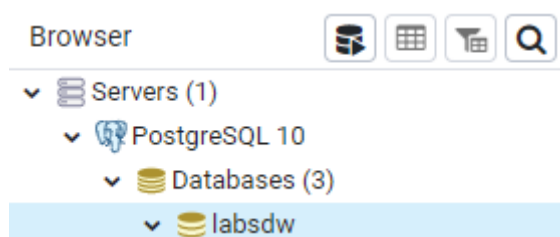
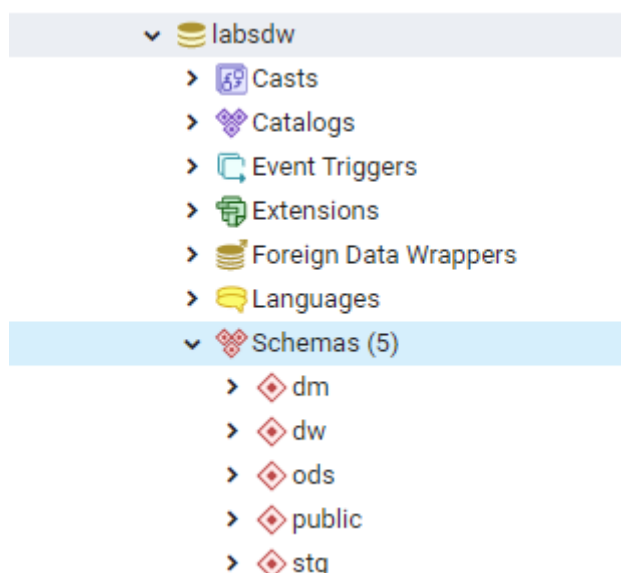


Exercício 01 – Repositório

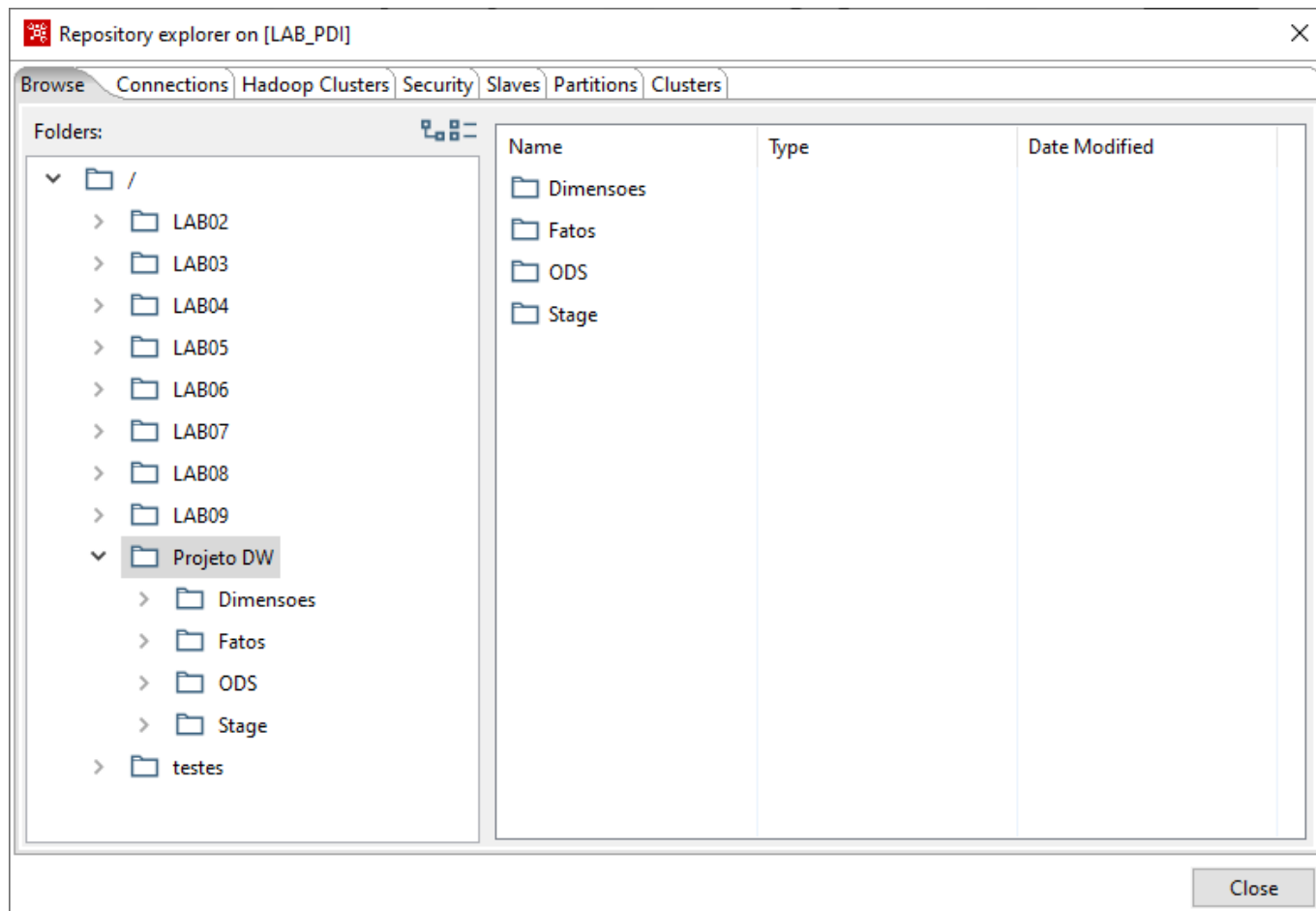
- 1.1** Esta primeira etapa já foi concluída, a criação de um repositório, e idealmente, o serviço do Pentaho deve rodar em um servidor, não apenas pela distribuição das cargas, mas também pela possibilidade de agendamento de tarefas. Crie um novo banco de dados no Postgres, chamado “labsdw”:



- 1.2** Neste novo banco de dados, crie os esquemas “DM”, “DW”, “ODS” e “STG”, que serão usados para armazenar as tabelas e objetos que serão criadas ao longo do projeto. O esquema ODS não será utilizado neste exercício, mas haverá uma explicação sobre sua função e exemplos práticos de uso:

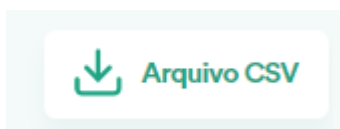


- 1.3** Em seu repositório do Pentaho, crie uma nova pasta chamada “Projeto DW”, contendo as subpastas “Dimensões”, “Fatos”, “Stage” e “ODS”:



Exercício 02 – Base de Dados

- 2.1** Vá até o repositório do GIT (<https://github.com/hgsilva/posbi-etl/tree/master/DATASETS>), baixe o arquivo “[HIST PAINEL COVIDBR 28set2020.7z](#)” e o descompacte. Esta base de dados foi extraída do Portal Covid, do Min. da Saúde, em 28/09/2020, e pode ser baixado novamente para atualizar seus dados a partir do endereço <https://covid.saude.gov.br/> clicando no botão “Arquivo CSV”.
- Esta base de dados será utilizada para a montagem de nosso Data Mart, com fatos e dimensões a serem exploradas a partir de seus dados.



2.2 Crie uma nova transformação, chamada “05 – Stage Carrega Base” e carregue este arquivo em uma tabela localizada no esquema de dados “STG” do banco de dados “labsdw”, e o execute:

2.2.1 Você deverá criar uma nova conexão, apontando para o novo banco de dados.

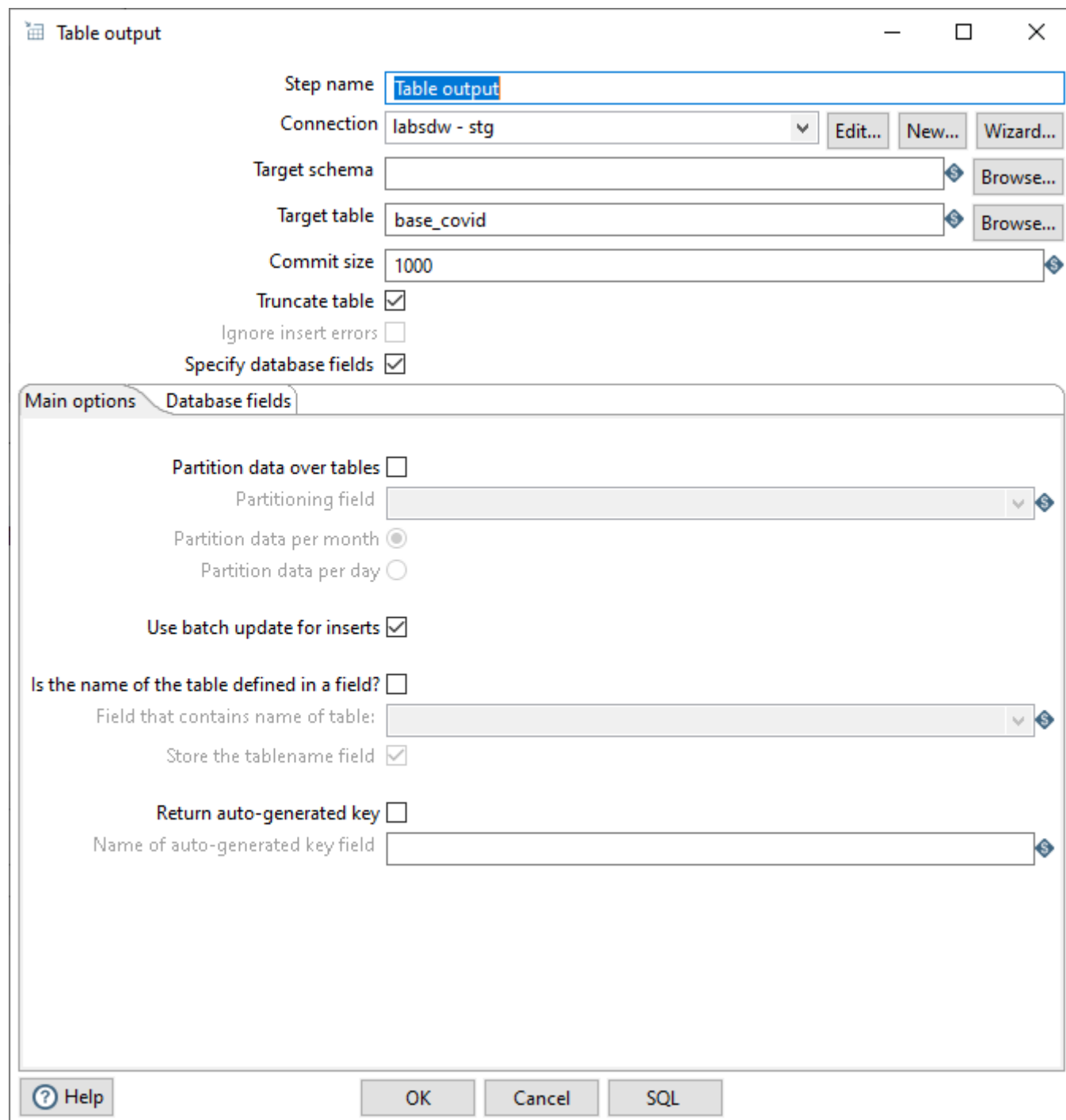


Table output

Step name:

Connection: Edit... New... Wizard...

Target schema: Browse...

Target table: Browse...

Commit size:

Truncate table: ☒

Ignore insert errors: ☐

Specify database fields: ☒

Main options | **Database fields**

Partition data over tables: ☐

Partitioning field:

Partition data per month: ☒

Partition data per day: ☐

Use batch update for inserts: ☒

Is the name of the table defined in a field?: ☐

Field that contains name of table:

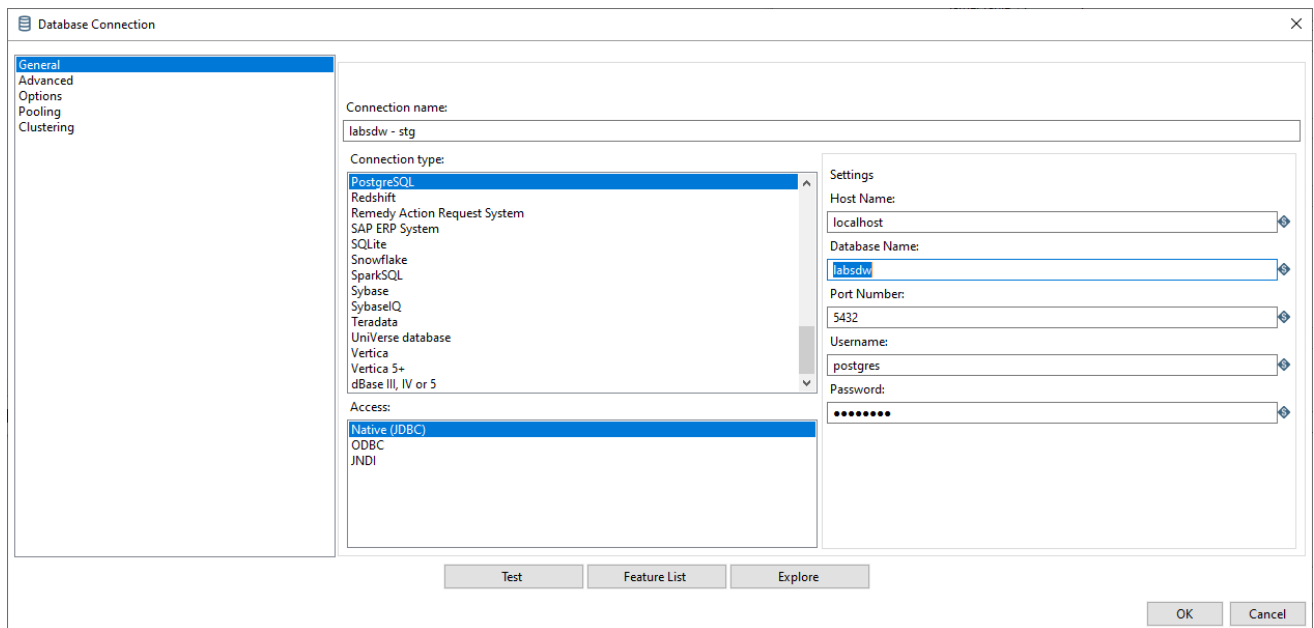
Store the tablename field: ☒

Return auto-generated key: ☐

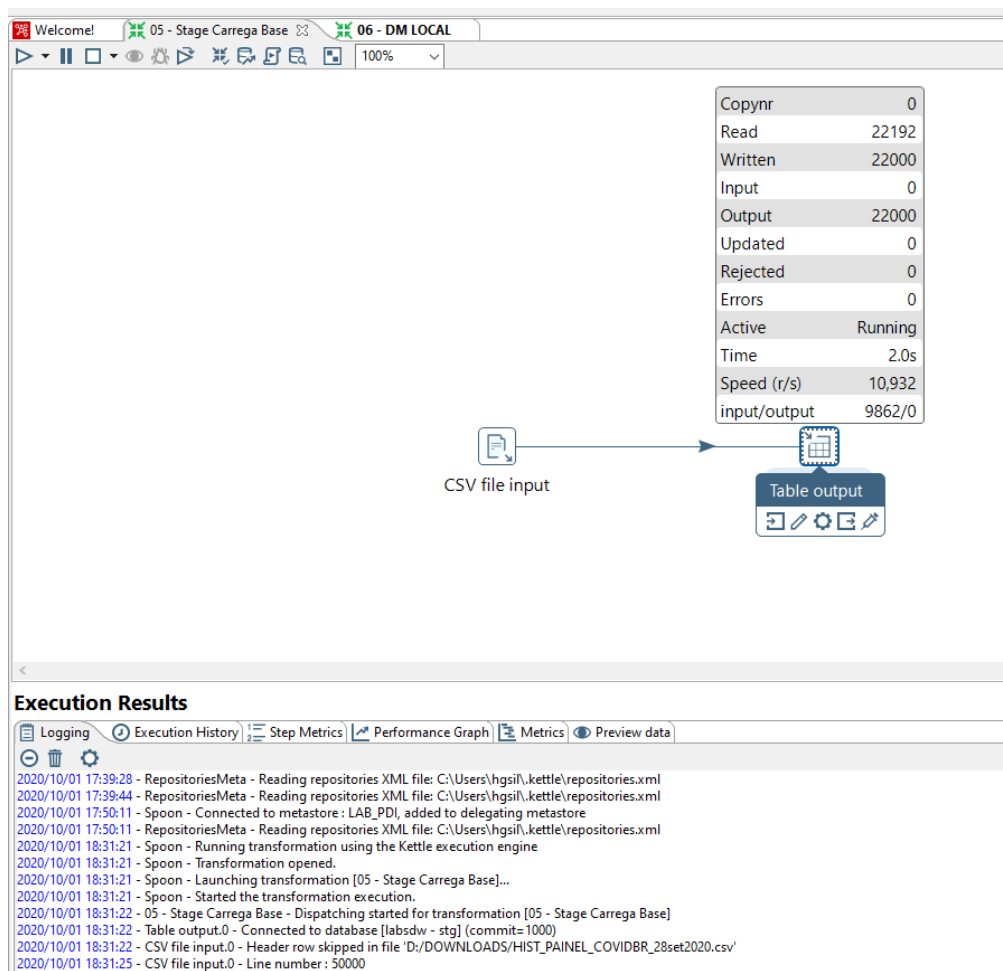
Name of auto-generated key field:

Help OK Cancel SQL

2.2.2 Detalhes da nova conexão de banco de dados:



2.2.3 Execute o processamento e observe a transformação:



Execution Results

Logging Execution History Step Metrics Performance Graph Metrics Preview data

2020/10/01 17:39:28 - RepositoriesMeta - Reading repositories XML file: C:\Users\hgsil\kettle\repositories.xml
 2020/10/01 17:39:44 - RepositoriesMeta - Reading repositories XML file: C:\Users\hgsil\kettle\repositories.xml
 2020/10/01 17:50:11 - Spoon - Connected to metastore: LAB_PDI, added to delegating metastore
 2020/10/01 17:50:11 - RepositoriesMeta - Reading repositories XML file: C:\Users\hgsil\kettle\repositories.xml
 2020/10/01 18:31:21 - Spoon - Running transformation using the Kettle execution engine
 2020/10/01 18:31:21 - Spoon - Transformation opened.
 2020/10/01 18:31:21 - Spoon - Launching transformation [05 - Stage Carrega Base]...
 2020/10/01 18:31:21 - Spoon - Started the transformation execution.
 2020/10/01 18:31:22 - 05 - Stage Carrega Base - Dispatching started for transformation [05 - Stage Carrega Base]
 2020/10/01 18:31:22 - Table output.0 - Connected to database [labsw - stg] (commit=1000)
 2020/10/01 18:31:22 - CSV file input.0 - Header row skipped in file 'D:\DOWNLOADS\HIST_PAINEL_COVIDBR_28set2020.csv'
 2020/10/01 18:31:25 - CSV file input.0 - Line number : 50000