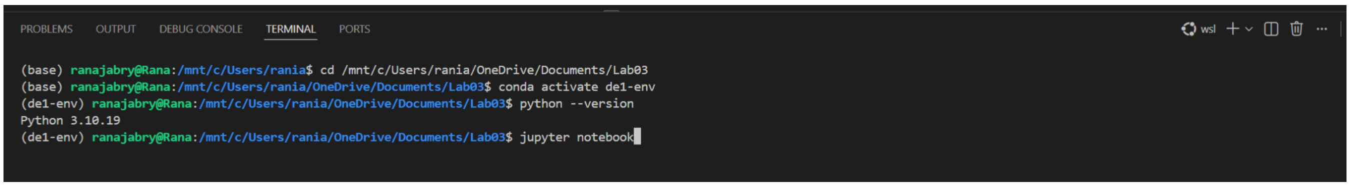
Rapport des labs practice

Lab01

Pour chaque lab, la première étape consiste à lancer l'environnement de travail.
Sachant que l'ensemble des dépendances (Python, Spark, environnement Conda, Jupyter Notebook) a déjà été installé et configuré lors du **Lab 0**

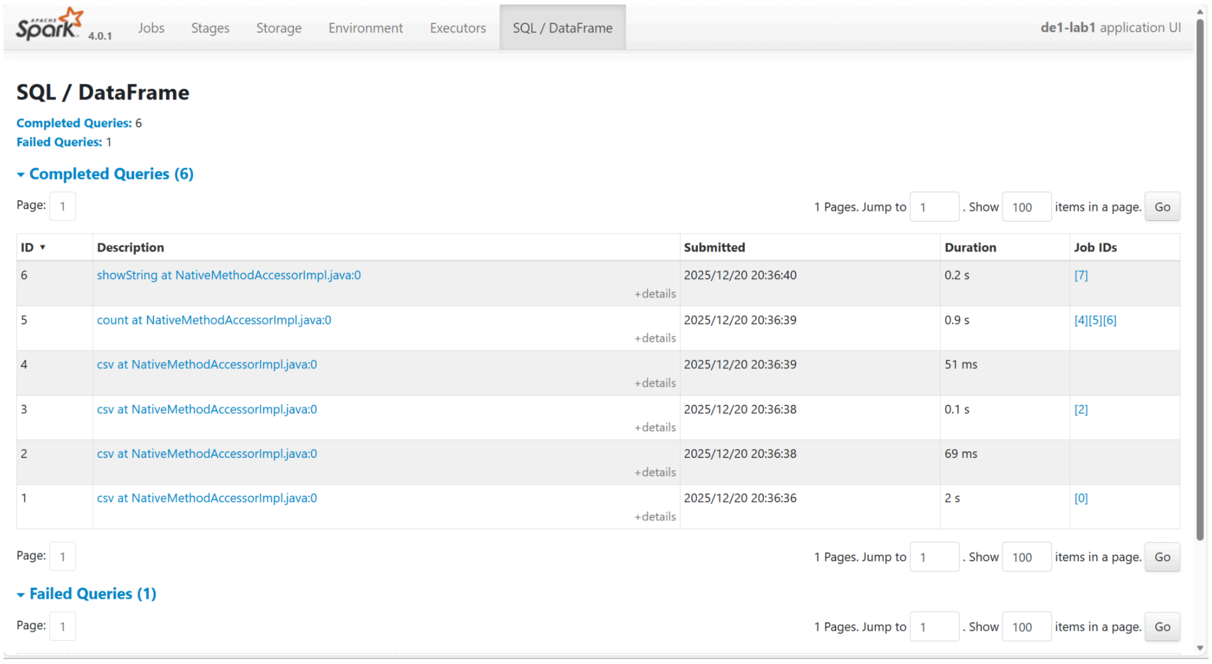


```
(base) ranajabry@Rana:/mnt/c/Users/rania$ cd /mnt/c/Users/rania/OneDrive/Documents/Lab03
(base) ranajabry@Rana:/mnt/c/Users/rania/OneDrive/Documents/Lab03$ conda activate de1-env
(de1-env) ranajabry@Rana:/mnt/c/Users/rania/OneDrive/Documents/Lab03$ python --version
Python 3.10.19
(de1-env) ranajabry@Rana:/mnt/c/Users/rania/OneDrive/Documents/Lab03$ jupyter notebook
```

La capture d'écran ci-dessus illustre cette étape initiale, qui est commune à tous les labs et permet d'exécuter correctement l'ensemble des notebooks.

Dans la première partie du lab "**Import and Spark session**", aucune action particulière n'est requise. À ce stade, l'environnement est correctement configuré : la version de Python utilisée est **3.10.19** et la version de Spark est **4.0.1**
Dans l'interface Spark UI, section SQL / DataFrame, on observe la présence de plusieurs jobs identifiés par des IDs. Toutefois, à ce moment-là, aucune action significative n'a encore été déclenchée.

Dans la partie **Top-N with RDD API**, le code s'exécute sans erreur et produit les résultats attendus. Les jobs associés sont visibles dans Spark UI



SQL / DataFrame

Completed Queries: 6
Failed Queries: 1

Completed Queries (6)

Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go

ID	Description	Submitted	Duration	Job IDs
6	showString at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:40	0.2 s	[7]
5	count at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:39	0.9 s	[4][5][6]
4	csv at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:39	51 ms	
3	csv at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:38	0.1 s	[2]
2	csv at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:38	69 ms	
1	csv at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:36	2 s	[0]

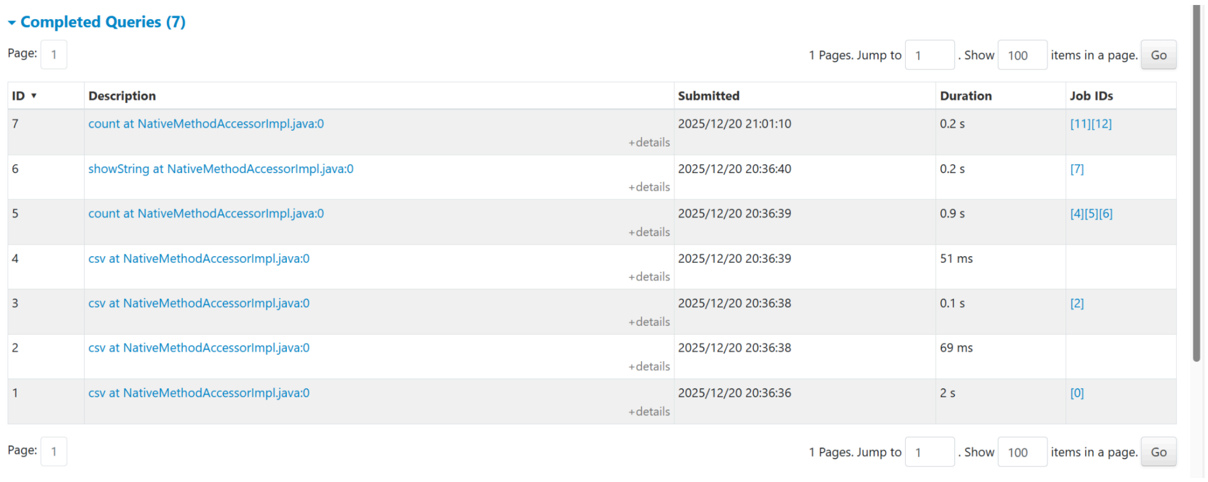
Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go

Failed Queries (1)

Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go

Afin de faciliter l'identification d'une exécution Spark SQL claire et isolée dans l'onglet SQL / DataFrame, nous avons volontairement ajouté une action explicite : df.select("text").count()

Cette instruction permet de forcer une exécution Spark SQL simple et facilement repérable dans Spark UI. La requête `count` apparaît clairement dans la liste des requêtes SQL (ID 7, Job associés [11][12])

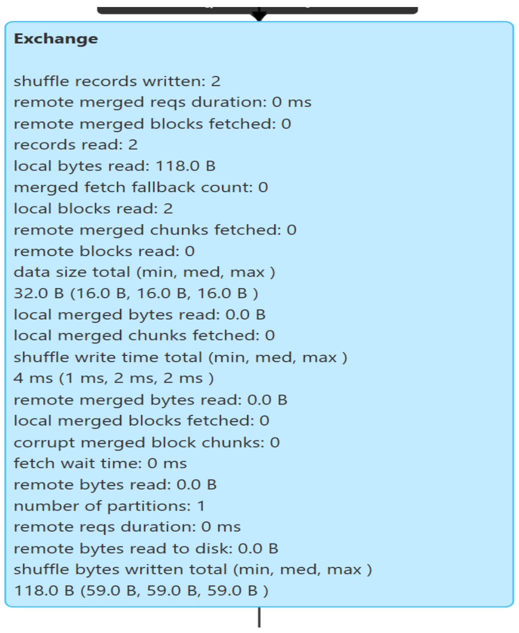


Completed Queries (7)

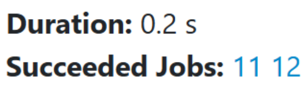
Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go

ID	Description	Submitted	Duration	Job IDs
7	count at NativeMethodAccessorImpl.java:0 +details	2025/12/20 21:01:10	0.2 s	[11][12]
6	showString at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:40	0.2 s	[7]
5	count at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:39	0.9 s	[4][5][6]
4	csv at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:39	51 ms	
3	csv at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:38	0.1 s	[2]
2	csv at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:38	69 ms	
1	csv at NativeMethodAccessorImpl.java:0 +details	2025/12/20 20:36:36	2 s	[0]

Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go



```
Exchange

shuffle records written: 2
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 2
local bytes read: 118.0 B
merged fetch fallback count: 0
local blocks read: 2
remote merged chunks fetched: 0
remote blocks read: 0
data size total (min, med, max )
32.0 B (16.0 B, 16.0 B, 16.0 B )
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time total (min, med, max )
4 ms (1 ms, 2 ms, 2 ms )
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 1
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written total (min, med, max )
118.0 B (59.0 B, 59.0 B, 59.0 B )
```

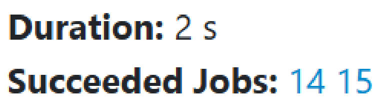


Duration: 0.2 s
Succeeded Jobs: 11 12

Les métriques sont présentes dans le nœud Exchange, notamment celles liées au shuffle et aux volumes de données traitées (Voir la capture d'écran ci-dessus). Ces informations ont ensuite été utilisées pour remplir la première ligne du fichier CSV des métriques correspondant à la partie ***Top-N with RDD API***

Enfin, après cette étape, nous passons à l’analyse du **DataFrame plan**, en appliquant exactement la même logique. O

On obtient:

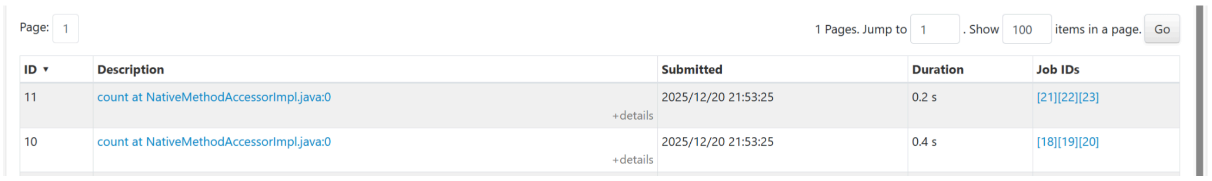


Duration: 2 s
Succeeded Jobs: 14 15

Dans la partie “**Projection experiment: select (*) vs minimal projection”**, l’objectif est de comparer l’impact de la projection des colonnes sur les performances d’exécution. Deux cas sont étudiés dans le même bloc de code.

Case A correspond à une projection complète, où toutes les colonnes sont sélectionnées (`select *`), avant d’appliquer une agrégation sur la catégorie.
Le Case B correspond à une projection minimale, où seules les colonnes strictement nécessaires à l’agrégation sont sélectionnées avant l’opération.

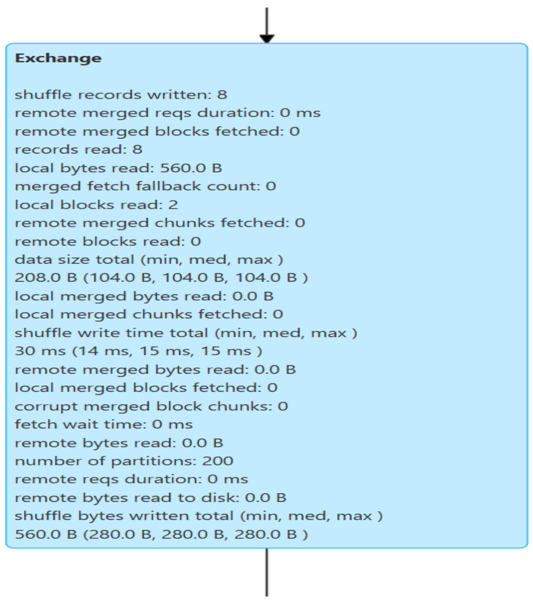
Lors de l’exécution de ce bloc, deux exécutions distinctes apparaissent dans la section **SQL / DataFrame** de Spark UI, chacune correspondant à l’un des deux cas.



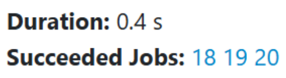
Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go

ID ▾	Description	Submitted	Duration	Job IDs
11	count at NativeMethodAccessorImpl.java:0 +details	2025/12/20 21:53:25	0.2 s	[21][22][23]
10	count at NativeMethodAccessorImpl.java:0 +details	2025/12/20 21:53:25	0.4 s	[18][19][20]

Projection_all_cols: La durée d’exécution observée est de 0,4 secondes, et les jobs associés sont les jobs [18], [19] et [20].
Les métriques ont été collectées à partir du nœud **Exchange**

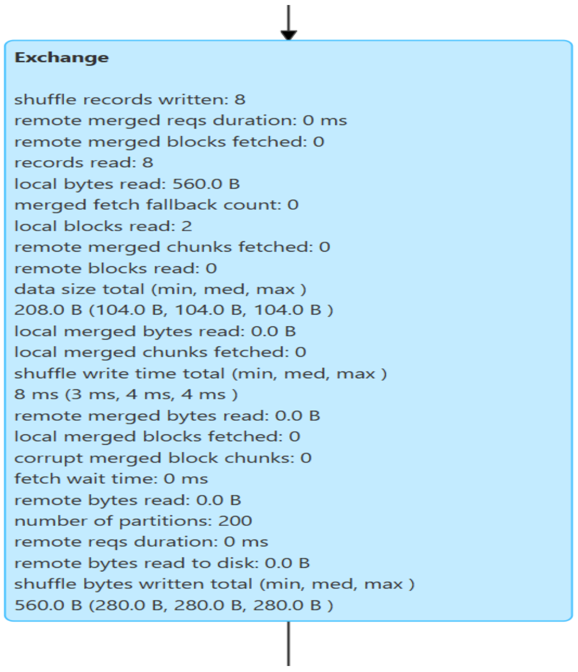


```
Exchange

shuffle records written: 8
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 8
local bytes read: 560.0 B
merged fetch fallback count: 0
local blocks read: 2
remote merged chunks fetched: 0
remote blocks read: 0
data size total (min, med, max )
208.0 B (104.0 B, 104.0 B, 104.0 B )
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time total (min, med, max )
30 ms (14 ms, 15 ms, 15 ms )
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written total (min, med, max )
560.0 B (280.0 B, 280.0 B, 280.0 B )
```

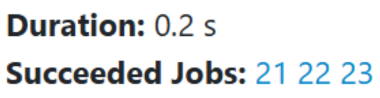


Duration: 0.4 s
Succeeded Jobs: 18 19 20

Projection_min: La durée d'exécution observée est de 0,2 secondes, et les jobs associés sont les jobs [21], [22] et [23].



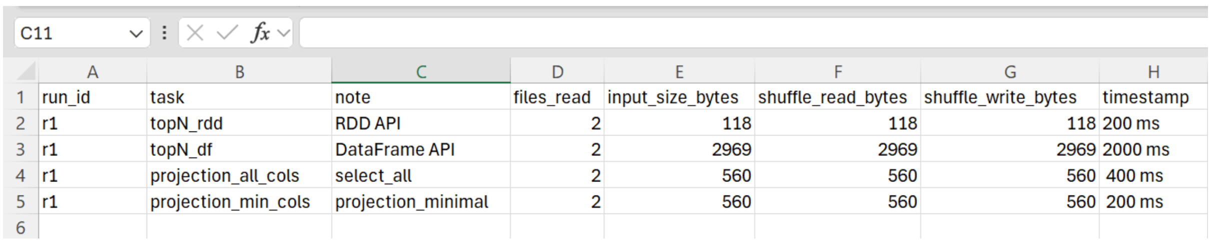
```
Exchange

shuffle records written: 8
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 8
local bytes read: 560.0 B
merged fetch fallback count: 0
local blocks read: 2
remote merged chunks fetched: 0
remote blocks read: 0
data size total (min, med, max )
208.0 B (104.0 B, 104.0 B, 104.0 B )
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time total (min, med, max )
8 ms (3 ms, 4 ms, 4 ms )
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written total (min, med, max )
560.0 B (280.0 B, 280.0 B, 280.0 B )
```



Duration: 0.2 s

Succeeded Jobs: 21 22 23

Le tableau présenté ci-dessous récapitule l'ensemble des métriques collectées au cours de ce lab, pour les différentes expérimentations réalisées : **RDD API**, **DataFrame API**, **projection de toutes les colonnes** et **projection minimale**. Les valeurs reportées dans ce tableau proviennent directement des métriques observées dans Spark UI, notamment à partir des informations de durée d'exécution, du nœud Exchange.

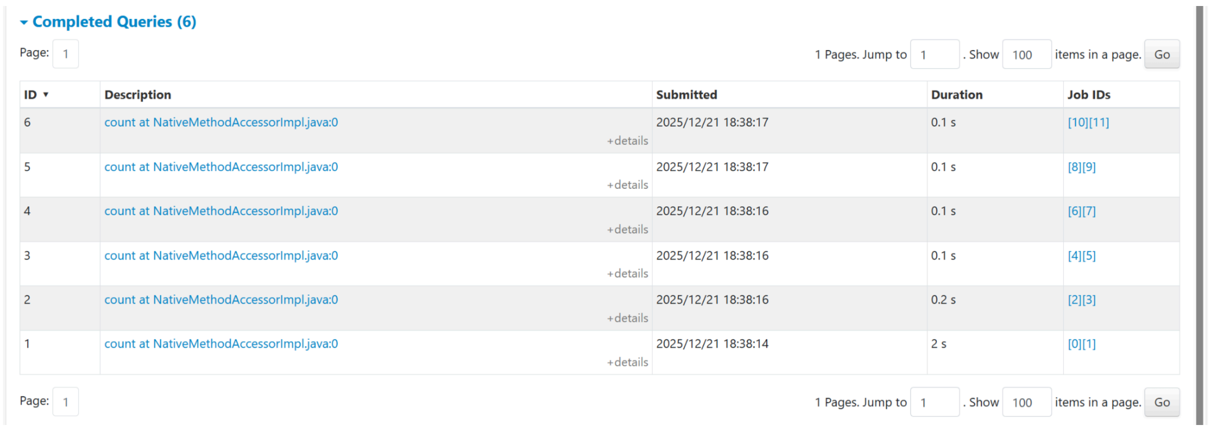


run_id	task	note	files_read	input_size_bytes	shuffle_read_bytes	shuffle_write_bytes	timestamp
r1	topN_rdd	RDD API	2	118	118	118	200 ms
r1	topN_df	DataFrame API	2	2969	2969	2969	2000 ms
r1	projection_all_cols	select_all	2	560	560	560	400 ms
r1	projection_min_cols	projection_minimal	2	560	560	560	200 ms

Fin du Lab01

Lab02

Dans cette première section “**Ingest operational tables from CSV exports**“ , nous avons chargé les tables opérationnelles à partir de fichiers CSV.
L’exécution des cellules a déclenché six jobs, correspondant aux différentes opérations de lecture et aux actions `count` utilisées pour valider les chargements.

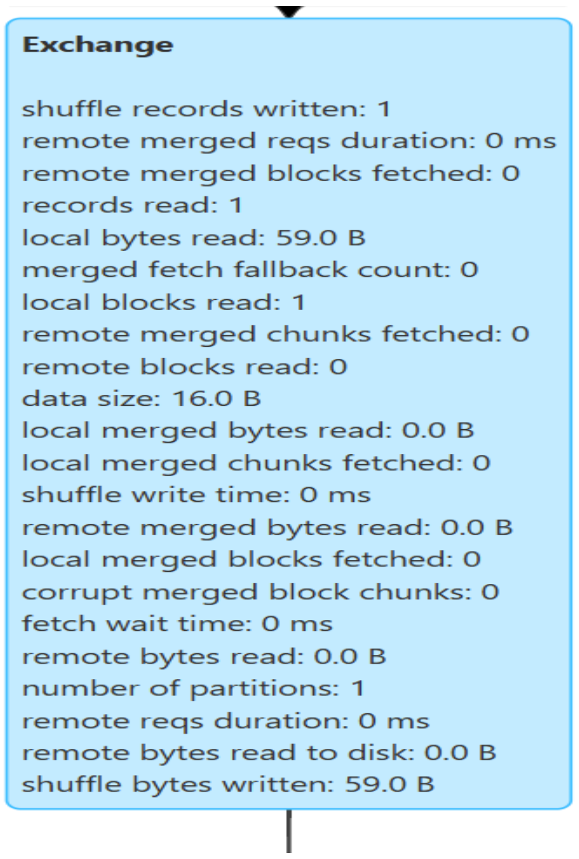


Completed Queries (6)

Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go

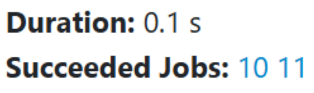
ID	Description	Submitted	Duration	Job IDs
6	count at NativeMethodAccessorImpl.java:0 +details	2025/12/21 18:38:17	0.1 s	[10][11]
5	count at NativeMethodAccessorImpl.java:0 +details	2025/12/21 18:38:17	0.1 s	[8][9]
4	count at NativeMethodAccessorImpl.java:0 +details	2025/12/21 18:38:16	0.1 s	[6][7]
3	count at NativeMethodAccessorImpl.java:0 +details	2025/12/21 18:38:16	0.1 s	[4][5]
2	count at NativeMethodAccessorImpl.java:0 +details	2025/12/21 18:38:16	0.2 s	[2][3]
1	count at NativeMethodAccessorImpl.java:0 +details	2025/12/21 18:38:14	2 s	[0][1]

Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go



Exchange

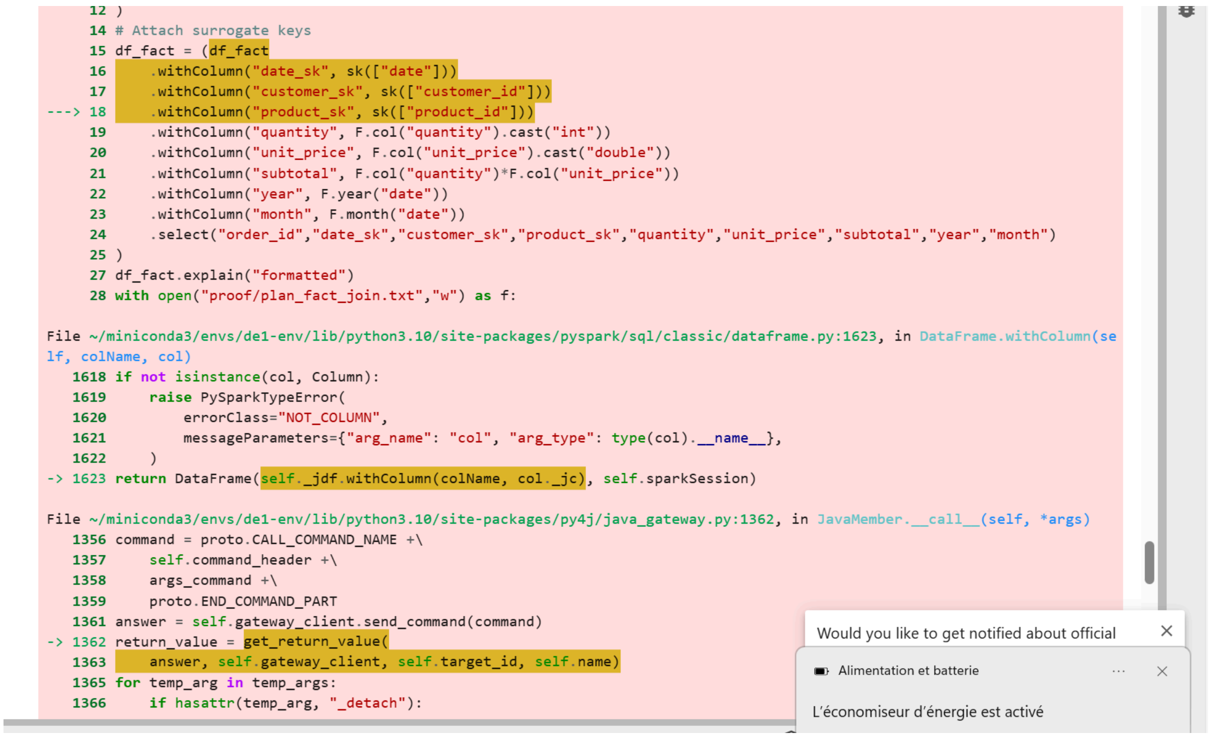
shuffle records written: 1
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 1
local bytes read: 59.0 B
merged fetch fallback count: 0
local blocks read: 1
remote merged chunks fetched: 0
remote blocks read: 0
data size: 16.0 B
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time: 0 ms
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 1
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written: 59.0 B



Duration: 0.1 s
Succeeded Jobs: 10 11

La durée observée est de 0,1 secondes, avec les jobs associés [10] et [11].
Les métriques ont été relevées à partir du nœud Exchange et ont permis de remplir la ligne ingest_plan du fichier CSV des métriques.
Après cette section, une erreur est apparue lors de l’exécution du notebook.

Cette erreur était liée à une ambiguïté de colonnes après les jointures, due à la présence de colonnes portant le même nom dans plusieurs tables.
La correction consiste à effectuer un select explicite après les jointures, afin de conserver une seule version de chaque colonne et d'éliminer toute ambiguïté.



```
      12 )
      14 # Attach surrogate keys
      15 df_fact = (df_fact
      16     .withColumn("date_sk", sk(["date"]))
      17     .withColumn("customer_sk", sk(["customer_id"]))
---> 18     .withColumn("product_sk", sk(["product_id"]))
      19     .withColumn("quantity", F.col("quantity").cast("int"))
      20     .withColumn("unit_price", F.col("unit_price").cast("double"))
      21     .withColumn("subtotal", F.col("quantity")*F.col("unit_price"))
      22     .withColumn("year", F.year("date"))
      23     .withColumn("month", F.month("date"))
      24     .select("order_id","date_sk","customer_sk","product_sk","quantity","unit_price","subtotal","year","month")
      25 )
      27 df_fact.explain("formatted")
      28 with open("proof/plan_fact_join.txt","w") as f:

File ~/miniconda3/envs/de1-env/lib/python3.10/site-packages/pyspark/sql/classic/dataframe.py:1623, in DataFrame.withColumn(se
lf, colName, col)
   1618 if not isinstance(col, Column):
   1619     raise PySparkTypeError(
   1620         errorClass="NOT_COLUMN",
   1621         messageParameters={"arg_name": "col", "arg_type": type(col).__name__},
   1622     )
-> 1623 return DataFrame(self._jdf.withColumn(colName, col._jc), self.sparkSession)

File ~/miniconda3/envs/de1-env/lib/python3.10/site-packages/py4j/java_gateway.py:1362, in JavaMember.__call__(self, *args)
   1356 command = proto.CALL_COMMAND_NAME +\
   1357     self.command_header +\
   1358     args_command +\
   1359     proto.END_COMMAND_PART
   1361 answer = self.gateway_client.send_command(command)
-> 1362 return_value = get_return_value(
   1363     answer, self.gateway_client, self.target_id, self.name)
   1365 for temp_arg in temp_args:
   1366     if hasattr(temp_arg, "_detach"):
```

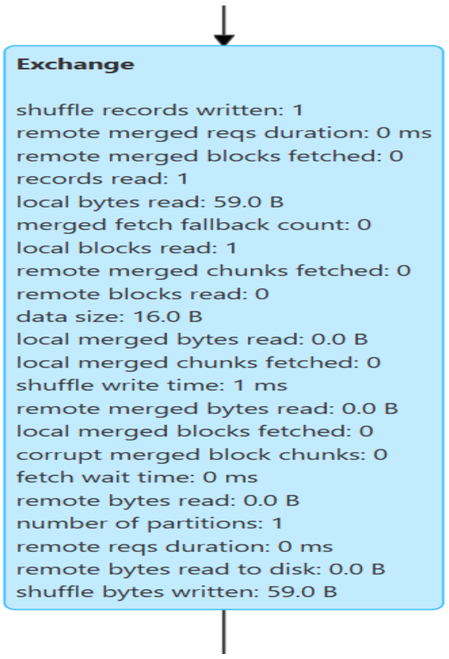
Would you like to get notified about official

Alimentation et batterie

L'économiseur d'énergie est activé

La version corrigée est visible dans le notebook final (La correction a été proposée par Chatgpt)

Dans la partie **Fact Join**, nous avons ajouté une action explicite (`count`) après la sauvegarde du plan logique afin de déclencher une exécution Spark mesurable.

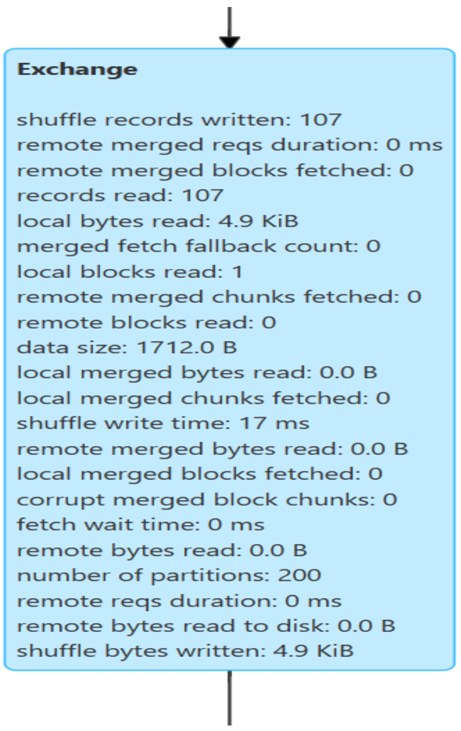


```
Exchange

shuffle records written: 1
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 1
local bytes read: 59.0 B
merged fetch fallback count: 0
local blocks read: 1
remote merged chunks fetched: 0
remote blocks read: 0
data size: 16.0 B
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time: 1 ms
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 1
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written: 59.0 B
```

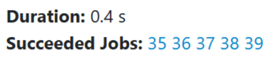
Duration: 0.5 s
Succeeded Jobs: 20 21 22 23 24

Pour cette section “Join then Project vs Project then Join”, les deux cas ont été exécutés dans des cellules séparées.

Case A — Join then Project

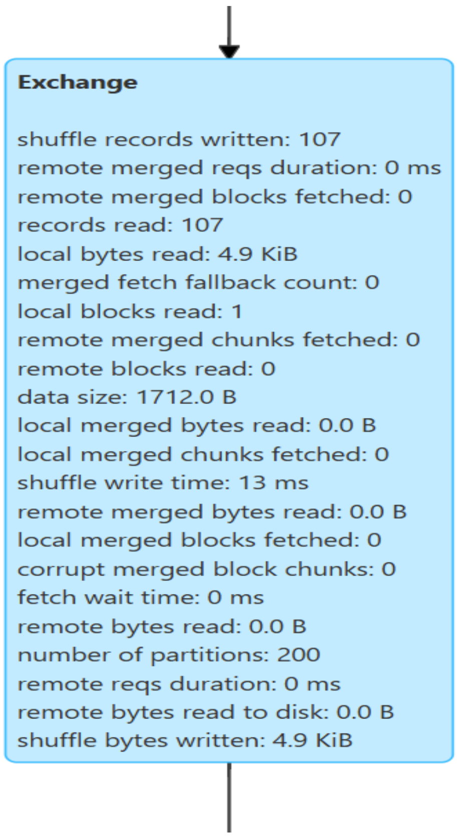


```
Exchange

shuffle records written: 107
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 107
local bytes read: 4.9 KiB
merged fetch fallback count: 0
local blocks read: 1
remote merged chunks fetched: 0
remote blocks read: 0
data size: 1712.0 B
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time: 17 ms
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written: 4.9 KiB
```

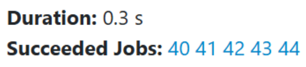


Duration: 0.4 s
Succeeded Jobs: 35 36 37 38 39

Case B — Project then Join

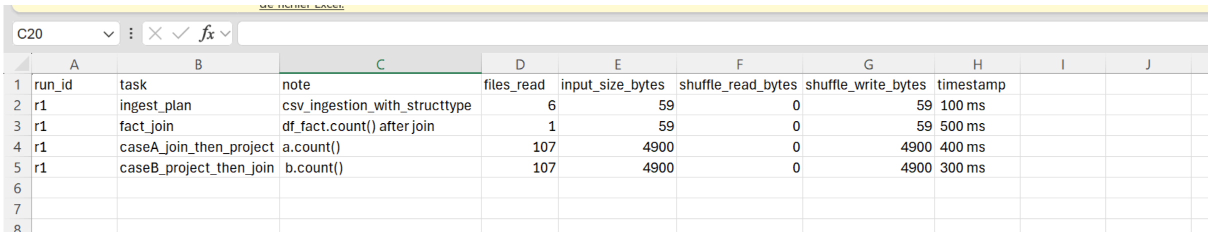


```
Exchange

shuffle records written: 107
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 107
local bytes read: 4.9 KiB
merged fetch fallback count: 0
local blocks read: 1
remote merged chunks fetched: 0
remote blocks read: 0
data size: 1712.0 B
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time: 13 ms
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written: 4.9 KiB
```



Duration: 0.3 s
Succeeded Jobs: 40 41 42 43 44

Le tableau présenté ci-dessous récapitule l’ensemble des métriques collectées au cours de ce lab

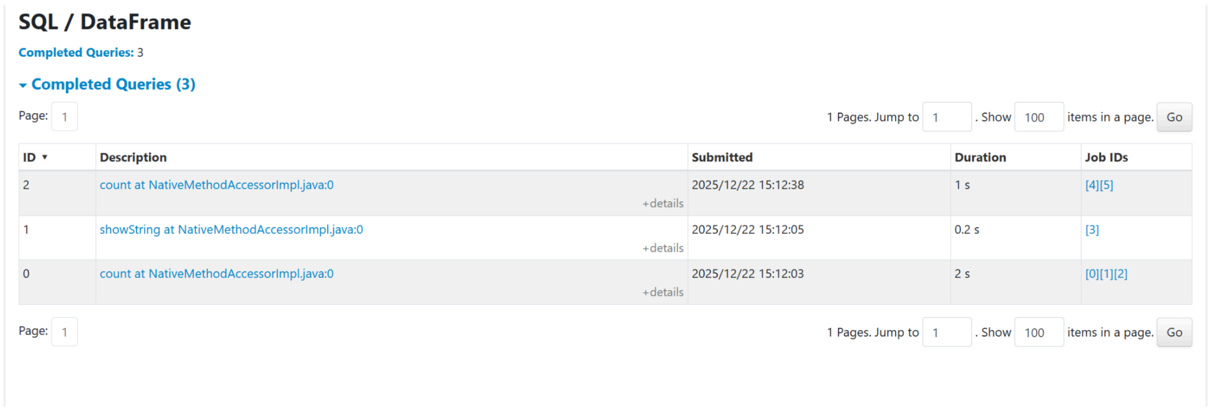


run_id	task	note	files_read	input_size_bytes	shuffle_read_bytes	shuffle_write_bytes	timestamp
r1	ingest_plan	csv_ingestion_with_structtype	6	59	0	59	100 ms
r1	fact_join	df_fact.count() after join	1	59	0	59	500 ms
r1	caseA_join_then_project	a.count()	107	4900	0	4900	400 ms
r1	caseB_project_then_join	b.count()	107	4900	0	4900	300 ms

Fin du Lab02

Lab03

Pour la requête Q1 en représentation row, une action explicite (`q1_row.count()`) a été ajoutée après la génération du plan d'exécution.
Cette action a déclenché une exécution Spark clairement identifiable dans Spark UI.



SQL / DataFrame

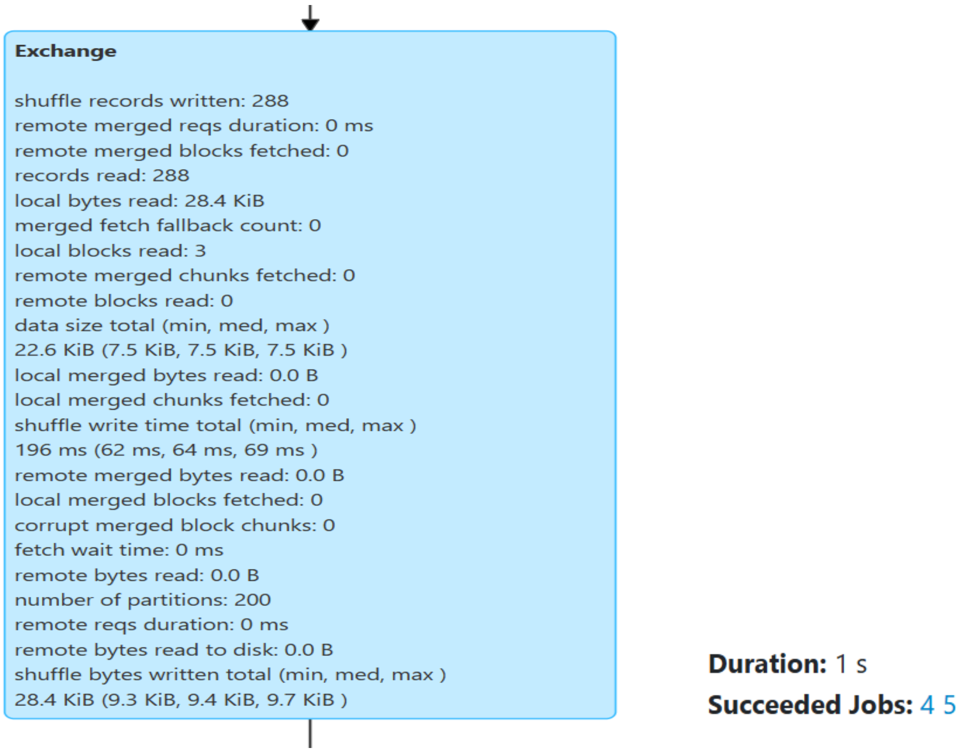
Completed Queries: 3

Completed Queries (3)

Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go

ID	Description	Submitted	Duration	Job IDs
2	count at NativeMethodAccessorImpl.java:0 +details	2025/12/22 15:12:38	1 s	[4][5]
1	showString at NativeMethodAccessorImpl.java:0 +details	2025/12/22 15:12:05	0.2 s	[3]
0	count at NativeMethodAccessorImpl.java:0 +details	2025/12/22 15:12:03	2 s	[0][1][2]

Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go

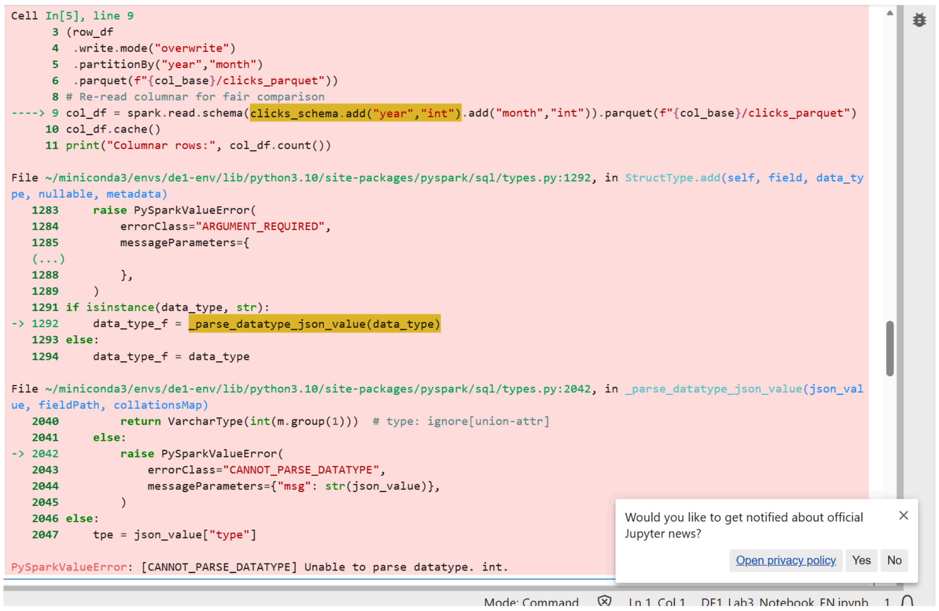


```
Exchange

shuffle records written: 288
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 288
local bytes read: 28.4 KiB
merged fetch fallback count: 0
local blocks read: 3
remote merged chunks fetched: 0
remote blocks read: 0
data size total (min, med, max )
22.6 KiB (7.5 KiB, 7.5 KiB, 7.5 KiB )
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time total (min, med, max )
196 ms (62 ms, 64 ms, 69 ms )
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written total (min, med, max )
28.4 KiB (9.3 KiB, 9.4 KiB, 9.7 KiB )
```

Duration: 1 s
Succeeded Jobs: 4 5

Ces informations ont servi à remplir la ligne Q1 – row dans le fichier CSV des métriques.

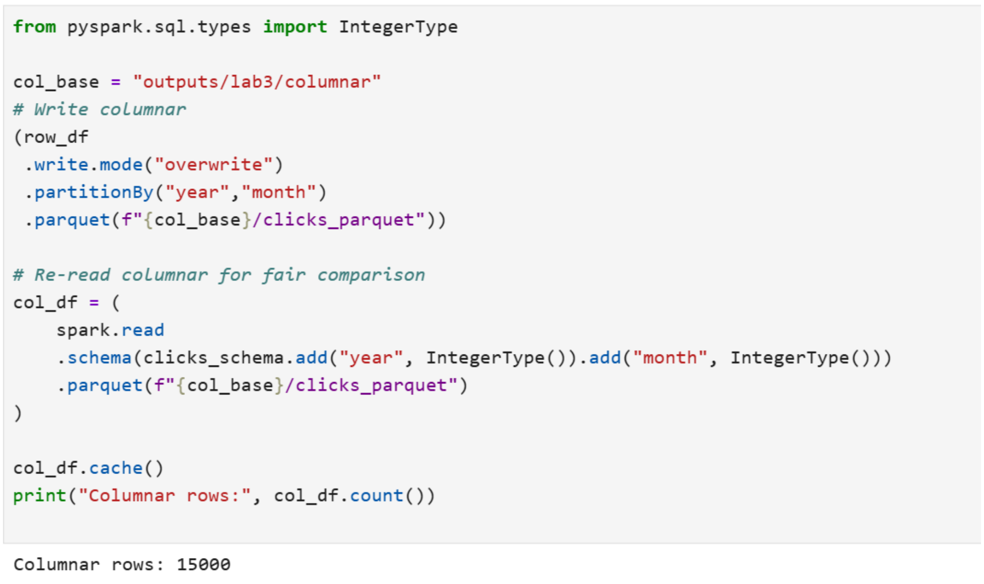
On a exécuté la cellule d’après, et on a eu une erreur



```
Cell In[5], line 9
      3 (row_df
      4  .write.mode("overwrite")
      5  .partitionBy("year","month")
      6  .parquet(f"{col_base}/clicks_parquet"))
      8 # Re-read columnar for fair comparison
----> 9 col_df = spark.read.schema(clicks_schema.add("year","int").add("month","int")).parquet(f"{col_base}/clicks_parquet")
     10 col_df.cache()
     11 print("Columnar rows:", col_df.count())

File ~/miniconda3/envs/de1-env/lib/python3.10/site-packages/pyspark/sql/types.py:1292, in StructType.add(self, field, data_ty
pe, nullable, metadata)
   1283     raise PySparkValueError(
   1284         errorClass="ARGUMENT_REQUIRED",
   1285         messageParameters={
   (...)
   1288         },
   1289     )
   1291 if isinstance(data_type, str):
-> 1292     data_type_f = _parse_datatype_json_value(data_type)
   1293 else:
   1294     data_type_f = data_type

File ~/miniconda3/envs/de1-env/lib/python3.10/site-packages/pyspark/sql/types.py:2042, in _parse_datatype_json_value(json_val
ue, fieldPath, collationsMap)
   2040         return VarcharType(int(m.group(1)))  # type: ignore[union-attr]
   2041     else:
-> 2042         raise PySparkValueError(
   2043             errorClass="CANNOT_PARSE_DATATYPE",
   2044             messageParameters={"msg": str(json_value)},
   2045         )
   2046 else:
   2047     tpe = json_value["type"]

PySparkValueError: [CANNOT_PARSE_DATATYPE] Unable to parse datatype. int.
```

Would you like to get notified about official Jupyter news?
Open privacy policy
Yes
No
Mode: Command
Ln 1, Col 1
DE1_Lab3_Notebook_EN.ipynb

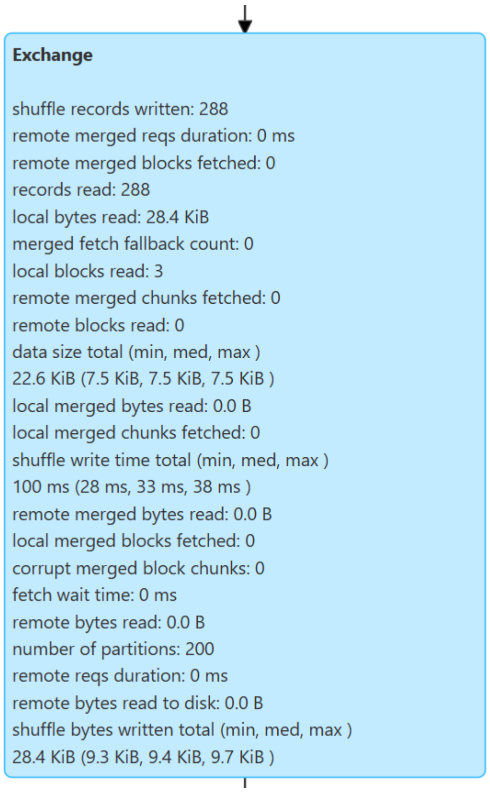
On a demandé à Chatgpt l’explication de cette erreur et il nous a proposé cette solution:



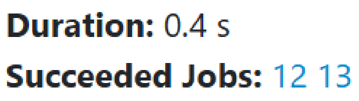
```
from pyspark.sql.types import IntegerType

col_base = "outputs/lab3/columnar"
# Write columnar
(row_df
 .write.mode("overwrite")
 .partitionBy("year","month")
 .parquet(f"{col_base}/clicks_parquet"))

# Re-read columnar for fair comparison
col_df = (
    spark.read
    .schema(clicks_schema.add("year", IntegerType()).add("month", IntegerType()))
    .parquet(f"{col_base}/clicks_parquet")
)

col_df.cache()
print("Columnar rows:", col_df.count())
```

```
Columnar rows: 15000
```

Cette exécution a permis d’obtenir une nouvelle requête `count` visible dans Spark UI,



```
Exchange

shuffle records written: 288
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 288
local bytes read: 28.4 KiB
merged fetch fallback count: 0
local blocks read: 3
remote merged chunks fetched: 0
remote blocks read: 0
data size total (min, med, max )
22.6 KiB (7.5 KiB, 7.5 KiB, 7.5 KiB )
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time total (min, med, max )
100 ms (28 ms, 33 ms, 38 ms )
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written total (min, med, max )
28.4 KiB (9.3 KiB, 9.4 KiB, 9.7 KiB )
```

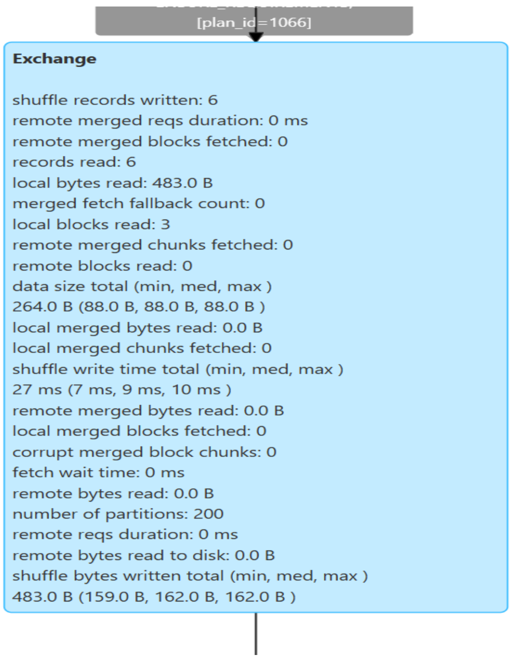


Duration: 0.4 s

Succeeded Jobs: 12 13

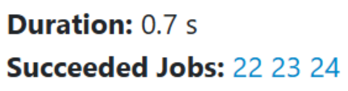
Pour les requêtes Q2 et Q3, la même logique a été appliquée, aussi bien pour la représentation row que column : chaque requête a été exécutée dans une cellule séparée, une action explicite (`count`) a été ajoutée afin de déclencher l’exécution Spark, les métriques ont été relevées à partir du nœud Exchange lorsque celui-ci était présent.

Q2_row



[plan_id=1066]

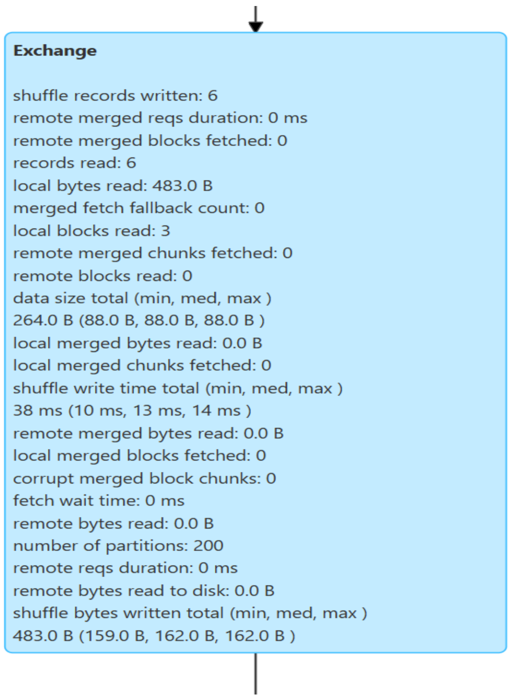
```
Exchange

shuffle records written: 6
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 6
local bytes read: 483.0 B
merged fetch fallback count: 0
local blocks read: 3
remote merged chunks fetched: 0
remote blocks read: 0
data size total (min, med, max )
264.0 B (88.0 B, 88.0 B, 88.0 B )
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time total (min, med, max )
27 ms (7 ms, 9 ms, 10 ms )
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written total (min, med, max )
483.0 B (159.0 B, 162.0 B, 162.0 B )
```



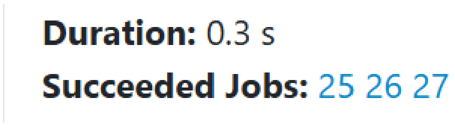
Duration: 0.7 s

Succeeded Jobs: 22 23 24

Q2_col



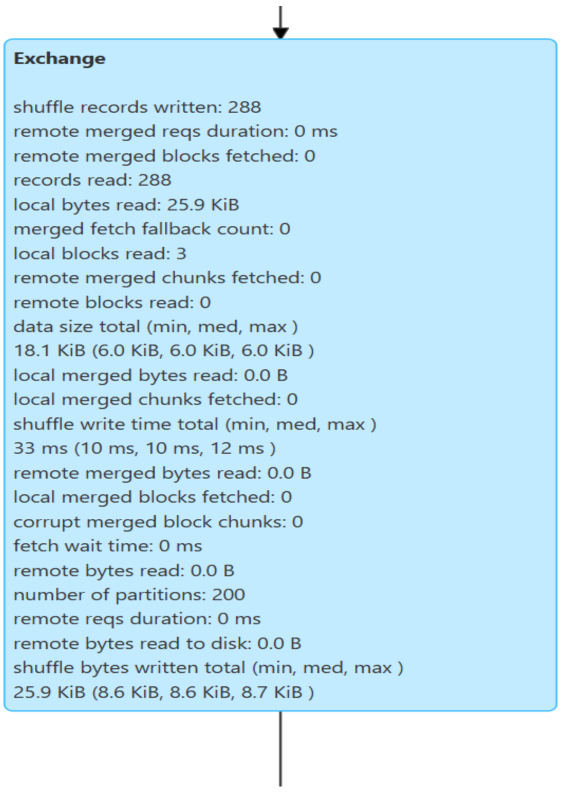
```
Exchange

shuffle records written: 6
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 6
local bytes read: 483.0 B
merged fetch fallback count: 0
local blocks read: 3
remote merged chunks fetched: 0
remote blocks read: 0
data size total (min, med, max )
264.0 B (88.0 B, 88.0 B, 88.0 B )
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time total (min, med, max )
38 ms (10 ms, 13 ms, 14 ms )
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written total (min, med, max )
483.0 B (159.0 B, 162.0 B, 162.0 B )
```



Duration: 0.3 s

Succeeded Jobs: 25 26 27

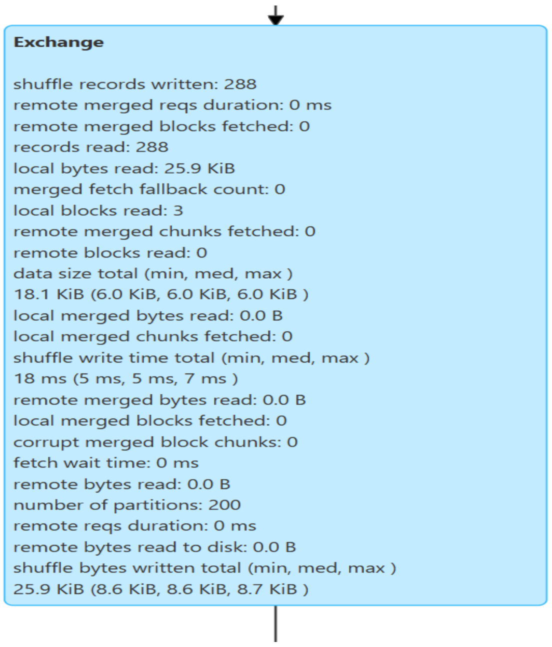
Q3_row



```
Exchange

shuffle records written: 288
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 288
local bytes read: 25.9 KiB
merged fetch fallback count: 0
local blocks read: 3
remote merged chunks fetched: 0
remote blocks read: 0
data size total (min, med, max )
18.1 KiB (6.0 KiB, 6.0 KiB, 6.0 KiB )
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time total (min, med, max )
33 ms (10 ms, 10 ms, 12 ms )
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written total (min, med, max )
25.9 KiB (8.6 KiB, 8.6 KiB, 8.7 KiB )
```

Duration: 0.4 s
Succeeded Jobs: 28 29

Q3_col

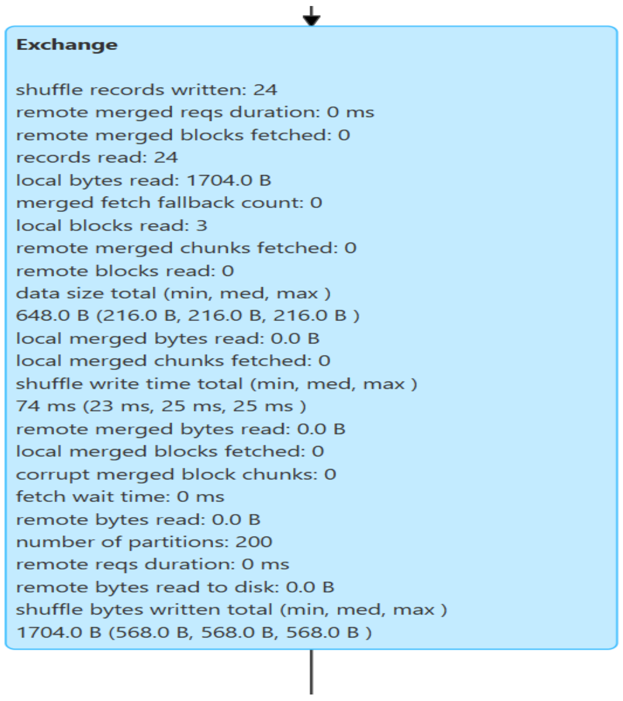


```
Exchange

shuffle records written: 288
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 288
local bytes read: 25.9 KiB
merged fetch fallback count: 0
local blocks read: 3
remote merged chunks fetched: 0
remote blocks read: 0
data size total (min, med, max )
18.1 KiB (6.0 KiB, 6.0 KiB, 6.0 KiB )
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time total (min, med, max )
18 ms (5 ms, 5 ms, 7 ms )
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written total (min, med, max )
25.9 KiB (8.6 KiB, 8.6 KiB, 8.7 KiB )
```

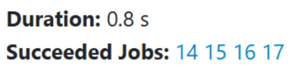
Duration: 0.1 s
Succeeded Jobs: 30 31

Dans cette partie “**Join strategy: normal vs broadcast**” deux stratégies de jointure ont été comparées :

Jointure normale (j1) :

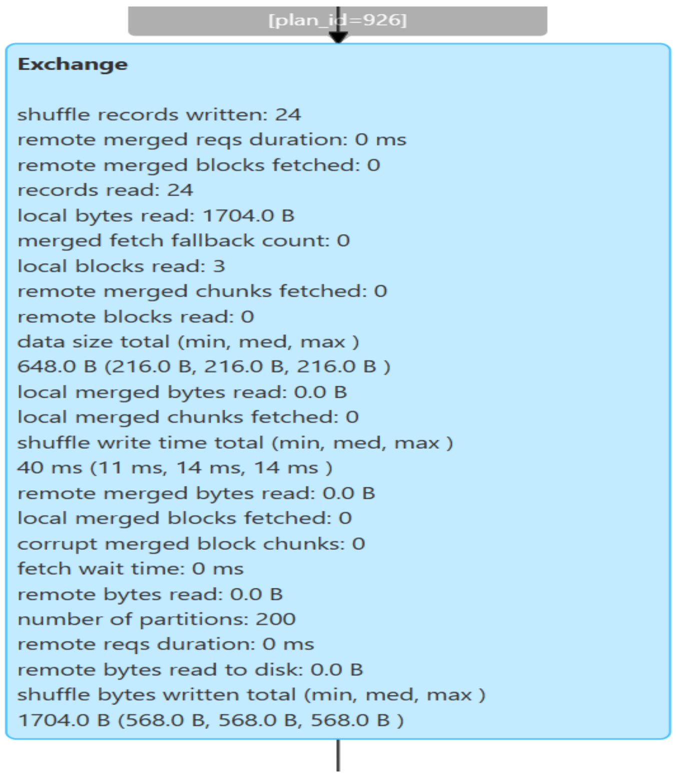


```
Exchange

shuffle records written: 24
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 24
local bytes read: 1704.0 B
merged fetch fallback count: 0
local blocks read: 3
remote merged chunks fetched: 0
remote blocks read: 0
data size total (min, med, max )
648.0 B (216.0 B, 216.0 B, 216.0 B )
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time total (min, med, max )
74 ms (23 ms, 25 ms, 25 ms )
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written total (min, med, max )
1704.0 B (568.0 B, 568.0 B, 568.0 B )
```

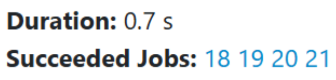


Duration: 0.8 s
Succeeded Jobs: 14 15 16 17

Jointure avec broadcast (j2) :

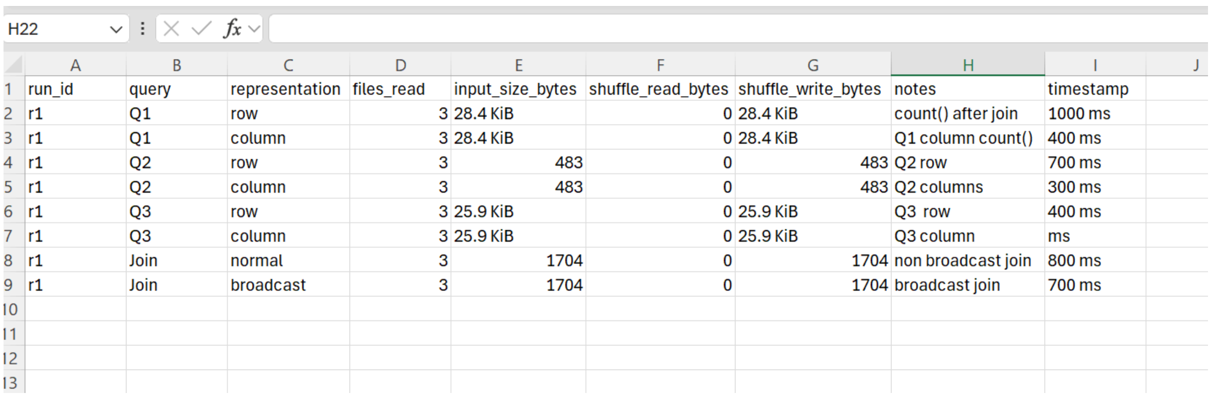


```
[plan_id=926]

Exchange

shuffle records written: 24
remote merged reqs duration: 0 ms
remote merged blocks fetched: 0
records read: 24
local bytes read: 1704.0 B
merged fetch fallback count: 0
local blocks read: 3
remote merged chunks fetched: 0
remote blocks read: 0
data size total (min, med, max )
648.0 B (216.0 B, 216.0 B, 216.0 B )
local merged bytes read: 0.0 B
local merged chunks fetched: 0
shuffle write time total (min, med, max )
40 ms (11 ms, 14 ms, 14 ms )
remote merged bytes read: 0.0 B
local merged blocks fetched: 0
corrupt merged block chunks: 0
fetch wait time: 0 ms
remote bytes read: 0.0 B
number of partitions: 200
remote reqs duration: 0 ms
remote bytes read to disk: 0.0 B
shuffle bytes written total (min, med, max )
1704.0 B (568.0 B, 568.0 B, 568.0 B )
```



Duration: 0.7 s
Succeeded Jobs: 18 19 20 21

Tableau final :



H22

run_id	query	representation	files_read	input_size_bytes	shuffle_read_bytes	shuffle_write_bytes	notes	timestamp
r1	Q1	row	3	28.4 KiB	0	28.4 KiB	count() after join	1000 ms
r1	Q1	column	3	28.4 KiB	0	28.4 KiB	Q1 column count()	400 ms
r1	Q2	row	3	483	0	483	Q2 row	700 ms
r1	Q2	column	3	483	0	483	Q2 columns	300 ms
r1	Q3	row	3	25.9 KiB	0	25.9 KiB	Q3 row	400 ms
r1	Q3	column	3	25.9 KiB	0	25.9 KiB	Q3 column	ms
r1	Join	normal	3	1704	0	1704	non broadcast join	800 ms
r1	Join	broadcast	3	1704	0	1704	broadcast join	700 ms

Fin du Lab03