1. Processo de carga dos dados

A arquitetura do processo de carga dos dados escolhido será o modelo ETL.

O framework escolhido para o processamento dos dados será em PySpark.

Motivo de não desenvolver o projeto no modelo ELT

No cenário atual, onde os dados relacionais são armazenados em um banco de dados relacional como o postgres, talvez o modelo ideal seria o ELT, onde extrairíamos os dados em CSV, carregaríamos os dados diretamente no postgres utilizando os utilitários do postgres e todo o tratamento seria dentro do postgres.

A escolha do modelo ETL, é para exercitar o processamento de dados utilizando o motor de processamento do **Spark**.

Importação de bibliotecas

In [1]:
```
import os
import shutil
from pyspark.sql import SparkSession
from pyspark.sql.functions import from_unixtime, col, to_timestamp, coalesce
from pyspark.sql.types import StringType, IntegerType, LongType, DecimalType, DateType
```

Variaveis do projeto

In [2]:
```
#Diretorio dos arquivos csv
v_diretorio_csv='/usr/local/spark/csv/'

#Diretorio de export do arquivo de flatfile
v_diretorio_export='/home/jovyan/work/export'

#Variaveis de conexao com postgres
v_caminho_jar_postgres='/home/jovyan/work/jars/postgresql-9.4.1207.jar'
v_url_jdbc='jdbc:postgresql://postgres/projeto'
v_user_jdbc='airflow'
v_pass_jdbc='airflow'
```

Criando sessao e contexto

In [3]:
```
spark = (SparkSession
         .builder
         .master('local')
         .appName('load-postgres')
         # Add postgres jar
         .config('spark.driver.extraClassPath', v_caminho_jar_postgres)
         .getOrCreate())
sc = spark.sparkContext
```

```
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.spark.unsafe.Platform (file:/usr/local/spark-
3.1.2-bin-hadoop3.2/jars/spark-unsafe_2.12-3.1.2.jar) to constructor java.nio.DirectByteBuffer
(long,int)
WARNING: Please consider reporting this to the maintainers of org.apache.spark.unsafe.Platform
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access oper
ations
WARNING: All illegal access operations will be denied in a future release
22/12/22 20:28:48 WARN NativeCodeLoader: Unable to load native-hadoop library for your platfor
m... using builtin-java classes where applicable
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
```

Lendo arquivo csv, criando dataframe spark, formatando e criando views

Essa fase do processo, carrega os dados dos arquivos csv em dataframes, formata os campos e cria views para posteriormente serem utilizados na fase de tratamento dos dados.

In [4]:
```
#Dataframe Associado
df_associado_csv = (
    spark.read
    .format('csv')
    .option('header', True)
    .option('delimiter', ';')
    .load(v_diretorio_csv + 'associado.csv')
)

#Definindo o tipo da coluna
df_associado_csv_fmt = (
    df_associado_csv
    .withColumn('id', col('id').cast(IntegerType()))
    .withColumn('idade', col('idade').cast(IntegerType()))
)

#Criando view do dataframe
df_associado_csv_fmt.createOrReplaceTempView('associado')
```

In [5]:
```
df_associado_csv_fmt.show()
```

```
+---+---------+---------+-----+--------------------+
| id|     nome|sobrenome|idade|               email|
+---+---------+---------+-----+--------------------+
|  1|   Alícia|  Cardoso|   29|alícia.cardoso@ho...|
|  2|  Mirella|    Moura|   25|mirella.moura@gma...|
|  3|  Rodrigo|Fernandes|   54|rodrigo.fernandes...|
|  4|   Rebeca|  Cardoso|   59|rebeca.cardoso@te...|
|  5|     Raul|   Barros|   51|raul.barros@yahoo...|
|  6|    Julia|    Nunes|   38|julia.nunes@yahoo...|
|  7|     João|   Miguel|   45|joão.miguel@uol.c...|
|  8|Francisco|    Gomes|   27|francisco.gomes@h...|
|  9| Vinicius|     Lima|   58|vinicius.lima@hot...|
| 10|  Cecília|    Souza|   40|cecília.souza@uol...|
| 11|      Ana|    Julia|   57|ana.julia@yahoo.c...|
| 12|  Anthony|    Neves|   40|anthony.neves@yah...|
| 13|    Lucas|    Costa|   34|lucas.costa@hotma...|
| 14|      Ana| Teixeira|   66|ana.teixeira@hotm...|
| 15|     João|    Lucas|   70|joão.lucas@uol.co...|
| 16|    Bruna|      Luz|   69|bruna.luz@hotmail...|
| 17|    Vitor|     Hugo|   67|vitor.hugo@hotmai...|
| 18|    Sarah|Fernandes|   39|sarah.fernandes@y...|
| 19|  Cecília|Rodrigues|   75|cecília.rodrigues...|
| 20|   Nathan|     Mota|   42|nathan.mota@yahoo...|
+---+---------+---------+-----+--------------------+
only showing top 20 rows
```

In [6]:
```
#Dataframe Conta
df_conta_csv = (
    spark.read
    .format('csv')
    .option('header', True)
    .option('delimiter', ';')
    .load(v_diretorio_csv + 'conta.csv')
)

#Definindo o tipo da coluna
df_conta_csv_fmt = (
    df_conta_csv
```

```
    .withColumn('id', col('id').cast(IntegerType()))
    .withColumn('data_criacao', col('data_criacao').cast(DateType()))
    .withColumn('id_associado', col('id_associado').cast(IntegerType()))
)

#Criando view do dataframe
df_conta_csv_fmt.createOrReplaceTempView('conta')
```

In [7]:

```
df_conta_csv_fmt.show()
```

```
+---+--------------+-----------+------------+
| id|          tipo|data_criacao|id_associado|
+---+--------------+-----------+------------+
|  1|Conta Corrente|  2019-03-28|           1|
|  2|Conta Corrente|  2021-04-02|           2|
|  3|Conta Corrente|  2019-05-24|           3|
|  4|Conta Corrente|  2018-10-22|           4|
|  5|Conta Corrente|  2022-11-29|           5|
|  6|Conta Corrente|  2018-05-26|           6|
|  7|Conta Corrente|  2020-08-23|           7|
|  8|Conta Corrente|  2019-02-16|           8|
|  9|Conta Corrente|  2021-03-09|           9|
| 10|Conta Corrente|  2022-04-09|          10|
| 11|Conta Corrente|  2019-10-08|          11|
| 12|Conta Corrente|  2022-04-28|          12|
| 13|Conta Corrente|  2019-02-15|          13|
| 14|Conta Corrente|  2022-08-21|          14|
| 15|Conta Corrente|  2022-07-15|          15|
| 16|Conta Corrente|  2019-12-27|          16|
| 17|Conta Corrente|  2022-07-31|          17|
| 18|Conta Corrente|  2018-07-13|          18|
| 19|Conta Corrente|  2019-04-14|          19|
| 20|Conta Corrente|  2022-12-07|          20|
+---+--------------+-----------+------------+
only showing top 20 rows
```

In [8]:

```
#Dataframe Cartao
df_cartao_csv = (
    spark.read
    .format('csv')
    .option('header', True)
    .option('delimiter', ';')
    .load(v_diretorio_csv + 'cartao.csv')
)

#Definindo o tipo da coluna
df_cartao_csv_fmt = (
    df_cartao_csv
    .withColumn('id', col('id').cast(IntegerType()))
    .withColumn('id_conta', col('id_conta').cast(IntegerType()))
    .withColumn('id_associado', col('id_associado').cast(IntegerType()))
)

#Criando view do dataframe
df_cartao_csv_fmt.createOrReplaceTempView('cartao')
```

In [9]:

```
df_cartao_csv_fmt.show()
```

```
+---+----------------+----------------+--------+------------+
| id|      num_cartao|     nom_impresso|id_conta|id_associado|
+---+----------------+----------------+--------+------------+
|  1|8692002900010397|   ALÍCIA CARDOSO|       1|           1|
|  2|1360002500020347|    MIRELLA MOURA|       2|           2|
|  3|3935005400035103|RODRIGO FERNANDES|       3|           3|
|  4|4371005900041388|   REBECA CARDOSO|       4|           4|
```

```
|  5|9500005100053578|      RAUL BARROS|       5|          5|
|  6|7915003800066514|      JULIA NUNES|       6|          6|
|  7|2184004500079616|      JOÃO MIGUEL|       7|          7|
|  8|2631002700088038|  FRANCISCO GOMES|       8|          8|
|  9|3191005800091087|    VINICIUS LIMA|       9|          9|
| 10|9897004000108416|    CECÍLIA SOUZA|      10|         10|
| 11|8684005700115334|        ANA JULIA|      11|         11|
| 12|8694004000128933|    ANTHONY NEVES|      12|         12|
| 13|9950003400138288|      LUCAS COSTA|      13|         13|
| 14|4373006600142001|     ANA TEIXEIRA|      14|         14|
| 15|6333007000157004|       JOÃO LUCAS|      15|         15|
| 16|9080006900166160|        BRUNA LUZ|      16|         16|
| 17|6279006700177996|       VITOR HUGO|      17|         17|
| 18|5432003900184311|  SARAH FERNANDES|      18|         18|
| 19|4667007500196222|CECÍLIA RODRIGUES|      19|         19|
| 20|5578004200205193|      NATHAN MOTA|      20|         20|
+---+----------------+-----------------+--------+-----------+
only showing top 20 rows
```

In [10]:

```
#Dataframe Movimento
df_movimento_csv = (
    spark.read
    .format("csv")
    .option("header", True)
    .option("delimiter", ";")
    .load(v_diretorio_csv + "movimento.csv")
)

#Definindo o tipo da coluna
df_movimento_csv_fmt = (
    df_movimento_csv
    .withColumn('id', col('id').cast(IntegerType()))
    .withColumn('vlr_transacao', col('vlr_transacao').cast(DecimalType(10,2)))
    .withColumn('data_movimento', col('data_movimento').cast(DateType()))
    .withColumn('id_cartao', col('id_cartao').cast(IntegerType()))
)

#Criando view do dataframe
df_movimento_csv_fmt.createOrReplaceTempView('movimento')
```

In [11]:

```
df_movimento_csv_fmt.show()
```

```
+----+-------------+-----------------+--------------+---------+
|  id|vlr_transacao|    des_transacao|data_movimento|id_cartao|
+----+-------------+-----------------+--------------+---------+
|4249|        65.80|      Restaurante|    2022-05-11|       26|
|4250|        51.64|            Roupa|    2022-05-16|       26|
|4251|       398.16|Posto combustivel|    2022-05-21|       26|
|4252|        55.99|Posto combustivel|    2022-05-25|       26|
|4253|       218.14|         Farmacia|    2022-06-03|       26|
|4254|       543.76|      Restaurante|    2022-06-10|       26|
|4255|       495.44|      Restaurante|    2022-06-13|       26|
|4256|       123.16|         Pet shop|    2022-06-14|       26|
|4257|        35.05|            Roupa|    2022-06-17|       26|
|4258|        23.29|         Pet shop|    2022-06-22|       26|
|4259|       474.36|     Supermercado|    2022-06-23|       26|
|4260|        81.47|      Restaurante|    2022-06-25|       26|
|4261|       464.14|         Pet shop|    2022-07-02|       26|
|4262|       303.69|      Restaurante|    2022-07-04|       26|
|4263|       658.54|      Restaurante|    2022-07-10|       26|
|4264|       376.01|      Restaurante|    2022-07-12|       26|
|4265|       598.44|      Restaurante|    2022-07-13|       26|
|4266|        49.41|     Supermercado|    2022-07-25|       26|
|4267|       493.84|            Roupa|    2022-07-27|       26|
|4268|       499.91|            Roupa|    2022-07-28|       26|
+----+-------------+-----------------+--------------+---------+
```

```
only showing top 20 rows
```

In [12]:

```
#Dataframe Encerramento
df_encerramento_csv = (
    spark.read
    .format('csv')
    .option('header', True)
    .option('delimiter', ';')
    .load(v_diretorio_csv + 'encerramento_conta.csv')
)

#Removendo colunas
new_df_encerramento_csv=df_encerramento_csv.drop('semente', 'data_parou_comprar', 'dias_sem_com

#Definindo o tipo da coluna
df_encerramento_csv_fmt = (
    new_df_encerramento_csv
    .withColumn('id', col('id').cast(IntegerType()))
    .withColumn('data_criacao', col('data_criacao').cast(DateType()))
    .withColumn('data_encerramento', col('data_encerramento').cast(DateType()))
)

#Criando view do dataframe
df_encerramento_csv_fmt.createOrReplaceTempView('encerramento_conta')
```

In [13]:

```
df_encerramento_csv_fmt.show()
```

```
+---+------------+-----------------+
| id|data_criacao|data_encerramento|
+---+------------+-----------------+
|  1|  2019-03-28|             null|
|  2|  2021-04-02|             null|
|  3|  2019-05-24|             null|
|  4|  2018-10-22|             null|
|  5|  2022-11-29|             null|
|  6|  2018-05-26|             null|
|  7|  2020-08-23|             null|
|  8|  2019-02-16|             null|
|  9|  2021-03-09|             null|
| 10|  2022-04-09|             null|
| 11|  2019-10-08|             null|
| 12|  2022-04-28|             null|
| 13|  2019-02-15|             null|
| 14|  2022-08-21|             null|
| 15|  2022-07-15|             null|
| 16|  2019-12-27|             null|
| 17|  2022-07-31|             null|
| 18|  2018-07-13|             null|
| 19|  2019-04-14|             null|
| 20|  2022-12-07|             null|
+---+------------+-----------------+
only showing top 20 rows
```

In [14]:

```
#Dataframe Fatura
df_fatura_csv = (
    spark.read
    .format('csv')
    .option('header', True)
    .option('delimiter', ';')
    .load(v_diretorio_csv + 'fatura.csv')
)

#Definindo o tipo da coluna
df_fatura_csv_fmt = (
```

```
    df_fatura_csv
    .withColumn('id', col('id').cast(IntegerType()))
    .withColumn('data_vencimento_fatura', col('data_vencimento_fatura').cast(DateType()))
    .withColumn('vlr_fatura', col('vlr_fatura').cast(DecimalType(10,2)))
    .withColumn('data_pagamento_fatura', col('data_pagamento_fatura').cast(DateType()))
    .withColumn('qtd_dias_atraso_pgto', col('qtd_dias_atraso_pgto').cast(IntegerType()))
    .withColumn('id_cartao', col('id_cartao').cast(IntegerType()))
)

#Criando view do dataframe
df_fatura_csv_fmt.createOrReplaceTempView('fatura')
```

In [15]:

```
df_fatura_csv_fmt.show()
```

```
+---+---------------------+---------+--------------------+-------------------+---------+
| id|data_vencimento_fatura|vlr_fatura|data_pagamento_fatura|qtd_dias_atraso_pgto|id_cartao|
+---+---------------------+---------+--------------------+-------------------+---------+
|  1|           2019-03-15|      0.00|          2019-03-15|                  0|        1|
|  2|           2019-04-15|   1470.86|          2019-04-11|                  0|        1|
|  3|           2019-05-15|   1634.88|          2019-05-11|                  0|        1|
|  4|           2019-06-15|    437.91|          2019-06-11|                  0|        1|
|  5|           2019-07-15|   1006.45|          2019-07-14|                  0|        1|
|  6|           2019-08-15|    932.13|          2019-08-12|                  0|        1|
|  7|           2019-09-15|    693.69|          2019-09-12|                  0|        1|
|  8|           2019-10-15|    349.34|          2019-10-12|                  0|        1|
|  9|           2019-11-15|   1609.69|          2019-11-11|                  0|        1|
| 10|           2019-12-15|   1456.59|          2019-12-14|                  0|        1|
| 11|           2020-01-15|   1129.89|          2020-01-14|                  0|        1|
| 12|           2020-02-15|    186.44|          2020-02-15|                  0|        1|
| 13|           2020-03-15|   1350.80|          2020-03-11|                  0|        1|
| 14|           2020-04-15|    880.98|          2020-04-12|                  0|        1|
| 15|           2020-05-15|   1493.61|          2020-05-12|                  0|        1|
| 16|           2020-06-15|    518.68|          2020-06-10|                  0|        1|
| 17|           2020-07-15|    958.56|          2020-07-11|                  0|        1|
| 18|           2020-08-15|    210.13|          2020-08-15|                  0|        1|
| 19|           2020-09-15|   1059.57|          2020-09-11|                  0|        1|
| 20|           2020-10-15|   1224.60|          2020-10-13|                  0|        1|
+---+---------------------+---------+--------------------+-------------------+---------+
only showing top 20 rows
```

Funcoes de carga dados do banco de dados do target e criando views das chaves

Essa fase do processo, serão carregados as chaves das tabelas do banco de dados do postgres em dataframes e views, para validacao se registro ja existe na base e validacao de integridade de relacionamento entre tabelas.

In [16]:

```
#Funcao para carregar as chaves
def f_carrega_associado_tgt():
    #Carregando dataframe com dados do banco de target
    df_associado_tgt = (
        spark.read
        .format('jdbc')
        .option('url', v_url_jdbc)
        .option('query', 'select id from target.associado')
        .option('user', v_user_jdbc)
        .option('password', v_pass_jdbc)
        .load()
    )

    #Criando view do dataframe
    df_associado_tgt.createOrReplaceTempView('associado_tgt')
```

In [17]:

```
#Funcao para carregar as chaves
def f_carrega_conta_tgt():
```

```
    #Carregando dataframe com dados do banco de target
    df_conta_tgt = (
        spark.read
        .format('jdbc')
        .option('url', v_url_jdbc)
        .option("query", 'select id from target.conta')
        .option('user', v_user_jdbc)
        .option('password', v_pass_jdbc)
        .load()
    )

    #Criando view do dataframe
    df_conta_tgt.createOrReplaceTempView('conta_tgt')
```

In [18]:

```
#Funcao para carregar as chaves
def f_carrega_cartao_tgt():
    #Carregando dataframe com dados do banco de target
    df_cartao_tgt = (
        spark.read
        .format('jdbc')
        .option('url', v_url_jdbc)
        .option("query", 'select id from target.cartao')
        .option('user', v_user_jdbc)
        .option('password', v_pass_jdbc)
        .load()
    )

    #Criando view do dataframe
    df_cartao_tgt.createOrReplaceTempView('cartao_tgt')

f_carrega_cartao_tgt()
```

In [19]:

```
#Funcao para carregar as chaves
def f_carrega_movimento_tgt():
    #Carregando dataframe com dados do banco de target
    df_movimento_tgt = (
        spark.read
        .format('jdbc')
        .option('url', v_url_jdbc)
        .option("query", 'select id from target.movimento')
        .option('user', v_user_jdbc)
        .option('password', v_pass_jdbc)
        .load()
    )

    #Criando view do dataframe
    df_movimento_tgt.createOrReplaceTempView('movimento_tgt')

f_carrega_movimento_tgt()
```

In [20]:

```
#Funcao para carregar as chaves
def f_carrega_encerramento_tgt():
    #Carregando dataframe com dados do banco de target
    df_encerramento_tgt = (
        spark.read
        .format('jdbc')
        .option('url', v_url_jdbc)
        .option("query", 'select id from target.encerramento_conta')
        .option('user', v_user_jdbc)
        .option('password', v_pass_jdbc)
        .load()
    )

    #Criando view do dataframe
    df_encerramento_tgt.createOrReplaceTempView('encerramento_conta_tgt')
```

```
f_carrega_encerramento_tgt()
```

In [21]:

```
#Funcao para carregar as chaves
def f_carrega_fatura_tgt():
    #Carregando dataframe com dados do banco de target
    df_fatura_tgt = (
        spark.read
        .format('jdbc')
        .option('url', v_url_jdbc)
        .option("query", 'select id from target.fatura')
        .option('user', v_user_jdbc)
        .option('password', v_pass_jdbc)
        .load()
    )

    #Criando view do dataframe
    df_fatura_tgt.createOrReplaceTempView('fatura_tgt')

f_carrega_fatura_tgt()
```

Verificacao, tratamento e carga de dados

Essa fase do processo, serão validados os dados,corrigidos integridade de relacionamento entre os dados e carregados os dados no banco de dados do target.

A estrategia de carga de dados, sera:

Caso o dado exista, nao sobrescrever, caso seja um registro novo, inserir.

Carga de dados do Associado

In [22]:

```
#Verificacao se a chave ja existe, caso nao exista, insere na tabela de target
f_carrega_associado_tgt()
df_associado_novo=spark.sql('''
    select
        wrk.*
    from associado wrk

    left join associado_tgt tgt
    on tgt.id=wrk.id

    where tgt.id is null
''')

#Inserindo dados novos
(df_associado_novo.write
    .format('jdbc')
    .option('url', v_url_jdbc)
    .option('dbtable', 'target.associado')
    .option('user', v_user_jdbc)
    .option('password', v_pass_jdbc)
    .mode('append')
    .save()
)

#Atualizando as chaves da view do target
f_carrega_associado_tgt()
```

Carga de dados da Conta Corrente

In [23]:

```
#Validacao e correcao de relacionamento entre a conta e o associado.
#Caso o associado nao exista, sera informado -1 na coluna.
```

```
df_conta_tratado=spark.sql('''
    select
        cco.id,
        cco.tipo,
        cco.data_criacao,
        coalesce(ass.id, -1) as id_associado
    from conta cco

    left join associado_tgt ass
    on ass.id=cco.id_associado
''')

#Criando view do dataframe
df_conta_tratado.createOrReplaceTempView('conta')
```

In [24]:

```
qtd_associados_invalidos=df_conta_tratado.filter(df_conta_tratado.id_associado==-1).count()

print(f"Associados invalidos: {qtd_associados_invalidos}. ")
```

```
[Stage 17:=================================================>   (186 + 1) / 200]
Associados invalidos: 51.
```

In [25]:

```
#Verificacao se a chave ja existe, caso nao exista, insere na tabela de target
f_carrega_conta_tgt()
df_conta_nova=spark.sql('''
    select
        wrk.*
    from conta wrk

    left join conta_tgt tgt
    on tgt.id=wrk.id

    where tgt.id is null
''')

#Inserindo dados novos
(df_conta_nova.write
    .format('jdbc')
    .option('url', v_url_jdbc)
    .option('dbtable', 'target.conta')
    .option('user', v_user_jdbc)
    .option('password', v_pass_jdbc)
    .mode('append').save()
)

#Atualizando as chaves da view do target
f_carrega_conta_tgt()
```

Carga de dados de Cartao

In [26]:

```
#Validacao e correcao de relacionamento entre a cartao, conta e o associado.
#Caso o associado/conta nao exista, sera informado -1 na coluna.

df_cartao_tratado=spark.sql('''
    select
        car.id,
        car.num_cartao,
        car.nom_impresso,
        coalesce(cco.id, -1) as id_conta,
        coalesce(ass.id, -1) as id_associado

    from cartao car
```

```
    left join conta_tgt cco
    on cco.id=car.id_conta

    left join associado_tgt ass
    on ass.id=car.id_associado
''')
```

In [27]:

```
qtd_contas_invalidos=df_cartao_tratado.filter(df_cartao_tratado.id_conta == -1).count()
qtd_associados_invalidos=df_cartao_tratado.filter(df_cartao_tratado.id_associado == -1).count()

print(f"Contas invalidas: {qtd_contas_invalidos}. Associados invalidos: {qtd_associados_invalid
```

```
[Stage 34:=====================================================(200 + 0) / 200]
Contas invalidas: 1. Associados invalidos: 51.
```

In [28]:

```
#Verificacao se a chave ja existe, caso nao exista, insere na tabela de target
f_carrega_cartao_tgt()
df_cartao_nova=spark.sql('''
    select
        wrk.*
    from cartao wrk

    left join cartao_tgt tgt
    on tgt.id=wrk.id

    where tgt.id is null
''')

#Inserindo dados novos
(df_cartao_nova.write
    .format('jdbc')
    .option('url', v_url_jdbc)
    .option('dbtable', 'target.cartao')
    .option('user', v_user_jdbc)
    .option('password', v_pass_jdbc)
    .mode('append').save()
)

#Atualizando as chaves da view do target
f_carrega_cartao_tgt()
```

Carga de dados de Movimento

In [29]:

```
#Validacao e correcao de relacionamento entre a movimento e cartao.
#Caso o cartao nao exista, sera informado -1 na coluna.

df_movimento_tratado=spark.sql('''
    select
        mov.id,
        mov.vlr_transacao,
        mov.des_transacao,
        mov.data_movimento,
        coalesce(car.id, -1) as id_cartao

    from movimento mov

    left join cartao_tgt car
    on car.id=mov.id_cartao
''')
```

In [30]:
```
qtd_cartoes_invalidos=df_movimento_tratado.filter(df_movimento_tratado.id_cartao == -1).count()

print(f"Cartoes invalidos: {qtd_cartoes_invalidos}. ")
```

```
[Stage 41:=============================================>          (160 + 1) / 200]
Cartoes invalidos: 0.
```

In [31]:
```
#Verificacao se a chave ja existe, caso nao exista, insere na tabela de target
f_carrega_movimento_tgt()
df_movimento_nova=spark.sql('''
    select
        wrk.*
    from movimento wrk

    left join movimento_tgt tgt
    on tgt.id=wrk.id

    where tgt.id is null
''')

#Inserindo dados novos
(df_movimento_nova.write
    .format('jdbc')
    .option('url', v_url_jdbc)
    .option('dbtable', 'target.movimento')
    .option('user', v_user_jdbc)
    .option('password', v_pass_jdbc)
    .mode('append').save()
)

#Atualizando as chaves da view do target
f_carrega_movimento_tgt()
```

Carga de dados de Encerramento da Conta

In [32]:
```
#Verificacao se a chave ja existe, caso nao exista, insere na tabela de target
f_carrega_encerramento_tgt()
df_encerramento_nova=spark.sql('''
    select
        wrk.*
    from encerramento_conta wrk

    left join encerramento_conta_tgt tgt
    on tgt.id=wrk.id

    where tgt.id is null
''')

#Inserindo dados novos
(df_encerramento_nova.write
    .format('jdbc')
    .option('url', v_url_jdbc)
    .option('dbtable', 'target.encerramento_conta')
    .option('user', v_user_jdbc)
    .option('password', v_pass_jdbc)
    .mode('append').save()
)

#Atualizando as chaves da view do target
f_carrega_encerramento_tgt()
```

Carga de dados da Fatura dos cartoes

In [33]:
```
#Validacao e correcao de relacionamento entre a fatura e cartao.
#Caso o cartao nao exista, sera informado -1 na coluna.

df_fatura_tratado=spark.sql('''
    select
        fat.id,
        fat.data_vencimento_fatura,
        fat.vlr_fatura,
        fat.data_pagamento_fatura,
        fat.qtd_dias_atraso_pgto,
        coalesce(car.id, -1) as id_cartao

    from fatura fat

    left join cartao car
    on car.id=fat.id_cartao
''')
```

In [34]:
```
qtd_cartoes_fat_invalidos=df_fatura_tratado.filter(df_fatura_tratado.id_cartao == -1).count()

print(f"Cartoes invalidos: {qtd_cartoes_fat_invalidos}. ")
```

```
Cartoes invalidos: 0.
```

In [35]:
```
#Verificacao se a chave ja existe, caso nao exista, insere na tabela de target
f_carrega_fatura_tgt()
df_fatura_nova=spark.sql('''
    select
        wrk.*
    from fatura wrk

    left join fatura_tgt tgt
    on tgt.id=wrk.id

    where tgt.id is null
''')

#Inserindo dados novos
(df_fatura_nova.write
    .format('jdbc')
    .option('url', v_url_jdbc)
    .option('dbtable', 'target.fatura')
    .option('user', v_user_jdbc)
    .option('password', v_pass_jdbc)
    .mode('append').save()
)

#Atualizando as chaves da view do target
f_carrega_fatura_tgt()
```

Geração do arquivo Flat

O objetivo do exercício, é exercitar o processamento no Spark. A extração dos dados e relacionamentos entre os dados, serão processados no **Spark**.

Por essa razão, os dados não serão relacionados e tratados no **postgres**.

In [36]:
```
#Carregando dados no Dataframe
df_associado_tgt = (
```

```
    spark.read
    .format('jdbc')
    .option('url', v_url_jdbc)
    .option('dbtable', 'target.associado')
    .option('user', v_user_jdbc)
    .option('password', v_pass_jdbc)
    .load()
)

#Criando view do dataframe
df_associado_tgt.createOrReplaceTempView('associado')
```

In [37]:

```
#Carregando dados no Dataframe
df_conta_tgt = (
    spark.read
    .format('jdbc')
    .option('url', v_url_jdbc)
    .option('dbtable', 'target.conta')
    .option('user', v_user_jdbc)
    .option('password', v_pass_jdbc)
    .load()
)

#Criando view do dataframe
df_conta_tgt.createOrReplaceTempView('conta')
```

In [38]:

```
#Carregando dados no Dataframe
df_cartao_tgt = (
    spark.read
    .format('jdbc')
    .option('url', v_url_jdbc)
    .option('dbtable', 'target.cartao')
    .option('user', v_user_jdbc)
    .option('password', v_pass_jdbc)
    .load()
)

#Criando view do dataframe
df_cartao_tgt.createOrReplaceTempView('cartao')
```

In [39]:

```
#Carregando dados no Dataframe
df_movimento_tgt = (
    spark.read
    .format('jdbc')
    .option('url', v_url_jdbc)
    .option('dbtable', 'target.movimento')
    .option('user', v_user_jdbc)
    .option('password', v_pass_jdbc)
    .load()
)

#Criando view do dataframe
df_movimento_tgt.createOrReplaceTempView('movimento')
```

Obs: 1 - *Somente será trabalhado com associados válidos.* 2 - *Na estrutura de dados gerados, não foi repassado a coluna de data da criação do cartão, temos somente a data de criação da conta, por essa razão, será utilizada a data de criação da conta, como data de criação do cartão.*

In [40]:

```
df_flat_file=spark.sql('''
    select
        ass.nome as nome_associado,
        ass.sobrenome as sobrenome_associado,
        ass.idade as idade_associado,
```

```
        mov.vlr_transacao as vlr_transacao_movimento,
        mov.des_transacao as des_transacao_movimento,
        mov.data_movimento as data_movimento,
        car.num_cartao as numero_cartao,
        car.nom_impresso as nome_impresso_cartao,
        cco.data_criacao as data_criacao_cartao,
        cco.tipo as tipo_conta,
        cco.data_criacao as data_criacao_conta

    from associado ass

    left join cartao car
    on car.id_associado=ass.id

    left join movimento mov
    on mov.id_cartao=car.id

    left join conta cco
    on cco.id_associado=ass.id

    where ass.id>0

''')
```

In [41]:

```
df_flat_file.show()
```

```
[Stage 60:==============================================>       (173 + 2) / 200]
+--------------+------------------+--------------+----------------------+------------------
----+--------------+---------------+--------------------+------------------+--------------+--
----------------+
|nome_associado|sobrenome_associado|idade_associado|vlr_transacao_movimento|des_transacao_movim
ento|data_movimento|   numero_cartao|nome_impresso_cartao|data_criacao_cartao|    tipo_conta|da
ta_criacao_conta|
+--------------+------------------+--------------+----------------------+------------------
----+--------------+---------------+--------------------+------------------+--------------+--
----------------+
|     Valentina|             Fogaça|            23|                159.36|           Supermer
cado|    2018-12-13|4250002301488069|    VALENTINA FOGAÇA|        2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|            23|                241.92|            Restaur
ante|    2020-06-06|4250002301488069|    VALENTINA FOGAÇA|        2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|            23|                154.46|                 R
oupa|    2020-06-16|4250002301488069|    VALENTINA FOGAÇA|        2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|            23|                188.04|            Restaur
ante|    2020-09-03|4250002301488069|    VALENTINA FOGAÇA|        2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|            23|                196.40|      Posto combust
ivel|    2019-08-24|4250002301488069|    VALENTINA FOGAÇA|        2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|            23|                 59.40|            Restaur
ante|    2020-03-26|4250002301488069|    VALENTINA FOGAÇA|        2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|            23|                225.65|            Restaur
ante|    2020-12-02|4250002301488069|    VALENTINA FOGAÇA|        2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|            23|                  2.88|               Pet
shop|    2018-12-20|4250002301488069|    VALENTINA FOGAÇA|        2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|            23|                 41.43|      Posto combust
ivel|    2020-02-13|4250002301488069|    VALENTINA FOGAÇA|        2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|            23|                271.61|                 R
oupa|    2020-04-23|4250002301488069|    VALENTINA FOGAÇA|        2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|            23|                 63.81|            Restaur
ante|    2019-02-02|4250002301488069|    VALENTINA FOGAÇA|        2018-08-14|Conta Corrente|
```

```
2018-08-14|
|     Valentina|             Fogaça|             23|                242.82|      Posto combust
ivel|    2020-12-24|4250002301488069|    VALENTINA FOGAÇA|         2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|             23|                 66.45|           Supermer
cado|    2020-02-14|4250002301488069|    VALENTINA FOGAÇA|         2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|             23|                293.68|               Farm
acia|    2021-05-25|4250002301488069|    VALENTINA FOGAÇA|         2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|             23|                134.12|            Restaur
ante|    2021-06-07|4250002301488069|    VALENTINA FOGAÇA|         2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|             23|                 31.12|                  R
oupa|    2018-12-10|4250002301488069|    VALENTINA FOGAÇA|         2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|             23|                378.84|                  R
oupa|    2019-11-11|4250002301488069|    VALENTINA FOGAÇA|         2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|             23|                122.03|               Farm
acia|    2021-09-04|4250002301488069|    VALENTINA FOGAÇA|         2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|             23|                178.75|               Pet
shop|    2018-12-11|4250002301488069|    VALENTINA FOGAÇA|         2018-08-14|Conta Corrente|
2018-08-14|
|     Valentina|             Fogaça|             23|                252.25|                  R
oupa|    2019-11-25|4250002301488069|    VALENTINA FOGAÇA|         2018-08-14|Conta Corrente|
2018-08-14|
+--------------+-------------------+---------------+----------------------+-------------------
----+--------------+---------------+--------------------+-------------------+--------------+--
---------------+
only showing top 20 rows
```

In [42]:

```
#Funcao para exportar um dataframe para arquivo csv
def f_exporta_flat_file(df, arquivo_csv):
    #Gera arquivo flat
    df.coalesce(1).write.mode('overwrite').options(header='True', delimiter=';').csv(v_diretori

    #Renomeia arquivo e move para a pasta export
    for arquivo in os.listdir(v_diretorio_export + '/flatfile/'):
        a_arquivo=arquivo.split('.')
        if(len(a_arquivo)==2 and a_arquivo[1]=='csv'):
            shutil.move(v_diretorio_export + '/flatfile/' + arquivo, v_diretorio_export + '/' +
            print('Arquivo flat gerado: ' + v_diretorio_export + '/' + arquivo_csv)

f_exporta_flat_file(df_flat_file, 'movimento_flat.csv')
```

```
[Stage 68:>                                                         (0 + 1) / 1]
Arquivo flat gerado: /home/jovyan/work/export/movimento_flat.csv
```

In [ ]: