

## Passo 1- Criar cluster

Compute > New compute >

### New Cluster

Cancel

Create Cluster

0 Workers: 0 GB Memory, 0 Cores, 0 DBU  
1 Driver: 15.3 GB Memory, 2 Cores, 1 DBU

Cluster name

fia\_trabalho\_grupo

Databricks runtime version

Runtime: 12.2 LTS (Scala 2.12, Spark 3.3.2)

Instance

Free 15 GB Memory: As a Community Edition user, your cluster will automatically terminate after an idle period of one or two hours. For [more configuration options](#), please [upgrade your Databricks subscription](#).

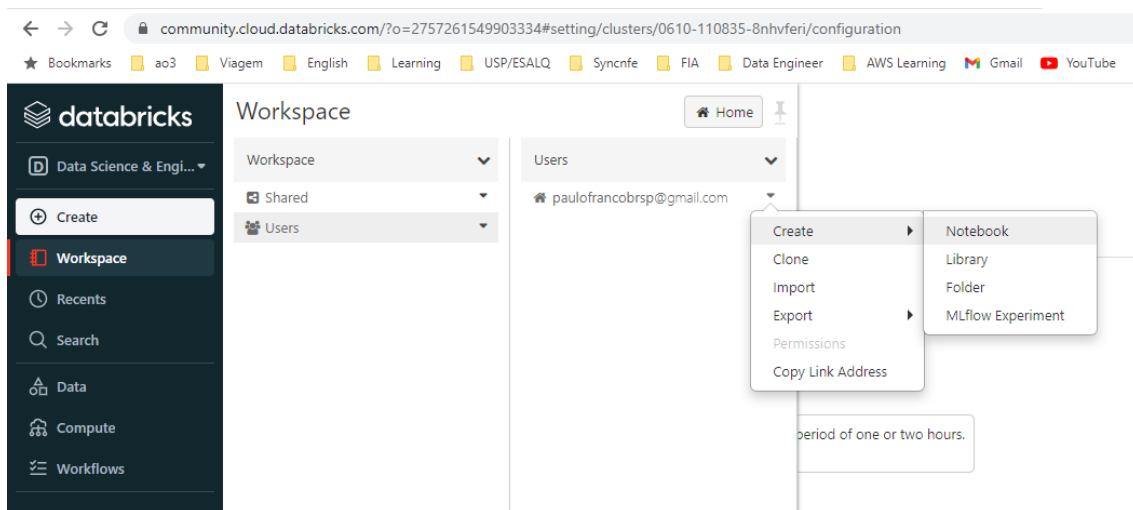
Instances

Spark

## Passo 2 – Montar bucket s3 para o Databricks

Executar somente após subida do Cluster

Crie um notebook no seu Workspace



## Passo 3 – Implemente o código abaixo no Notebook (Notebook disponível no Repositório – pasta Notebooks)

```
# import libraries
```

```
import pyspark.sql.functions as f
```

```
import urllib
```

```
# Credentials to access data-lake-fia bucket on s3
```

```
ACCESS_KEY = "AKIAQJ2IVF6JBGLXOBEF"
```

```
SECRET_KEY = "qy1M5alr/hxpNDgseObIPvwc0nC3ZZDR98SOM8XQ"
```

```
# Encoded secret key for security purposes
```

```
ENCODED_SECRET_KEY = urllib.parse.quote(SECRET_KEY, safe='')
```

```
# variables used to mount drive
```

```
AWS_S3_BUCKET = 'data-lake-fia'
```

```
MOUNT_NAME = '/mnt/data-lake-fia'
```

```
SOURCE_URL = f"s3n://{ACCESS_KEY}:{ENCODED_SECRET_KEY}@{AWS_S3_BUCKET}"
```

```
print(SOURCE_URL)
```

```
# Mount the drive
```

```
dbutils.fs.mount(SOURCE_URL, MOUNT_NAME)
```

```
# view content of s3 bucket
```

```
display(dbutils.fs.ls(MOUNT_NAME))
```

```
# Define location of parquet files (raw-data and context tier)
```

```
raw_tier_files = "/mnt/data-lake-fia/raw-data/datasus-imunizacao/"
```

```
context_tier_files = "/mnt/data-lake-fia/context/datasus_db/covid_dataset/"
```

```
# create Dataframe to make analysis
```

```
df_covid = spark.read.parquet(context_tier_files)
```

```
df_covid.display()
```

```
# view schema
```

```
df_covid.printSchema()
```