|Baronford & Assoc...| Android Developer| 240000| 2|Bangalore| Full Time| Android|
| 3.7| Wibmo| Android Developer| 900000| 2|Bangalore| Full Time| Android|
| 4.8| Retail Pulse|Android Developer...| 24000| 2|Bangalore| Intern| Android|
| 3.9| Bookmyshow| Android Developer| 600000| 2|Bangalore| Full Time| Android|
| 3.9| Knowledge Flex| Android Developer| 228000| 2|Bangalore| Full Time| Android|
| 3.6| Novopay Solutions| Android Developer| 600000| 2|Bangalore| Full Time| Android|
| 3.7| WealthEngine| Android Developer| 360000| 2|Bangalore| Full Time| Android|
| 4.0| J.P. Morgan| Android Developer| 1000000| 2|Bangalore| Full Time| Android|
| 3.6| Acviss| Android Developer| 500000| 2|Bangalore| Full Time| Android|
+-----+
only showing top 20 rows
```

3. Compter le nombre total de lignes dans le DataFrame :

sal.count()

description:Cette commande compte le nombre total de lignes dans le DataFrame

sal.

```

Activities Terminal 02:30 6 juil root@hadoop-master:-
Fichier Machine Ecran Entrée Périphériques Aide

>>> val sal = spark.read.csv("hdfs://hadoop-master:9000/datasets/salaries.csv")
>>> sal.show(20)
+-----+-----+-----+-----+-----+
|Rating|Company Name|Job Title|Salary|Salaries Reported|Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+
| 3.8|Sasken|Android Developer|400000|      3|Bangalore| Full Time| Android|
| 4.5|Advanced Millenni...|Android Developer|400000|      3|Bangalore| Full Time| Android|
| 4.1|Academy|Android Developer|100000|      3|Bangalore| Full Time| Android|
| 3.8|Snapspliz Cloudtech|Android Developer|400000|      3|Bangalore| Full Time| Android|
| 4.4|Appolds Tech Solu...|Android Developer|600000|      3|Bangalore| Full Time| Android|
| 3.7|SQUARE N CUBE|Android Developer|192000|      3|Bangalore| Full Time| Android|
| 3.1|Samsung R&D Insti...|Android Developer|400000|      3|Bangalore| Full Time| Android|
| 3.7|DXMinds Technologies|Android Developer|300000|      3|Bangalore| Full Time| Android|
| 3.6|Endeavour Software|Android Developer|600000|      3|Bangalore| Full Time| Android|
| 3.6|Craft Silicon|Android Developer|300000|      3|Bangalore| Full Time| Android|
+-----+-----+-----+-----+-----+
only showing top 20 rows
>>> sal.count()
[Stage 2:>
770
>>> sal.describe("Salary", "Salaries Reported").show()
[Stage 5:>
[Stage 7:>
+-----+-----+-----+
|summary|Salary|Salaries Reported|
+-----+-----+-----+
| count| 22770| 22770|
| mean|609387.211438634|1.855775142731664|
| stddev|884399.0136761835| 6.823668180058269|
| min| 100000| 1|
| max| 996000| 98|
+-----+-----+-----+
>>> high_salaries = sal.filter(sal["Salary"] > 100000)
>>> high_salaries.show()
+-----+-----+-----+-----+-----+
|Rating|Company Name|Job Title|Salary|Salaries Reported|Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+
| 3.8|Sasken|Android Developer|400000|      3|Bangalore| Full Time| Android|
| 4.5|Advanced Millenni...|Android Developer|400000|      3|Bangalore| Full Time| Android|
| 4.1|Academy|Android Developer|100000|      3|Bangalore| Full Time| Android|
| 3.8|Snapspliz Cloudtech|Android Developer|400000|      3|Bangalore| Full Time| Android|
| 4.4|Appolds Tech Solu...|Android Developer|600000|      3|Bangalore| Full Time| Android|
| 3.7|SQUARE N CUBE|Android Developer|192000|      3|Bangalore| Full Time| Android|
| 3.1|Samsung R&D Insti...|Android Developer|400000|      3|Bangalore| Full Time| Android|
| 3.7|DXMinds Technologies|Android Developer|300000|      3|Bangalore| Full Time| Android|
| 3.6|Endeavour Software|Android Developer|600000|      3|Bangalore| Full Time| Android|
| 3.6|Craft Silicon|Android Developer|300000|      3|Bangalore| Full Time| Android|
+-----+-----+-----+-----+

```

4. Afficher des statistiques descriptives pour les colonnes "Salary" et "Salaries Reported" :

`sal.describe("Salary", "Salaries Reported").show()`

description:Cela affiche des statistiques descriptives telles que la moyenne, l'écart type, le minimum, le maximum, etc., pour les colonnes spécifiées.

5. Filtrer les lignes avec un salaire supérieur à 100 000 :

`val highSalaries = sal.filter($"Salary" > 100000)`

`highSalaries.show()`

description:Cela filtre les lignes du DataFrame où la valeur de la colonne "Salary" est supérieure à 100 000 et affiche le résultat.

```

Activities Terminal 02:31 6 ↗ root@hadoop-master: ~
>_
>>> min|      100000|      1|
>>> max|      996000|     98|
>>>
>>> high_salaries = sal.filter(sal["Salary"] > 100000)
>>> high_salaries.show()
+-----+-----+-----+-----+-----+-----+
|Rating| Company Name| Job Title| Salary|Salaries Reported| Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+-----+
| 3.8| Sasken|Android Developer| 400000|      3|Bangalore| Full Time| Android|
| 4.5|Advanced Millenni...|Android Developer| 400000|      3|Bangalore| Full Time| Android|
| 4.0| Unacademy|Android Developer|1000000|      3|Bangalore| Full Time| Android|
| 3.8| SnapBizz Cloudtech|Android Developer| 300000|      3|Bangalore| Full Time| Android|
| 4.4|Appoids Tech Solu...|Android Developer| 600000|      3|Bangalore| Full Time| Android|
| 3.7| SQUARE N CUBE|Android Developer| 192000|      3|Bangalore| Full Time| Android|
| 3.1|Samsung R&D Insti...|Android Developer| 400000|      3|Bangalore| Full Time| Android|
| 3.7|DXMinds Technologies|Android Developer| 300000|      3|Bangalore| Full Time| Android|
| 3.6|Endeavour Softwar...|Android Developer| 600000|      3|Bangalore| Full Time| Android|
| 3.0| Craft Silicon|Android Developer| 300000|      3|Bangalore| Full Time| Android|
| 3.9|Baronford & Assoc...|Android Developer| 240000|      2|Bangalore| Full Time| Android|
| 3.7| Wibmo|Android Developer| 900000|      2|Bangalore| Full Time| Android|
| 3.9| Bookmyshow|Android Developer| 600000|      2|Bangalore| Full Time| Android|
| 3.9| Knowledge Flex|Android Developer| 228000|      2|Bangalore| Full Time| Android|
| 3.6| Novopay Solutions|Android Developer| 600000|      2|Bangalore| Full Time| Android|
| 3.7| WealthEngine|Android Developer| 360000|      2|Bangalore| Full Time| Android|
| 4.0| J.P. Morgan|Android Developer|1000000|      2|Bangalore| Full Time| Android|
| 3.6| Acvils|Android Developer| 500000|      2|Bangalore| Full Time| Android|
| 4.1| Fresher|Android Developer| 408000|      2|Bangalore| Full Time| Android|
| 4.2| MedOnGo|Android Developer| 300000|      2|Bangalore| Full Time| Android|
+-----+
only showing top 20 rows
>>> sorted_by_salary = sal.orderBy(sal["Salary"].desc())
>>> sorted_by_salary.show()
[Stage 9:>
+-----+-----+-----+-----+-----+-----+
|Rating| Company Name| Job Title|Salary|Salaries Reported| Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+-----+
| 3.9| ennVee TechnoGroup|Oracle Applicatio...|996000|      1|Bangalore| Full Time| Database|
+-----+

```

6. Calculer le salaire moyen par entreprise :

```

val avgSalaryByCompany = sal.groupBy("Company Name").avg("Salary")
avgSalaryByCompany.show()

```

```

maram [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Pérophériques Aide

Activities Terminal 02:10 ٦ مارس root@hadoop-master: ~
scala> avgSalaryByCompany.show()
+-----+-----+
| Company Name| avg(Salary)|
+-----+-----+
| Coder| 398000.0|
| Launchers World S...| 192000.0|
| Mobile Apps Company| 200000.0|
| Aequalis| 516000.0|
| Digital Minds (In...| 408000.0|
| YumzyX| 600000.0|
| Newfold Digital| 924888.888888889|
| Fyp| 1200000.0|
| Grey Coconut Designs| 300000.0|
| Dr. Reddy's| 500000.0|
| Intertec Systems| 588000.0|
| Think201| 120000.0|
| AskGalore Digital| 120000.0|
| IBI Group| 700000.0|
| nagesh patil| 300000.0|
| Appgram Technologies| 100000.0|
| GoalsR| 216000.0|
| Fresher Zones| 308615.3846153846|
| Teachers College| 228000.0|
| Blue Ribbon| 168000.0|
+-----+
only showing top 20 rows

scala> val sortedBySalary = sal.orderBy($"Salary".desc)
sortedBySalary: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [Rating: string, Company Name: string ... 6 more fields]

scala> sortedBySalary.show()
+-----+-----+-----+-----+-----+-----+-----+
|Rating| Company Name| Job Title|Salary|Salaries Reported| Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+-----+-----+
| 3.9| ennVee TechnoGroup|Oracle Application|996000| 1|Bangalore| Full Time| Database|
| 4.2| MakeMyTrip|Software Developm...|996000| 4|Bangalore| Full Time| SDE|
| 4.3| Esper| Android Engineer|996000| 1|Bangalore| Full Time| Android|
| 4.3| Iris Software|Senior Android De...|996000| 1|New Delhi| Contractor| Android|
+-----+

```

description:Cela regroupe les données par entreprise et calcule la moyenne du salaire pour chaque entreprise, puis affiche le résultat.

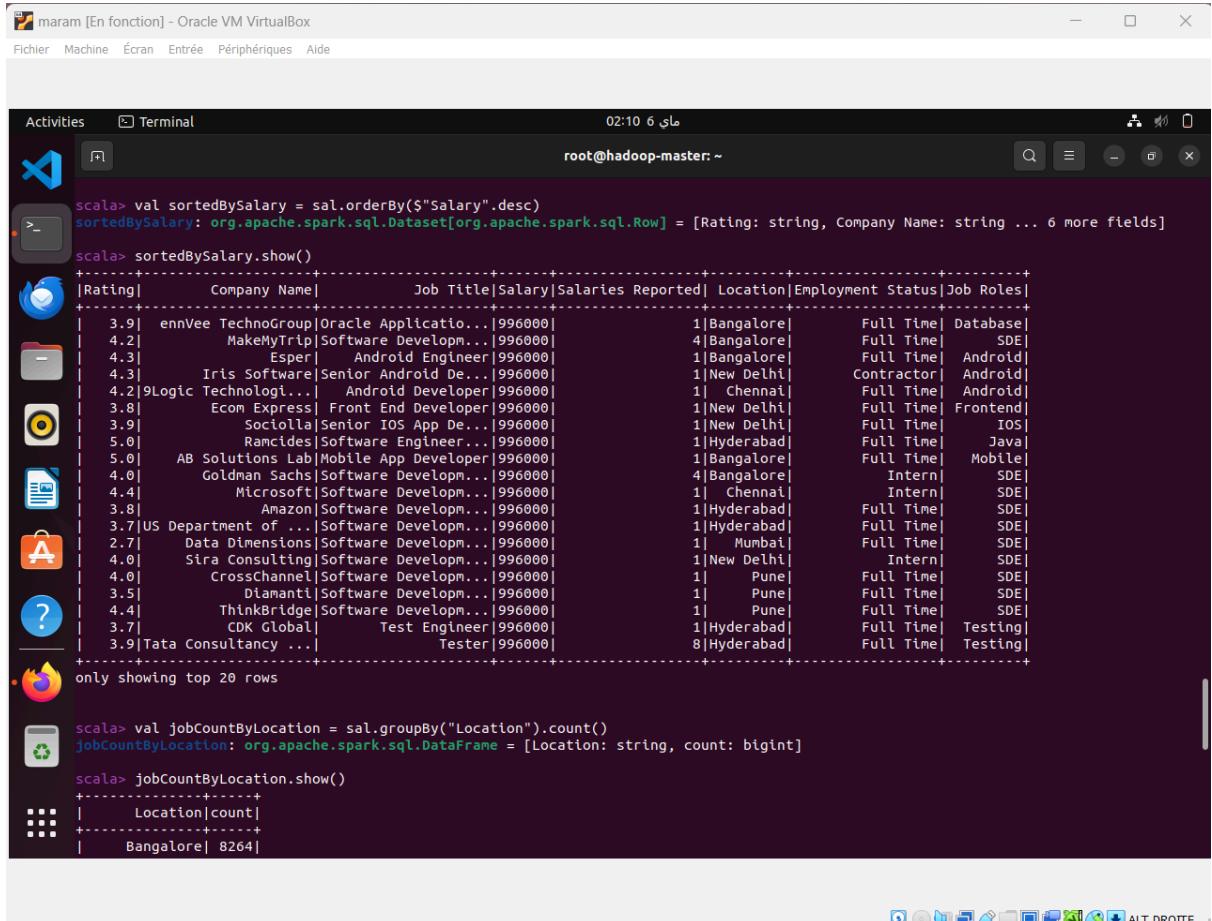
7. Trier le DataFrame par salaire décroissant :

```

val sortedBySalary = sal.orderBy($"Salary".desc)
sortedBySalary.show()

```

description:Cela trie le DataFrame en fonction de la colonne "Salary" de manière décroissante et affiche le résultat.



The screenshot shows a terminal window titled "maram [En fonction] - Oracle VM VirtualBox". The window has a menu bar with "Fichier", "Machine", "Écran", "Entrée", "Périphériques", and "Aide". The title bar shows the time as "02:10 6 ماي". The terminal window has tabs for "Activities" and "Terminal". The command prompt is "root@hadoop-master: ~". The Scala code runs on this terminal:

```
scala> val sortedBySalary = sal.orderBy($"Salary".desc)
sortedBySalary: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [Rating: string, Company Name: string ... 6 more fields]

scala> sortedBySalary.show()
+-----+-----+-----+-----+-----+
|Rating|Company Name|Job Title|Salary|Salaries Reported|Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+
| 3.9| ennVee TechnoGroup|Oracle Applicatio...|996000| 1|Bangalore| Full Time| Database|
| 4.2| MakeMyTrip|Software Develop...|996000| 4|Bangalore| Full Time| SDE|
| 4.3| Esper| Android Engineer|996000| 1|Bangalore| Full Time| Android|
| 4.3| Iris Software|Senior Android De...|996000| 1|New Delhi| Contractor| Android|
| 4.2| 9Logic Technologi...| Android Developer|996000| 1| Chennai| Full Time| Android|
| 3.8| Ecom Express| Front End Developer|996000| 1|New Delhi| Full Time| Frontend|
| 3.9| Sociolla|Senior IOS App De...|996000| 1|New Delhi| Full Time| IOS|
| 5.0| Ramcides|Software Engineer...|996000| 1|Hyderabad| Full Time| Java|
| 5.0| AB Solutions Lab|Mobile App Developer|996000| 1|Bangalore| Full Time| Mobile|
| 4.0| Goldman Sachs|Software Developm...|996000| 4|Bangalore| Intern| SDE|
| 4.4| Microsoft|Software Developm...|996000| 1| Chennai| Intern| SDE|
| 3.8| Amazon|Software Developm...|996000| 1|Hyderabad| Full Time| SDE|
| 3.7| US Department of ...|Software Developm...|996000| 1|Hyderabad| Full Time| SDE|
| 2.7| Data Dimensions|Software Developm...|996000| 1|Mumbai| Full Time| SDE|
| 4.0| Sira Consulting|Software Developm...|996000| 1|New Delhi| Intern| SDE|
| 4.0| CrossChannel|Software Developm...|996000| 1| Pune| Full Time| SDE|
| 3.5| Dlamanti|Software Developm...|996000| 1| Pune| Full Time| SDE|
| 4.4| ThinkBridge|Software Developm...|996000| 1| Pune| Full Time| SDE|
| 3.7| CDK Global| Test Engineer|996000| 1|Hyderabad| Full Time| Testing|
| 3.9| Tata Consultancy ...| Tester|996000| 8|Hyderabad| Full Time| Testing|
+-----+-----+-----+-----+-----+
only showing top 20 rows

scala> val jobCountByLocation = sal.groupBy("Location").count()
jobCountByLocation: org.apache.spark.sql.DataFrame = [Location: string, count: bigint]

scala> jobCountByLocation.show()
+-----+-----+
| Location|count|
+-----+-----+
| Bangalore| 8264|
+-----+-----+
```

8. Compter le nombre d'emplois par emplacement :

```
val jobCountByLocation = sal.groupBy("Location").count()
jobCountByLocation.show()
```

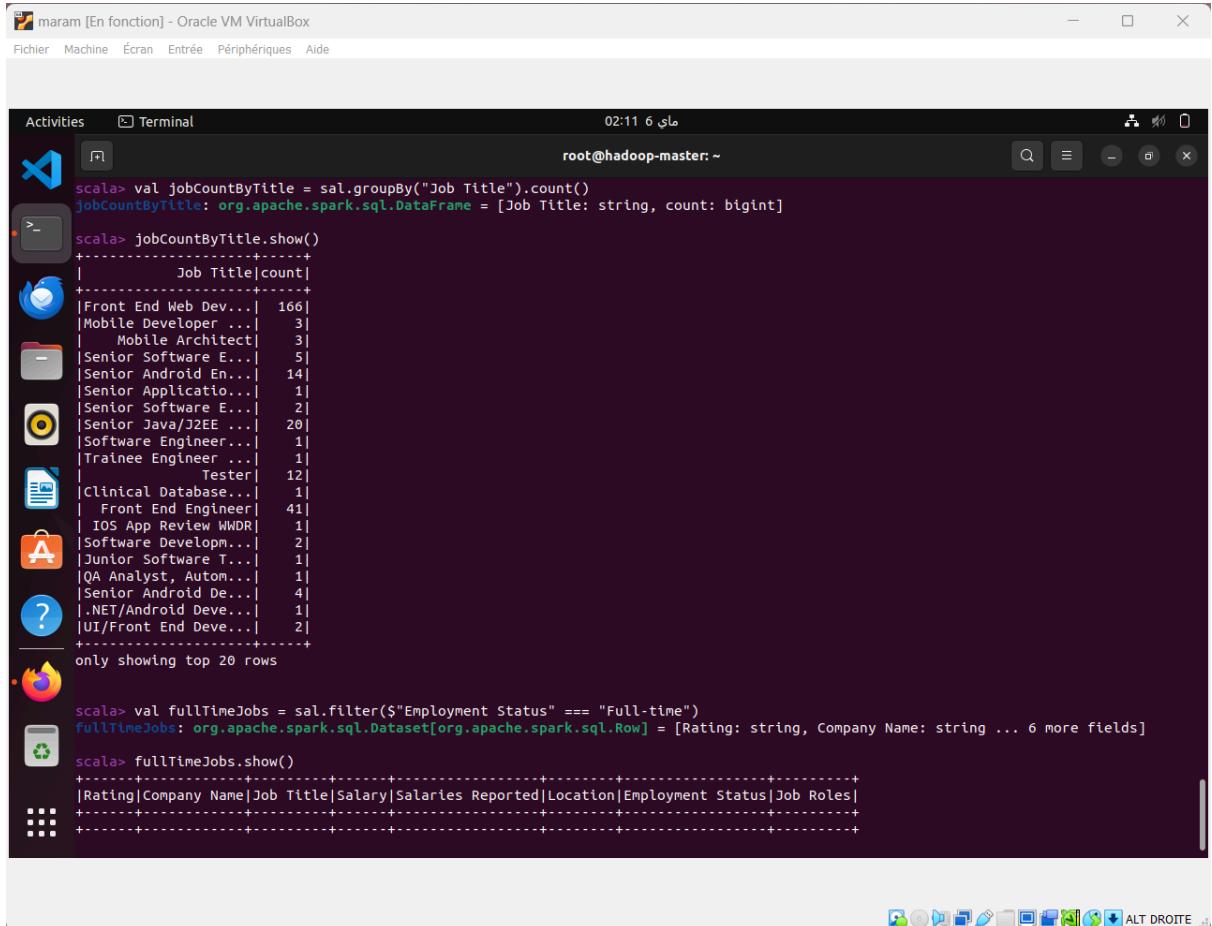
description:Cela regroupe les données par emplacement et compte le nombre d'emplois pour chaque emplacement, puis affiche le résultat.

```
Activities Terminal 02:11 6 آی root@hadoop-master: ~
+-----+
only showing top 20 rows
+-----+
scala> val jobCountByLocation = sal.groupBy("Location").count()
jobCountByLocation: org.apache.spark.sql.DataFrame = [Location: string, count: bigint]
+-----+
| Location|count|
+-----+
| Bangalore| 8264|
| Kerala| 108|
| Madhya Pradesh| 155|
| Chennai| 2458|
| Mumbai| 749|
| Kolkata| 178|
| Pune| 2134|
| New Delhi| 4176|
| Hyderabad| 4467|
| Jaipur| 81|
+-----+
scala> val jobCountByTitle = sal.groupBy("Job Title").count()
jobCountByTitle: org.apache.spark.sql.DataFrame = [Job Title: string, count: bigint]
+-----+
| Job Title|count|
+-----+
|Front End Web Dev...| 166|
|Mobile Developer ...| 3|
| Mobile Architect| 3|
|Senior Software E...| 5|
|Senior Android En...| 14|
|Senior Application...| 1|
|Senior Software E...| 2|
|Senior Java/J2EE ...| 20|
|Software Engineer...| 1|
+-----+
```

9. Compter le nombre d'emplois par titre de poste :

```
val jobCountByTitle = sal.groupBy("Job Title").count()
jobCountByTitle.show()
```

description:Cela regroupe les données par titre de poste et compte le nombre d'emplois pour chaque titre de poste, puis affiche le résultat.



The screenshot shows a terminal window titled "Activities Terminal" on a Linux desktop. The window title bar includes "maram [En fonction] - Oracle VM VirtualBox" and "Fichier Machine Écran Entrée Pérophériques Aide". The terminal window has a dark background and displays the following Scala code and its output:

```
scala> val jobCountByTitle = sal.groupBy("Job Title").count()
jobCountByTitle: org.apache.spark.sql.DataFrame = [Job Title: string, count: bigint]

scala> jobCountByTitle.show()
+-----+-----+
|      Job Title|count|
+-----+-----+
|Front End Web Dev...| 166|
|Mobile Developer ...|   3|
|  Mobile Architect|   3|
|Senior Software E...|   5|
|Senior Android En...|  14|
|Senior Application...|   1|
|Senior Software E...|   2|
|Senior Java/J2EE ...|  20|
|Software Engineer...|   1|
|Trainee Engineer ...|   1|
|        Tester|  12|
|Clinical Database...|   1|
|  Front End Engineer|  41|
|IOS App Review WWDR|   1|
|Software Developm...|   2|
|Junior Software T...|   1|
|QA Analyst, Autom...|   1|
|Senior Android De...|   4|
|.NET/Android Deve...|   1|
|UI/Front End Deve...|   2|
+-----+-----+
only showing top 20 rows

scala> val fullTimeJobs = sal.filter($"Employment Status" === "Full-time")
fullTimeJobs: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [Rating: string, Company Name: string ... 6 more fields]

scala> fullTimeJobs.show()
+-----+-----+-----+-----+-----+-----+
|Rating|Company Name|Job Title|Salary|Salaries Reported|Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+-----+
|.....|.....|.....|.....|.....|.....|.....|.....|
```

10. Filtrer les emplois à temps plein :

```
val fullTimeJobs = sal.filter($"Employment Status" === "Full-time")
fullTimeJobs.show()
```

description:Cela filtre les emplois à temps plein en sélectionnant les lignes où la valeur de la colonne "Employment Status" est "Full-time", puis affiche le résultat.

```
scala> val jobCountByTitle = sal.groupBy("Job Title").count()
jobCountByTitle: org.apache.spark.sql.DataFrame = [Job Title: string, count: bigint]

scala> jobCountByTitle.show()
+-----+-----+
|      Job Title|count|
+-----+-----+
|Front End Web Dev...| 166|
|Mobile Developer ...|   3|
|  Mobile Architect|   3|
|Senior Software E...|   5|
|Senior Android En...|  14|
|Senior Application...|   1|
|Senior Software E...|   2|
|Senior Java/J2EE ...|  20|
|Software Engineer...|   1|
|Trainee Engineer ...|   1|
|        Tester|  12|
|Clinical Database...|   1|
| Front End Engineer|  41|
| IOS App Review WWDR|   1|
|Software Developmen...|   2|
|Junior Software T...|   1|
|QA Analyst, Autom...|   1|
|Senior Android De...|   4|
|.NET/Android Deve...|   1|
|UI/Front End Deve...|   2|
+-----+-----+
only showing top 20 rows

scala> val fullTimeJobs = sal.filter($"Employment Status" === "Full-time")
fullTimeJobs: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [Rating: string, Company Name: string ... 6 more fields]

scala> fullTimeJobs.show()
+-----+-----+-----+-----+-----+-----+
|Rating|Company Name|Job Title|Salary|Salaries Reported|Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+
```

1/ code pyspark:

1. Créer une session Spark et charger le fichier CSV :

```
from pyspark.sql import SparkSession
spark = SparkSession.builder \
    .appName("Analysis") \
    .getOrCreate()
path = "hdfs://hadoop-master:9000/myinput/sal.csv"
sal = spark.read.option("header", "true").csv(path)
```

description : Cela crée une session Spark et charge le fichier CSV depuis HDFS dans un DataFrame Spark appelé `sal`.

The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "root@hadoop-master:~". The terminal content shows a Python script being run to read a CSV file from HDFS:

```
root@hadoop-master:~# pyspark
Python 3.10.12 (main, Nov 20 2023, 15:14:05) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
24/05/06 01:22:14 WARN Client: Neither spark.yarn.jars nor spark.yarn.archive is set, falling back to uploading libraries under SPAR
K_HOME.
Welcome to
   ____          _ _ _ _ 
  / \ \ \ \ \ \ \ \ \ \ \ \ \ \ 
 /_ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ 
 /_ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ 
 /_ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ 
version 3.5.0

Using Python version 3.10.12 (main, Nov 20 2023 15:14:05)
Spark context Web UI available at http://hadoop-master:0@40
Spark context available as 'sc' (master = yarn, app id = application_1714958453059_0001).
SparkSession available as 'spark'.
>>> spark = SparkSession.builder \
...     .appName("Analysis") \
...     .getOrCreate()

24/05/06 01:23:30 WARN SparkSession: Using an existing Spark session; only runtime SQL configurations will take effect.

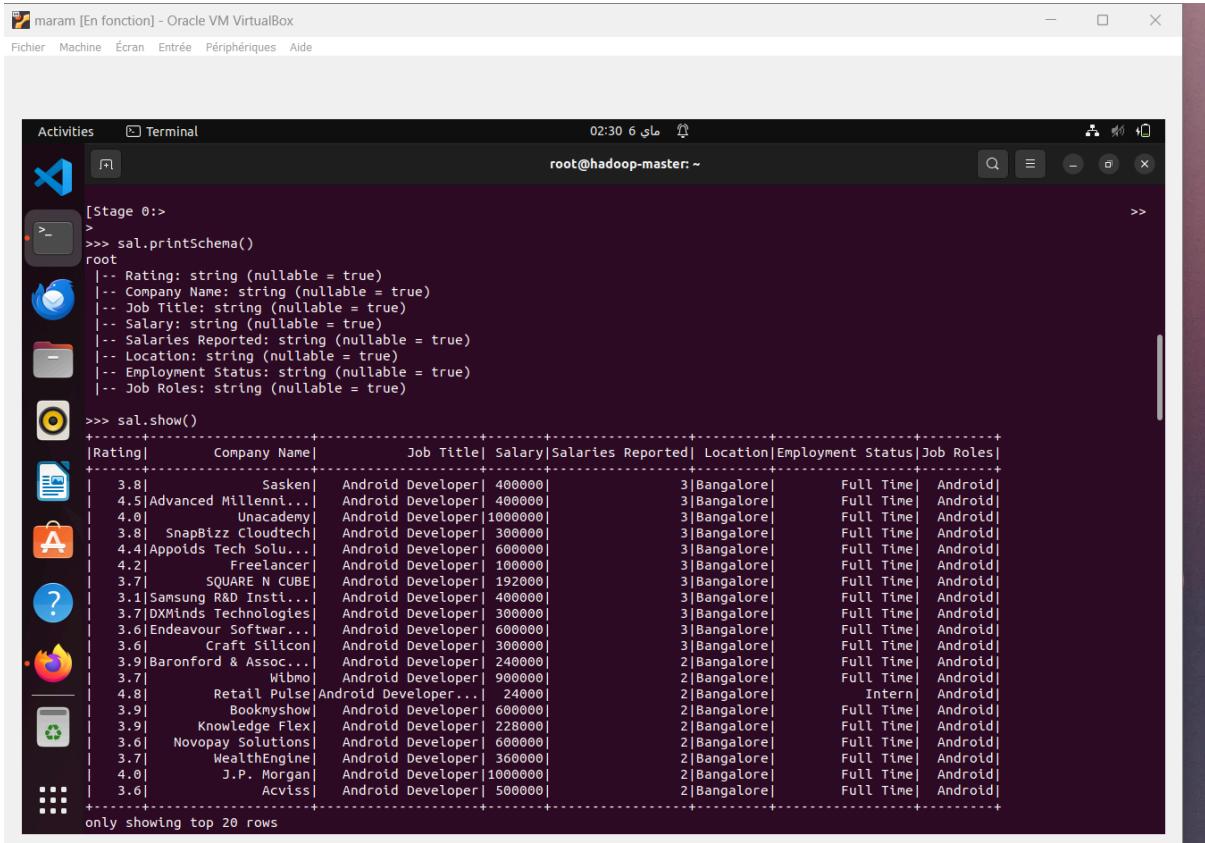
??
>>> path = "hdfs://hadoop-master:9000/myinput/sal.csv"
>>> sal = spark.read.option("header", "true").csv(path)

[Stage 0:>
>
>>> sal.printSchema()
root
 |-- Rating: string (nullable = true)
 |-- Company Name: string (nullable = true)
 |-- Job Title: string (nullable = true)
 |-- Salary: string (nullable = true)
 |-- Salaries Reported: string (nullable = true)
 |-- Location: string (nullable = true)
 |-- Employment Status: string (nullable = true)
```

2. Afficher le schéma des données :

```
sal.printSchema()
```

description: Cette commande affiche le schéma des données du DataFrame `sal`, montrant les noms des colonnes et leurs types.

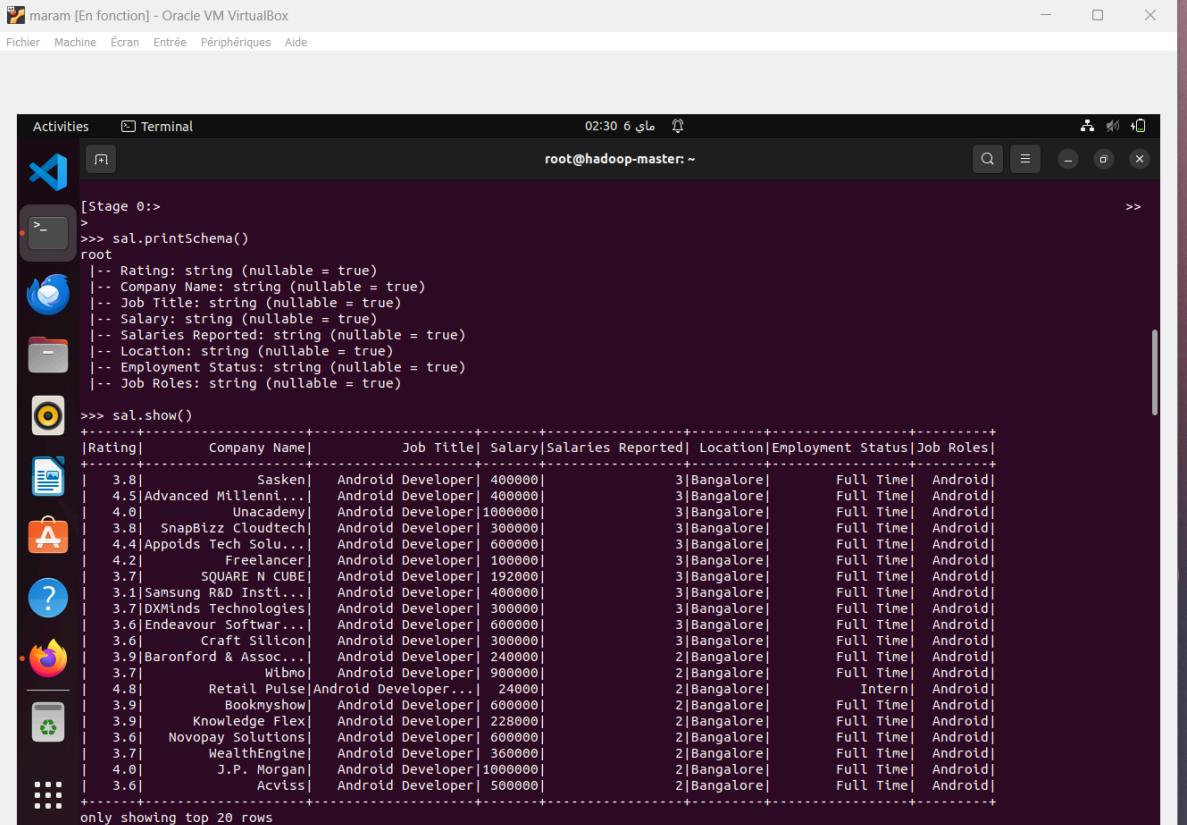


```
[Stage 0:>
> >>> sal.printSchema()
root
|-- Rating: string (nullable = true)
|-- Company Name: string (nullable = true)
|-- Job Title: string (nullable = true)
|-- Salary: string (nullable = true)
|-- Salaries Reported: string (nullable = true)
|-- Location: string (nullable = true)
|-- Employment Status: string (nullable = true)
|-- Job Roles: string (nullable = true)
>>> sal.show()
+---+-----+-----+-----+-----+-----+-----+
|Rating| Company Name| Job Title| Salary|Salaries Reported| Location|Employment Status|Job Roles|
+---+-----+-----+-----+-----+-----+-----+
| 3.8| Sasken      | Android Developer| 400000|      3|Bangalore| Full Time| Android|
| 4.5|Advanced Millenni...| Android Developer| 400000|      3|Bangalore| Full Time| Android|
| 4.0| Unacademy    | Android Developer|1000000|      3|Bangalore| Full Time| Android|
| 3.8| SnapBizz Cloudtech| Android Developer| 300000|      3|Bangalore| Full Time| Android|
| 4.4|Appolds Tech Solu...| Android Developer| 600000|      3|Bangalore| Full Time| Android|
| 4.2| Freelancer   | Android Developer| 100000|      3|Bangalore| Full Time| Android|
| 3.7|      SQUARE N CUBE| Android Developer| 192000|      3|Bangalore| Full Time| Android|
| 3.1|Samsung R&D Insti...| Android Developer| 400000|      3|Bangalore| Full Time| Android|
| 3.7|DXMinds Technologies| Android Developer| 300000|      3|Bangalore| Full Time| Android|
| 3.6|Endeavour Software...| Android Developer| 600000|      3|Bangalore| Full Time| Android|
| 3.6| Craft Silicon   | Android Developer| 300000|      3|Bangalore| Full Time| Android|
| 3.9|Baronford & Assoc...| Android Developer| 240000|      2|Bangalore| Full Time| Android|
| 3.7| Wibmo         | Android Developer| 900000|      2|Bangalore| Full Time| Android|
| 4.8| Retail Pulse   |Android Developer...| 24000|      2|Bangalore| Intern| Android|
| 3.9| Bookmyshow    | Android Developer| 600000|      2|Bangalore| Full Time| Android|
| 3.9| Knowledge Flex | Android Developer| 228000|      2|Bangalore| Full Time| Android|
| 3.6| Novopay Solutions| Android Developer| 600000|      2|Bangalore| Full Time| Android|
| 3.7| WealthEngine   | Android Developer| 360000|      2|Bangalore| Full Time| Android|
| 4.0| J.P. Morgan    | Android Developer|1000000|      2|Bangalore| Full Time| Android|
| 3.6| Acviss        | Android Developer| 500000|      2|Bangalore| Full Time| Android|
+---+-----+-----+-----+-----+-----+-----+
only showing top 20 rows
```

3. Afficher les premières lignes du DataFrame :

```
sal.show()
```

description: Cela affiche les premières lignes du DataFrame `sal`.



The screenshot shows a terminal window titled "maram [En fonction] - Oracle VM VirtualBox". The window contains Scala code and the resulting output of a DataFrame.

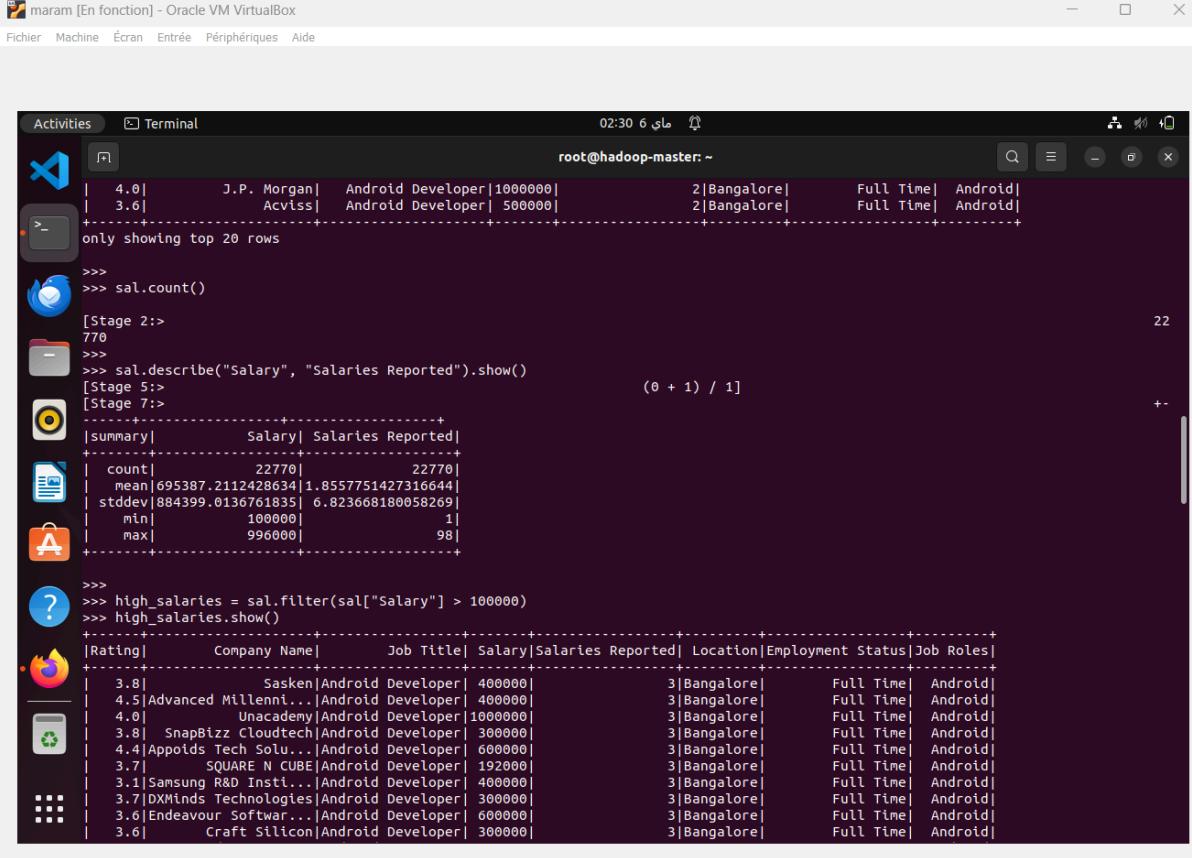
```
[Stage 0:>
>>> sal.printSchema()
root
|-- Rating: string (nullable = true)
|-- Company Name: string (nullable = true)
|-- Job Title: string (nullable = true)
|-- Salary: string (nullable = true)
|-- Salaries Reported: string (nullable = true)
|-- Location: string (nullable = true)
|-- Employment Status: string (nullable = true)
|-- Job Roles: string (nullable = true)

>>> sal.show()
+-----+-----+-----+-----+-----+-----+
|Rating|Company Name|Job Title|Salary|Salaries Reported|Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+-----+
| 3.8|Sasken|Android Developer|400000| 3|Bangalore| Full Time| Android|
| 4.5|Advanced Millenni...|Android Developer|400000| 3|Bangalore| Full Time| Android|
| 4.0|Unacademy|Android Developer|1000000| 3|Bangalore| Full Time| Android|
| 3.8|SnapBizz Cloudtech|Android Developer|300000| 3|Bangalore| Full Time| Android|
| 4.4|Appoids Tech Solu...|Android Developer|600000| 3|Bangalore| Full Time| Android|
| 4.2|Freelancer|Android Developer|100000| 3|Bangalore| Full Time| Android|
| 3.7|SQUARE N CUBE|Android Developer|192000| 3|Bangalore| Full Time| Android|
| 3.1|Samsung R&D Insti...|Android Developer|400000| 3|Bangalore| Full Time| Android|
| 3.7|DXMinds Technologies|Android Developer|300000| 3|Bangalore| Full Time| Android|
| 3.6|Endeavour Software...|Android Developer|600000| 3|Bangalore| Full Time| Android|
| 3.6|Craft Silicon|Android Developer|300000| 3|Bangalore| Full Time| Android|
| 3.9|Baronford & Assoc...|Android Developer|240000| 2|Bangalore| Full Time| Android|
| 3.7|Wibmo|Android Developer|900000| 2|Bangalore| Full Time| Android|
| 4.8|Retail Pulse|Android Developer...|24000| 2|Bangalore| Intern| Android|
| 3.9|Bookmyshow|Android Developer|600000| 2|Bangalore| Full Time| Android|
| 3.9|Knowledge Flex|Android Developer|228000| 2|Bangalore| Full Time| Android|
| 3.6|Novopay Solutions|Android Developer|600000| 2|Bangalore| Full Time| Android|
| 3.7|WealthEngine|Android Developer|360000| 2|Bangalore| Full Time| Android|
| 4.0|J.P. Morgan|Android Developer|1000000| 2|Bangalore| Full Time| Android|
| 3.6|Acviss|Android Developer|500000| 2|Bangalore| Full Time| Android|
+-----+
only showing top 20 rows
```

4. Compter le nombre total de lignes dans le DataFrame :

```
sal.count()
```

description: Cette commande compte le nombre total de lignes dans le DataFrame sal.



The screenshot shows a terminal window titled "maram [En fonction] - Oracle VM VirtualBox". The command `sal.count()` is run, resulting in the output "22". Below this, the command `sal.describe("Salary", "Salaries Reported").show()` is run, displaying descriptive statistics for the "Salary" and "Salaries Reported" columns. At the bottom, the command `high_salaries = sal.filter(sal["Salary"] > 100000).show()` is run, showing a filtered DataFrame with 3 rows of data where the salary is greater than 100,000.

```
| 4.0|      J.P. Morgan| Android Developer|1000000|      2|Bangalore| Full Time| Android|
| 3.6|          Acviss| Android Developer| 500000|      2|Bangalore| Full Time| Android|
+-----+
only showing top 20 rows

>>> >>> sal.count()
[Stage 2:>
770
>>>
>>> sal.describe("Salary", "Salaries Reported").show()
[Stage 5:>
[Stage 7:>
+-----+-----+
|summary|      Salary| Salaries Reported|
+-----+-----+
| count|    22770|      22770|
| mean| 695387.2112428634| 1.8557751427316644|
| stddev| 1884399.0136761835| 6.823668180058269|
| min|    100000|      1|
| max|    996000|     98|
+-----+-----+
>>>
>>> high_salaries = sal.filter(sal["Salary"] > 100000)
>>> high_salaries.show()
+-----+-----+-----+-----+-----+-----+-----+
|Rating| Company Name| Job Title| Salary|Salaries Reported| Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+-----+-----+
| 3.8| Sasken|Android Developer| 400000|      3|Bangalore| Full Time| Android|
| 4.5|Advanced Millenni...|Android Developer| 400000|      3|Bangalore| Full Time| Android|
| 4.0|      Unacademy|Android Developer|1000000|      3|Bangalore| Full Time| Android|
| 3.8| SnapBizz Cloudtch|Android Developer| 300000|      3|Bangalore| Full Time| Android|
| 4.4|Appoids Tech Solu...|Android Developer| 600000|      3|Bangalore| Full Time| Android|
| 3.7|      SQUARE N CUBE|Android Developer| 192000|      3|Bangalore| Full Time| Android|
| 3.1|Samsung R&D Insti...|Android Developer| 400000|      3|Bangalore| Full Time| Android|
| 3.7|DXMinds Technologies|Android Developer| 300000|      3|Bangalore| Full Time| Android|
| 3.6|Endeavour Software...|Android Developer| 600000|      3|Bangalore| Full Time| Android|
| 3.6|      Craft Silicon|Android Developer| 300000|      3|Bangalore| Full Time| Android|
```

5. Afficher des statistiques descriptives pour les colonnes "Salary" et "Salaries Reported" :

```
sal.describe("Salary", "Salaries Reported").show()
```

description: Cela affiche des statistiques descriptives telles que la moyenne, l'écart type, le minimum, le maximum, etc., pour les colonnes spécifiées.

```
| 4.0| J.P. Morgan| Android Developer|1000000| 2|Bangalore| Full Time| Android|
| 3.6| Acviss| Android Developer| 500000| 2|Bangalore| Full Time| Android|
+---+-----+-----+-----+-----+
only showing top 20 rows

>>> sal.count()
[Stage 2:: 770]
>>> sal.describe("Salary", "Salaries Reported").show()
[Stage 5::]
[Stage 7::]
+-----+-----+
|summary| Salary| Salaries Reported|
+-----+-----+
| count| 22770| 22770|
| mean| 695387.2112428634| 1.8557751427316644|
| stddev| 884399.0136761835| 6.823668180058269|
| min| 100000| 1|
| max| 996000| 98|
+-----+-----+
>>> high_salaries = sal.filter(sal["Salary"] > 100000)
>>> high_salaries.show()
+-----+-----+-----+-----+-----+-----+
|Rating| Company Name| Job Title| Salary| Salaries Reported| Location| Employment Status| Job Roles|
+-----+-----+-----+-----+-----+-----+
| 3.8| Sasken|Android Developer| 400000| 3|Bangalore| Full Time| Android|
| 4.5|Advanced Millenni...|Android Developer| 400000| 3|Bangalore| Full Time| Android|
| 4.0| Unacademy|Android Developer| 1000000| 3|Bangalore| Full Time| Android|
| 3.8| SnapBlitz CloudTech|Android Developer| 300000| 3|Bangalore| Full Time| Android|
| 4.4|Appoids Tech Solu...|Android Developer| 600000| 3|Bangalore| Full Time| Android|
| 3.7| SQUARE N CUBE|Android Developer| 192000| 3|Bangalore| Full Time| Android|
| 3.1|Samsung R&D Insti...|Android Developer| 400000| 3|Bangalore| Full Time| Android|
| 3.7|DXMhinds Technologies|Android Developer| 300000| 3|Bangalore| Full Time| Android|
| 3.6|Endeavour Software...|Android Developer| 600000| 3|Bangalore| Full Time| Android|
| 3.6| Craft Silicon|Android Developer| 300000| 3|Bangalore| Full Time| Android|
+-----+-----+-----+-----+-----+-----+
```

6. Filtrer les lignes avec un salaire supérieur à 100 000 :

```
high_salaries = sal.filter(sal["Salary"] > 100000)
high_salaries.show()
```

description: Cela filtre les lignes du DataFrame où la valeur de la colonne "Salary" est supérieure à 100 000 et affiche le résultat.

```

maram [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
Activities Terminal 02:31 6 آذار root@hadoop-master: ~
>_
>>>
>>> high_salaries = sal.filter(sal["Salary"] > 100000)
>>> high_salaries.show()
+-----+-----+-----+-----+-----+-----+
|Rating|Company Name|Job Title|Salary|Salaries Reported|Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+-----+
| 3.8|Sasken|Android Developer|400000| 3|Bangalore| Full Time| Android|
| 4.5|Advanced Millenni...|Android Developer|400000| 3|Bangalore| Full Time| Android|
| 4.0|Unacademy|Android Developer|1000000| 3|Bangalore| Full Time| Android|
| 3.8|SnapBizz Cloudech|Android Developer|300000| 3|Bangalore| Full Time| Android|
| 4.4|Appolds Tech Solu...|Android Developer|600000| 3|Bangalore| Full Time| Android|
| 3.7|SQUARE N CUBE|Android Developer|192000| 3|Bangalore| Full Time| Android|
| 3.1|Samsung R&D Insti...|Android Developer|400000| 3|Bangalore| Full Time| Android|
| 3.7|DXMinds Technologies|Android Developer|300000| 3|Bangalore| Full Time| Android|
| 3.6|Endeavour Software...|Android Developer|600000| 3|Bangalore| Full Time| Android|
| 3.6|Craft Silicon|Android Developer|300000| 3|Bangalore| Full Time| Android|
| 3.9|Baronford & Assoc...|Android Developer|240000| 2|Bangalore| Full Time| Android|
| 3.7|Wibmo|Android Developer|900000| 2|Bangalore| Full Time| Android|
| 3.9|Bookmyshow|Android Developer|600000| 2|Bangalore| Full Time| Android|
| 3.9|Knowledge Flex|Android Developer|228000| 2|Bangalore| Full Time| Android|
| 3.6|Novopay Solutions|Android Developer|600000| 2|Bangalore| Full Time| Android|
| 3.7|WealthEngine|Android Developer|360000| 2|Bangalore| Full Time| Android|
| 4.0|J.P. Morgan|Android Developer|1000000| 2|Bangalore| Full Time| Android|
| 3.6|Acviss|Android Developer|500000| 2|Bangalore| Full Time| Android|
| 4.1|Fresher|Android Developer|408000| 2|Bangalore| Full Time| Android|
| 4.2|MedOnGo|Android Developer|300000| 2|Bangalore| Full Time| Android|
+-----+
only showing top 20 rows
>>> sorted_by_salary = sal.orderBy(sal["Salary"].desc())
>>> sorted_by_salary.show()
[Stage 9:]
+-----+-----+-----+-----+-----+
|Rating|Company Name|Job Title|Salary|Salaries Reported|Location|Employment Status|Job Roles|
+-----+-----+-----+-----+-----+
| 3.9|ennVee TechnoGroup|Oracle Applicatio...|996000| 1|Bangalore| Full Time| Database|
+-----+

```

7. Calculer le salaire moyen par entreprise :

```
avg_salary_by_company = sal.groupBy("Company Name").avg("Salary")
avg_salary_by_company.show()
```

description: Cela regroupe les données par entreprise et calcule la moyenne du salaire pour chaque entreprise, puis affiche le résultat.

8. Trier le DataFrame par salaire décroissant :

```
sorted_by_salary = sal.orderBy(sal["Salary"].desc())
sorted_by_salary.show()
```

description: Cela trie le DataFrame en fonction de la colonne "Salary" de manière décroissante et affiche le résultat.

```

Activities Terminal 02:31 6 مارس 2024 root@hadoop-master: ~
root@hadoop-master: ~
+-----+
| 4.1| Fresher|Android Developer| 408000| 2|Bangalore| Full Time| Android|
| 4.2| MedOnGo|Android Developer| 300000| 2|Bangalore| Full Time| Android|
+-----+
only showing top 20 rows

>>> sorted_by_salary = sal.orderBy(sal["Salary"].desc())
>>> sorted_by_salary.show()
[Stage 9:]
+-----+
|Rating| Company Name| Job Title|Salary|Salaries Reported| Location|Employment Status|Job Roles|
+-----+
| 3.9| ennVee TechnoGroup|Oracle Applicatio...|996000| 1|Bangalore| Full Time| Database|
| 4.2| MakeMyTrip|Software Developm...|996000| 4|Bangalore| Full Time| SDE|
| 4.3| Esper| Android Engineer|996000| 1|Bangalore| Full Time| Android|
| 4.3| Iris Software|Senior Android De...|996000| 1|New Delhi| Contractor| Android|
| 4.2| 9Logic Technologi...| Android Developer|996000| 1| Chennai| Full Time| Android|
| 3.8| Ecom Express| Front End Developer|996000| 1|New Delhi| Full Time| Frontend|
| 3.9| Sociolla|Senior iOS App De...|996000| 1|New Delhi| Full Time| IOS|
| 5.0| Ramcides|Software Engineer...|996000| 1|Hyderabad| Full Time| Java|
| 5.0| AB Solutions Lab|Mobile App Developer|996000| 1|Bangalore| Full Time| Mobile|
| 4.0| Goldman Sachs|Software Developm...|996000| 4|Bangalore| Intern| SDE|
| 4.4| Microsoft|Software Developm...|996000| 1| Chennai| Intern| SDE|
| 3.8| Amazon|Software Developm...|996000| 1|Hyderabad| Full Time| SDE|
| 3.7| US Department of ...|Software Developm...|996000| 1|Hyderabad| Full Time| SDE|
| 2.7| Data Dimensions|Software Developm...|996000| 1| Mumbai| Full Time| SDE|
| 4.0| Sira Consulting|Software Developm...|996000| 1|New Delhi| Intern| SDE|
| 4.0| CrossChannel|Software Developm...|996000| 1| Pune| Full Time| SDE|
| 3.5| Diamanti|Software Developm...|996000| 1| Pune| Full Time| SDE|
| 4.4| ThinkBridge|Software Developm...|996000| 1| Pune| Full Time| SDE|
| 3.7| CDK Global| Test Engineer|996000| 1|Hyderabad| Full Time| Testing|
| 3.9| Tata Consultancy ...| Tester|996000| 8|Hyderabad| Full Time| Testing|
+-----+
only showing top 20 rows

>>> df.select('Company Name', 'Salary').show()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'df' is not defined
>>> avg_salary_by_company = sal.groupBy("Company Name").avg("Salary")
Traceback (most recent call last):

```

III-les erreurs et problèmes rencontrés:

1/Lorsque vous avez téléchargé l'image(image lilia dans ma projet) et vous voulez de nouveau exécuter le root hadoop-master vous devez start tous les workers et le master en exécutant:

`sudo docker start hadoop-master`

`sudo docker start hadoop-worker 1 sudo docker start hadoop-worker n`

2/lorsque j'ai exécuté la commande mapreduce j'ai besoin de corriger le fichier mapred car il était incomplet il faut exécuter :

```

maram [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
Activities Terminal 11:23 6 مای 2023
root@hadoop-master: /usr/local/hadoop/etc/hadoop
[+]
netty-transport-sctp-4.1.89.Final.jar
netty-transport-udt-4.1.89.Final.jar
nimbus-jose-jwt-9.8.1.jar
okhttp-4.9.3.jar
okio-2.8.0.jar
paranamer-2.9.jar
protobuf-java-2.5.0.jar
re2j-1.1.jar
reloadadj-1.2.22.jar
snappy-java-1.1.8.2.jar
stax2-apt-4.2.1.jar
token-provider-1.0.1.jar
woodstox-core-5.4.0.jar
zookeeper-3.6.3.jar
zookeeper-jute-3.6.3.jar
root@hadoop-master: /usr/local/hadoop/share/hadoop/hdfs/lib# cd /usr/local/hadoop/etc/hadoop
root@hadoop-master: /usr/local/hadoop/etc/hadoop# ls
capacity-scheduler.xml      kms-log4j.properties
configuration.xml           kms-site.xml
container-executor.cfg      log4j.properties
core-site.xml                mapred-env.cmd
hadoop-env.cmd              mapred-env.sh
hadoop-env.sh               mapred-queues.xml.template
hadoop-metrics2.properties   mapred-site.xml
hadoop-policy.xml            shellprofile.d
hadoop-user-functions.sh.example ssl-client.xml.example
hdfs-rbf-site.xml            ssl-server.xml.example
hdfs-site.xml                user_ec_policies.xml.template
httppfs-env.sh               workers
httppfs-log4j.properties     yarn-env.cmd
httppfs-site.xml             yarn-env.sh
kms-acls.xml                 yarn-site.xml
kms-env.sh                   yarnservice-log4j.properties
root@hadoop-master: /usr/local/hadoop/etc/hadoop# nano mapred-site.xml
bash: nano: command not found
root@hadoop-master: /usr/local/hadoop/etc/hadoop# vi mapred-site.xml
[1]+ Stopped                  vi mapred-site.xml
root@hadoop-master: /usr/local/hadoop/etc/hadoop#

```



```

Activities Terminal 11:23 6 مای 2023
root@hadoop-master: /usr/local/hadoop/etc/hadoop
[+]
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
    Licensed 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. See accompanying LICENSE file.
-->
<!-- Put site-specific property overrides in this file. -->
<configuration>
    <property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
    </property>
    <property>
        <name>yarn.app.mapreduce.am.env</name>
        <value>HADOOP_MAPRED_HOME=/usr/local/hadoop</value>
    </property>
    <property>
        <name>mapreduce.map.env</name>
        <value>HADOOP_MAPRED_HOME=/usr/local/hadoop</value>
    </property>
    <property>
        <name>mapreduce.reduce.env</name>
        <value>HADOOP_MAPRED_HOME=/usr/local/hadoop</value>
    </property>
</configuration>

```

3. Erreur de connexion au réseau lors de la création du réseau Docker :

- Problème : Vous pourriez rencontrer des erreurs lors de la création du réseau Docker en raison de problèmes de connectivité réseau.

- Solution : Assurez-vous d'avoir une connexion réseau active sur votre machine. Vérifiez également si Docker est autorisé à accéder au réseau en vérifiant les paramètres de votre pare-feu ou de votre proxy.
4. Erreur lors de la copie de fichiers vers le conteneur Hadoop :
- Problème : Vous pourriez rencontrer des erreurs lors de la copie de fichiers vers le conteneur Hadoop en raison de permissions insuffisantes ou de chemins de fichiers incorrects.
 - Solution : Assurez-vous d'avoir les permissions nécessaires pour copier les fichiers vers le conteneur Docker. Utilisez la commande `sudo` si nécessaire. Vérifiez également que le chemin de fichier spécifié est correct.

```

DESKTOP      Downloads      Mapreduce      Public      spark-3.2.0-bin-hadoop3.2  Videos
maran@maram-VirtualBox:~$ cd ~
maran@maram-VirtualBox:~$ cd Downloads
maran@maram-VirtualBox:~/Downloads$ ls
avecLiliasfaximapreduce.pdf  OpenJDK8U-jdk_x64_linux_hotspot_8u402b06.tar.gz  Python-3.12.2.tar.xz  Salary.csv.zip
Mapred      purchases.txt  Salary.csv
maran@maram-VirtualBox:~/Downloads$ chmod 755 purchases.txt
maran@maram-VirtualBox:~/Downloads$ 

```

5/lors de l'exécution du mapreduce vous pouvez rencontrer l'erreur de safemode vous devez le désactiver en exécutant :

`hdfs dfsadmin -safemode leave`