

Professor:
Fabiano Tavares



ETL - LAB I

ANO: 2019

LAB 1 - PDI

Repositório, Criação de Jobs e Transformação

Pré-requisitos

- ❖ JAVA;
- ❖ Instalar Pentaho Data Integration;
- ❖ Possuir o diretório de “Bases” de teste;
- ❖ Instalar MySQL;

Exercício 1

Criando a primeira Transformação

Exercício 1 – Criando a primeira Transformação

1 – Crie uma pasta chamada `etl_labs` no seu computador.

2 – Abra o Spoon.

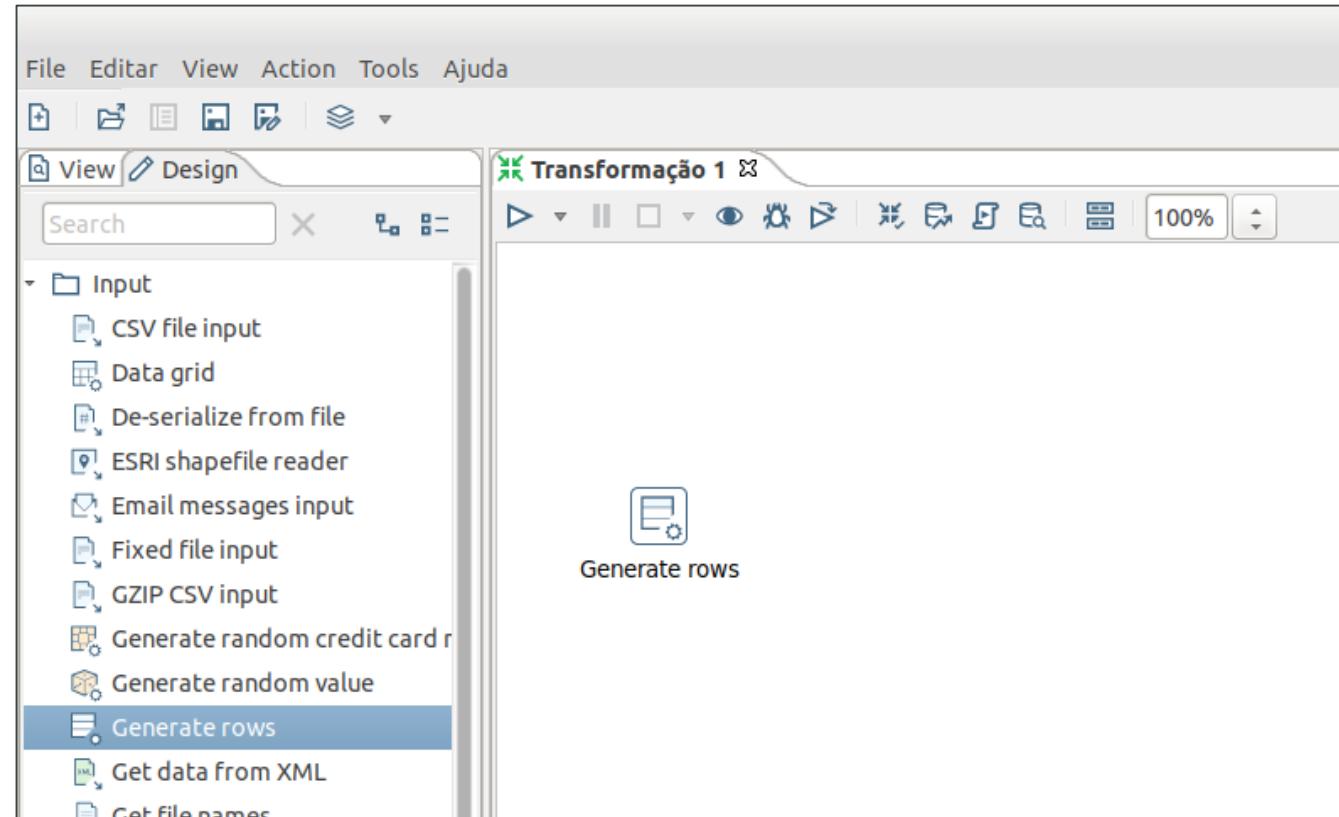
3 – A partir do menu principal escolha **Arquivo -> Nova Transformação**. Será mostrada a transformação *Transformation 1* na área de trabalho.

Obs.: Caso deseje, utilize as teclas de atalho CTRL-N

4 – Do lado direito da tela é mostrada a árvore de *steps*. Clique na opção *Design* e Expanda a opção *Input*.

Exercício 1 – Criando a primeira Transformação

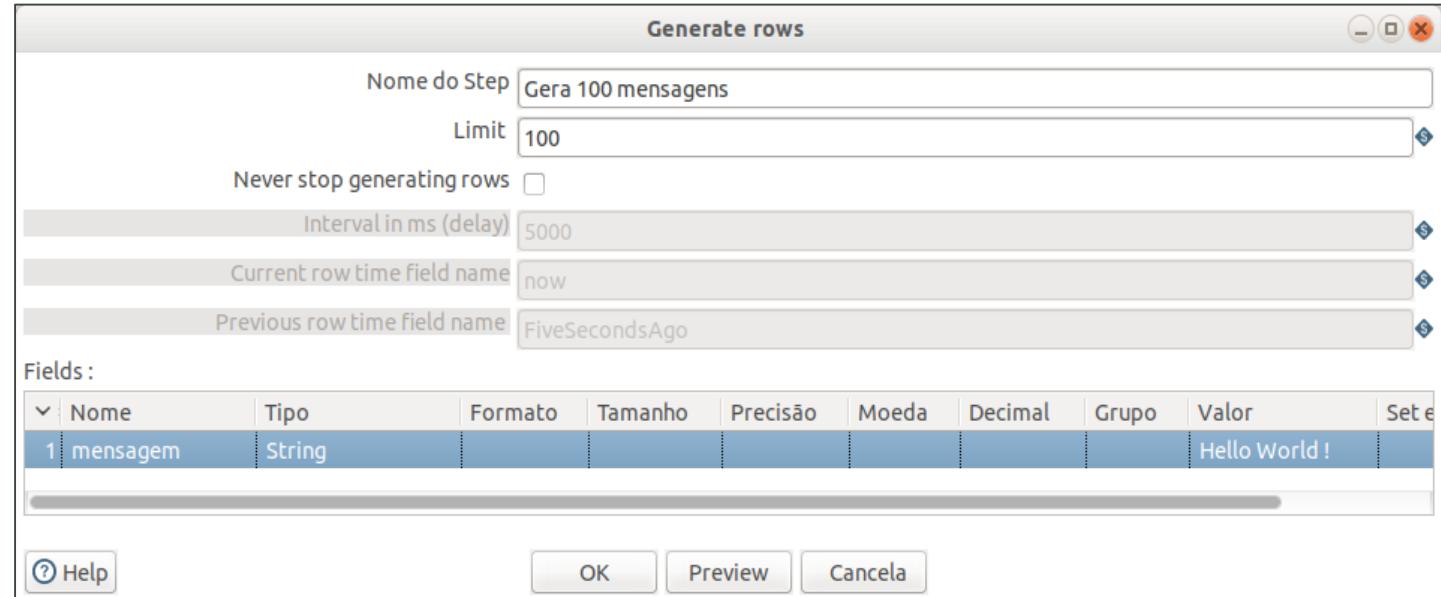
5 – Arraste e solte o ícone do step *Generate Rows* para a área de trabalho da transformação.



Exercício 1 – Criando a primeira Transformação

6 – Dê um duplo-clique para abrir o step e digite os dados abaixo e clique OK em seguida:

- Nome do step: Gera 100 mensagens
- Em Fields:
 - Nome: mensagem
 - Tipo: String
 - Valor: Hello World!

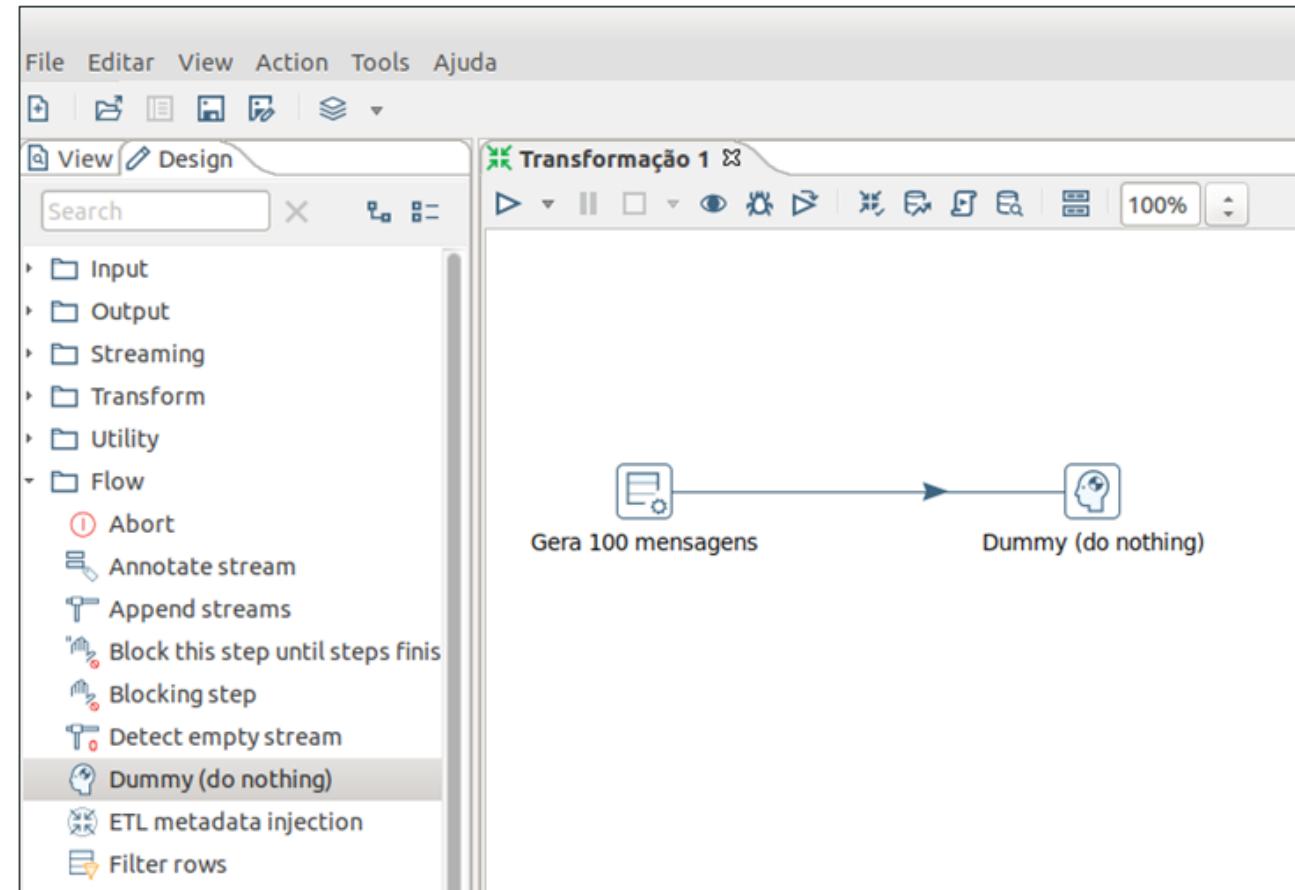


Obs.: Você também pode abrir o step para edição clicando com o botão direito e escolhendo a opção *Editar Step*.

Exercício 1 – Criando a primeira Transformação

7 – Na árvore de steps, expanda a opção *Flow* e arraste e solte o ícone do step *Dummy* para a área de trabalho da transformação.

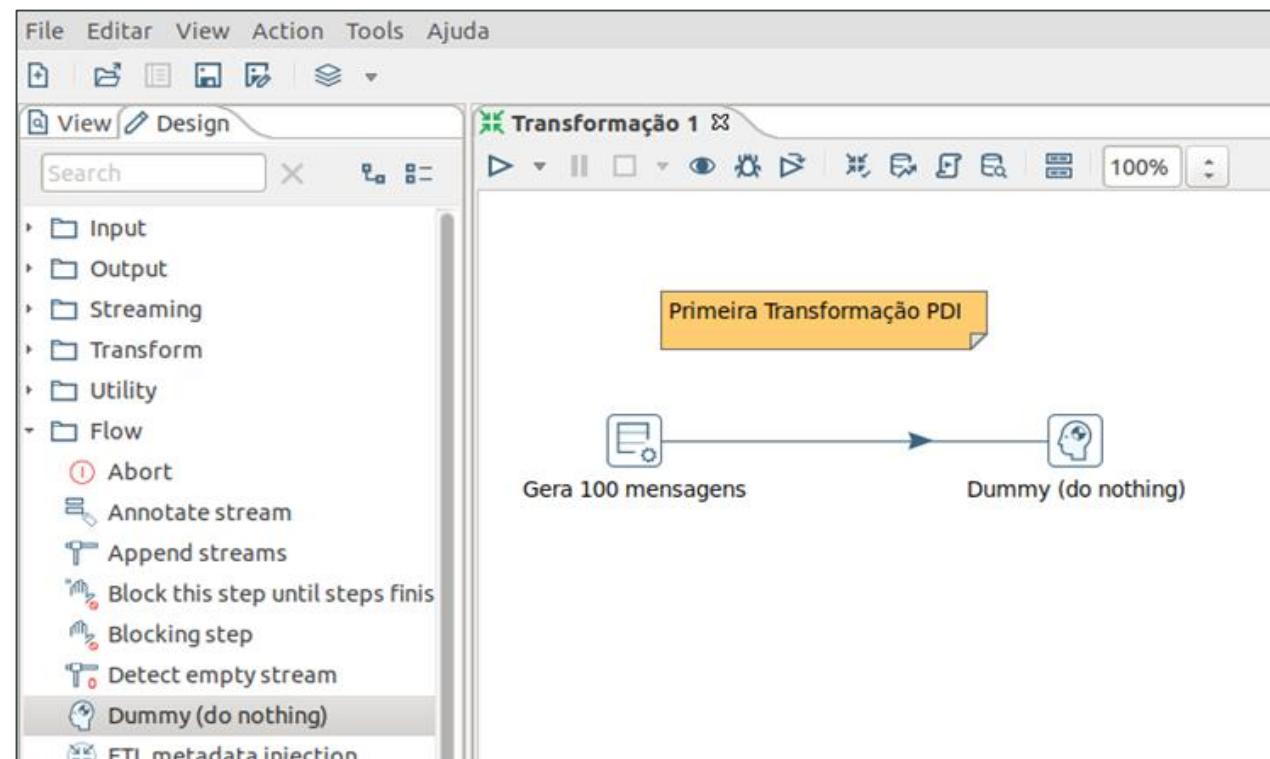
8 – Vamos criar um *hop* para ligar os dois steps. Clique com o botão direito no step *Generate Rows*, segure a tecla *Shift* e arraste o cursor do mouse para o step *Dummy*.



Exercício 1 – Criando a primeira Transformação

9 – Vamos adicionar uma nota à transformação. Clique com o botão direito em qualquer parte da área de trabalho da transformação e escolha a opção *Nova Nota*. Digite o texto da nota.

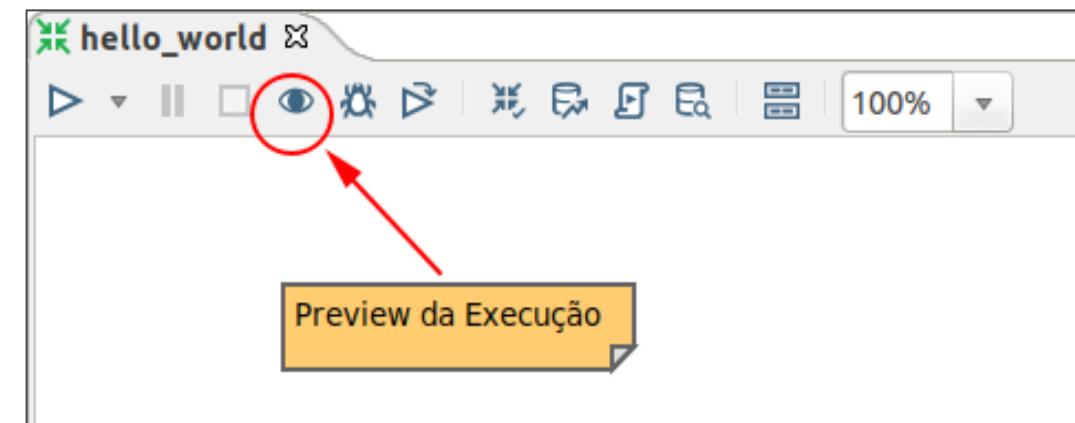
10 – Salve a transformação na pasta *etl_labs* com o nome *hello_world*. Será gerado um arquivo XML com extensão .ktr.



Exercício 1 – Criando a primeira Transformação

11 – Podemos rodar uma prévia da transformação, antes de sua execução real. Clique com o botão direito no step *Dummy* e escolha a opção *Preview*.

Obs.: Você poderá rodar a previsão da transformação clicando no step desejado e, em seguida, clicando no ícone *Preview* da barra de opções.



Exercício 1 – Criando a primeira Transformação

12 – Após carregar a janela de diálogo de *Preview*, clique no botão *Quick Launch* para visualizar o resultado da transformação.

13 – O que foi feito nessa transformação?

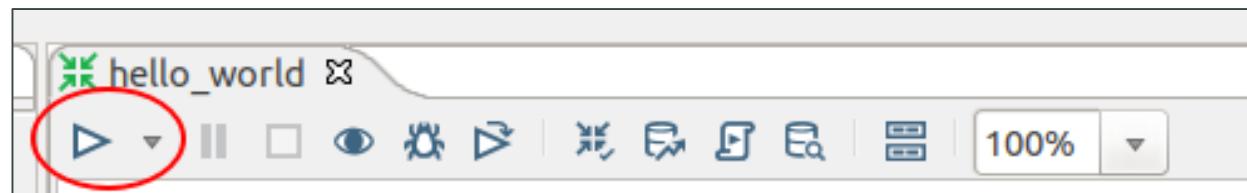
O step *Generate Rows* simplesmente gera várias linhas de registros, de acordo com os parâmetros informados. Experimente editar o step para acrescentar outros campos e mudar o tamanho do conjunto de registros gerados. O step *Dummy* apenas recebe os registros gerados e não faz nada.

Examine preview data	
Rows of step: Dummy (do nothing) (10 rows)	
▼	mensagem
1	Hello World !
2	Hello World !
3	Hello World !
4	Hello World !
5	Hello World !
6	Hello World !
7	Hello World !
8	Hello World !
9	Hello World !
10	Hello World !

[Close](#) [Stop](#) [Get more rows](#)

Exercício 1 – Criando a primeira Transformação

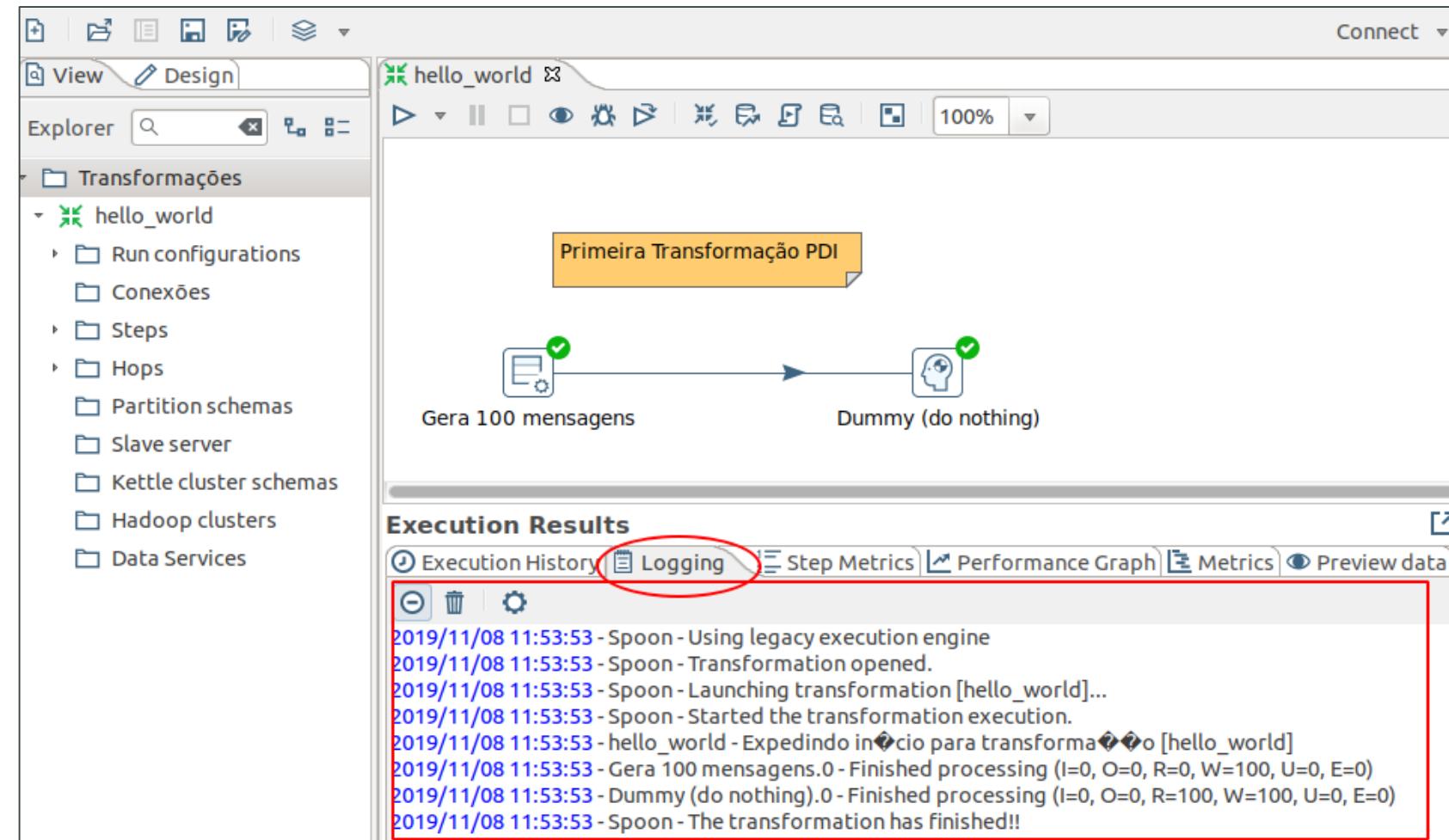
14 – Um pequeno detalhe: nós não executamos a transformação, apenas visualizamos uma prévia de sua execução. Para executar a transformação clique no botão *Run* e após abrir a janela de diálogo clique no botão *Launch*.



Exercício 1 – Criando a primeira Transformação

15 – O resultado real da execução aparece na aba do log. As métricas de execução aparecem na aba da janela do log. O nível de detalhe do log de execução pode ser configurado na tela anterior.

Obs.: na execução de grandes volumes de dados recomenda-se deixar o nível do log como *Basic*.



Exercício 2

Criando a transformação “Hello World Fulano”

Exercício 2 – Criando a transformação “Hello World Fulano”

1 – Vamos expandir a transformação do exercício anterior para obter dados de uma fonte, fazer uma pequena transformação nos dados e armazenar o resultado em um arquivo texto. Crie um arquivo texto com os dados abaixo:

Fulano

Sicrano

Beltrano

Salve o arquivo na pasta *etl_labs* com o nome *fulano.txt*.

2 – Crie uma nova transformação e salve-a com o nome de *hello_world_fulano.ktr*. O objetivo dessa transformação é ler o arquivo texto acima, adicionar a frase “Hello World” e uma sequência para cada frase criada. Ao final, todas as frases criadas serão gravadas em um arquivo texto.

Exercício 2 – Criando a transformação “Hello World Fulano”

1 – Vamos expandir a transformação do exercício anterior para obter dados de uma fonte, fazer uma pequena transformação nos dados e armazenar o resultado em um arquivo texto. Crie um arquivo texto com os dados abaixo:

Fulano

Sicrano

Beltrano

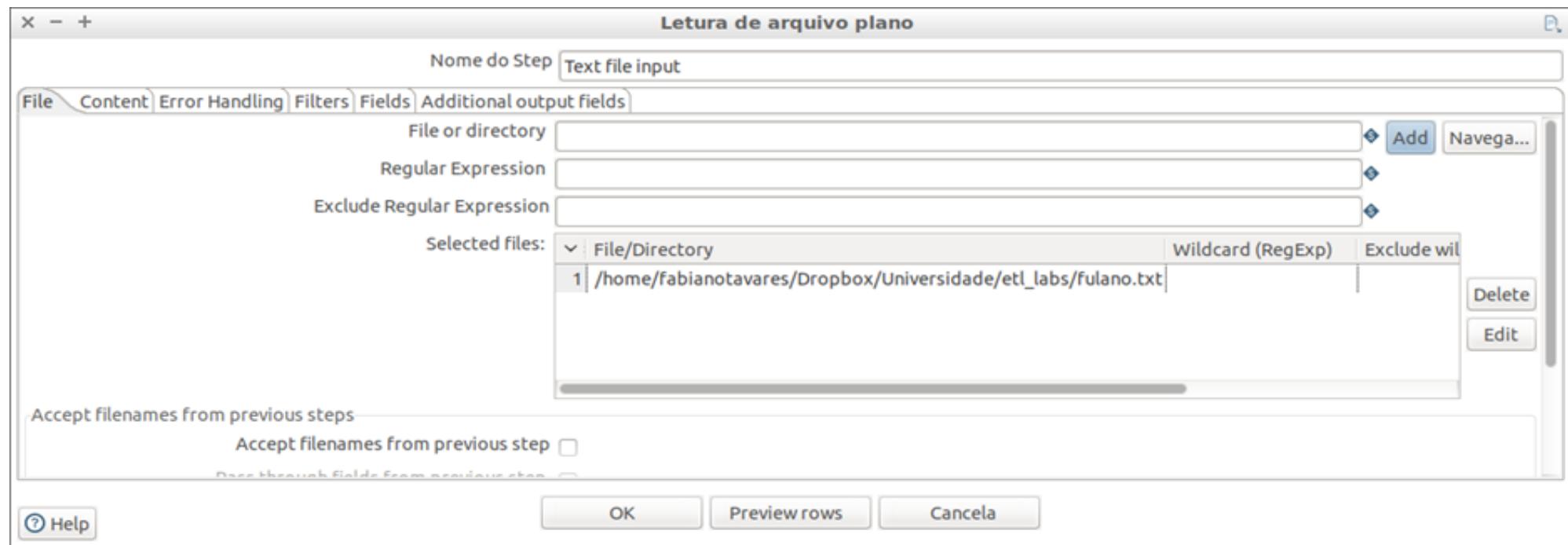
Salve o arquivo na pasta *etl_labs* com o nome *fulano.txt*.

2 – Crie uma nova transformação e salve-a com o nome de *hello_world_fulano.ktr*. O objetivo dessa transformação é ler o arquivo texto acima, adicionar a frase “Hello World” e uma sequência para cada frase criada. Ao final, todas as frases criadas serão gravadas em um arquivo texto.

Exercício 2 – Criando a transformação “Hello World Fulano”

3 – Adicione um step *Text file input* da opção *Input*. Edite o step com as seguintes informações:

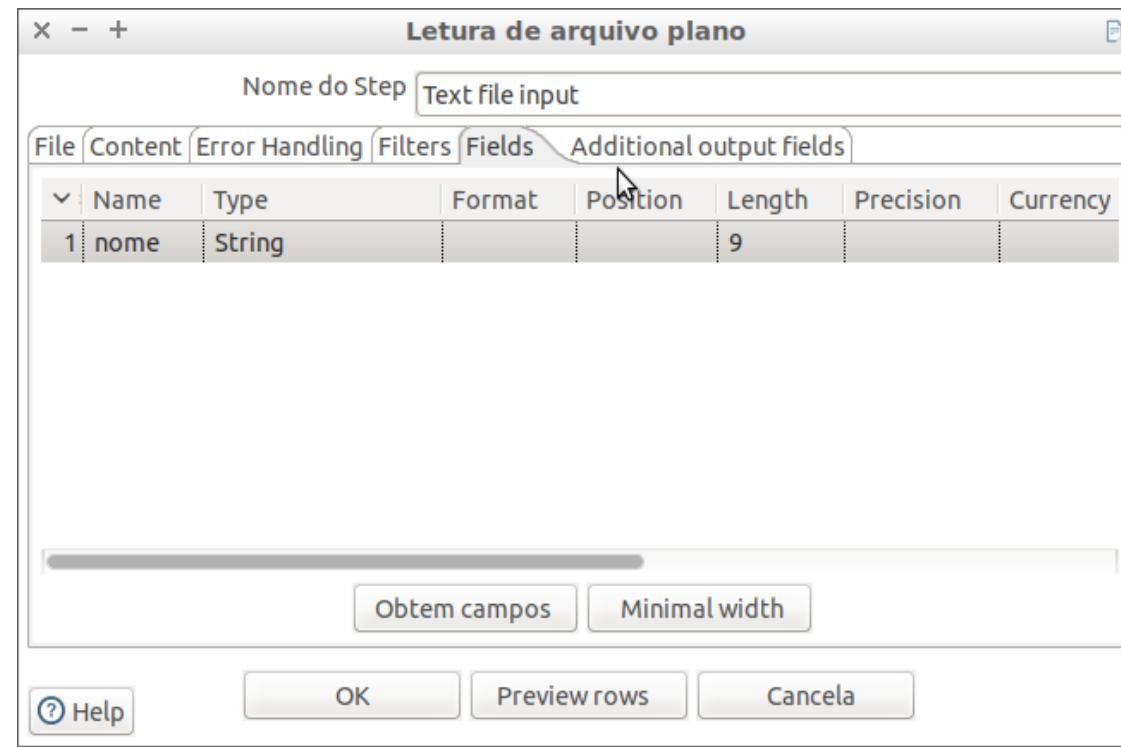
-Na aba *File*, campo *File or directory*, clique no botão *Navegar* para localizar o arquivo *fulano.txt*. Em seguida, clique no botão *Add* para colocar o arquivo na lista.



Exercício 2 – Criando a transformação “Hello World Fulano”

Na aba *Content*, desmarque a opção *Header*. Se você não fizer isso, o *Spoon* entenderá que a primeira linha do arquivo texto é um cabeçalho com o nome de um ou mais campos.

Na aba *Fields*, digite o nome do campo (*nome*) e o seu tipo (*String*). Clique OK.



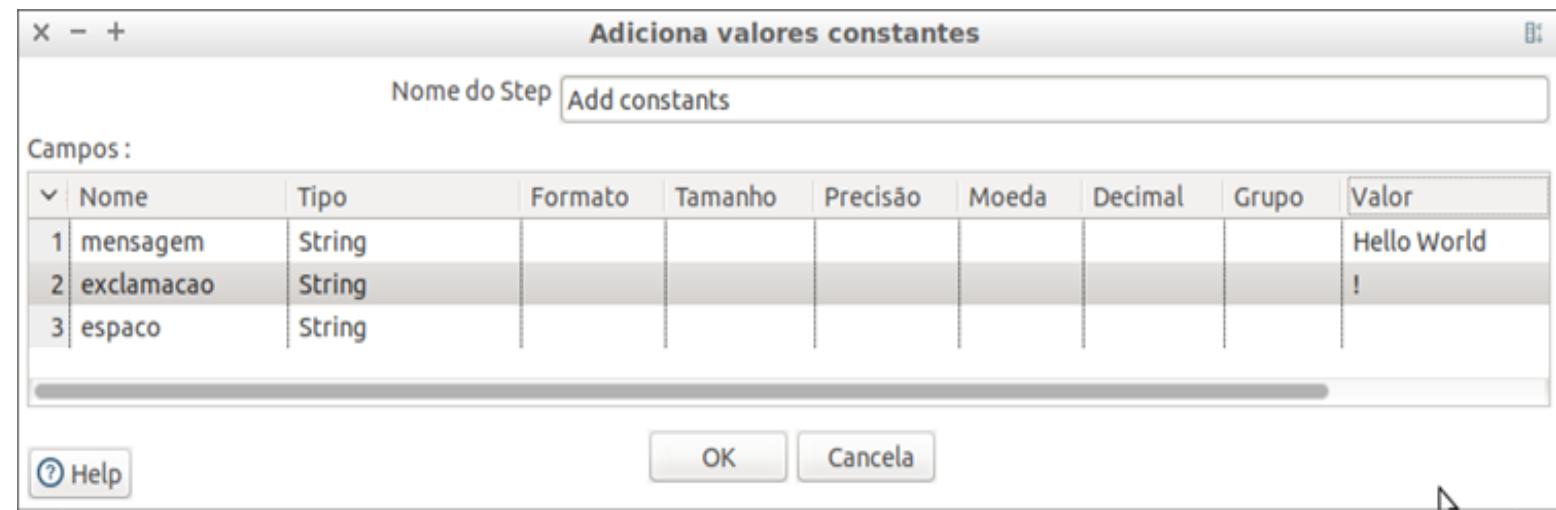
Exercício 2 – Criando a transformação “Hello World Fulano”

4 – Adicione um step *Add constants*, do tipo *Transform*. Edite o step com as seguintes informações nos campos *Nome*, *Tipo* e *valor*:

Nome: mensagem; Tipo: String; Valor=Hello World

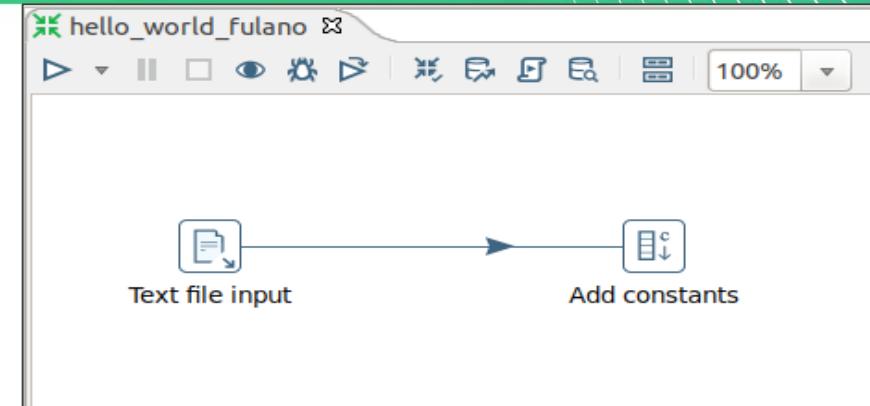
Nome: exclamação; Tipo: String; Valor=!

Nome: espaço; Tipo: String; Valor=“ ” (espaço em branco)



Exercício 2 – Criando a transformação “Hello World Fulano”

5 – Crie um hop ligando os dois steps.



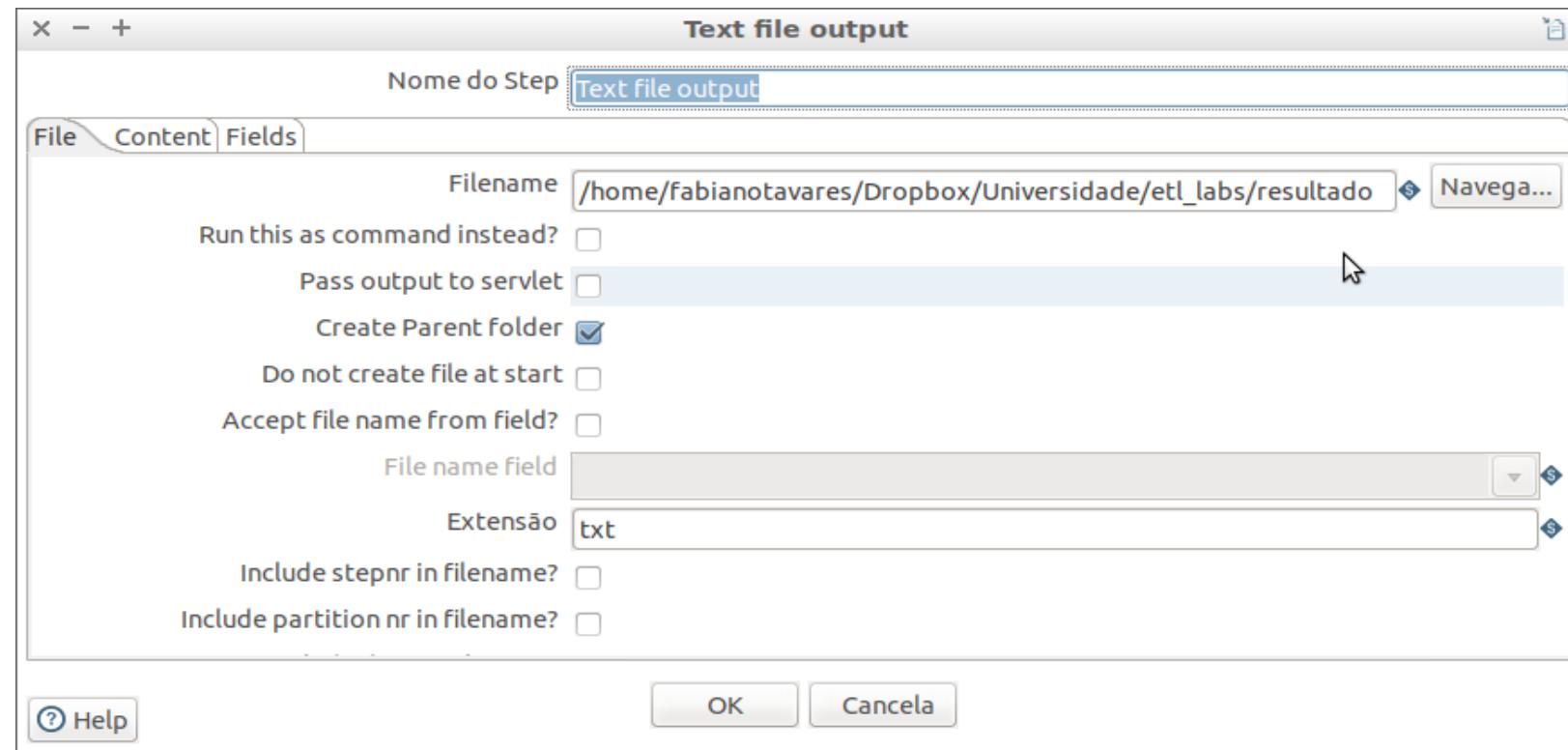
6 – Ainda nos steps do tipo *Transform*, adicione um *Add Sequence*. Em seguida, crie um hop ligando o *Add constants* ao *Add sequence*. Edite esse step para ver os seus parâmetros (não vamos alterá-los para esse exercício).

7 – Entre nas opções de steps do tipo *Output* e adicione um step *Text file output*. Em seguida, crie um hop ligando o step *Add sequence* a *Text file output*. Edite as configurações do step:

Exercício 2 – Criando a transformação “Hello World Fulano”

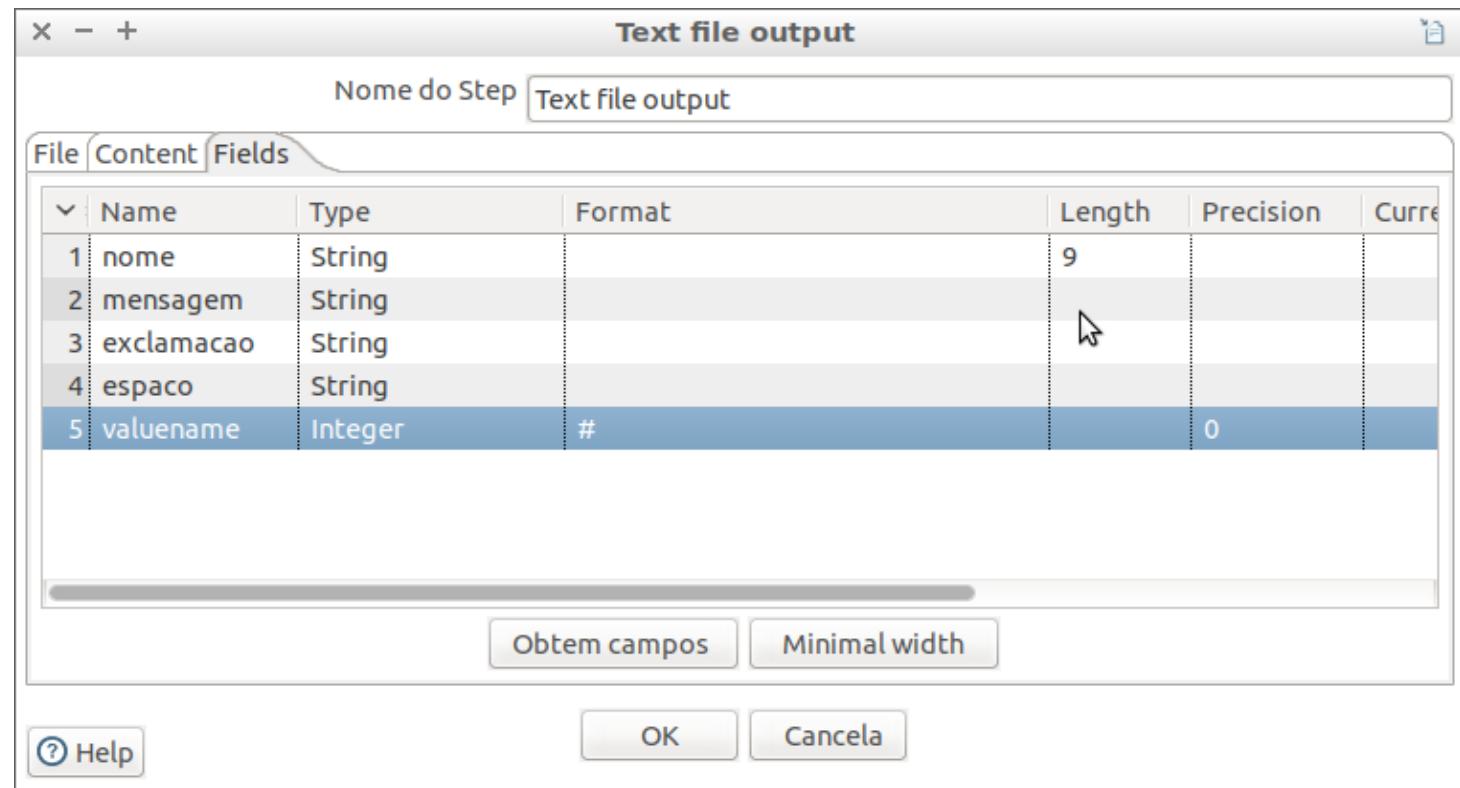
Na aba *File*, digite o caminho de um arquivo de texto com o nome *resultado* (o step adiciona por default a extensão .txt) no campo *Filename*.

Na aba *Content*, limpe o conteúdo do campo *Separator* (caso contrário, o step colocará um separador na saída do arquivo) e desmarque a opção *Header*.



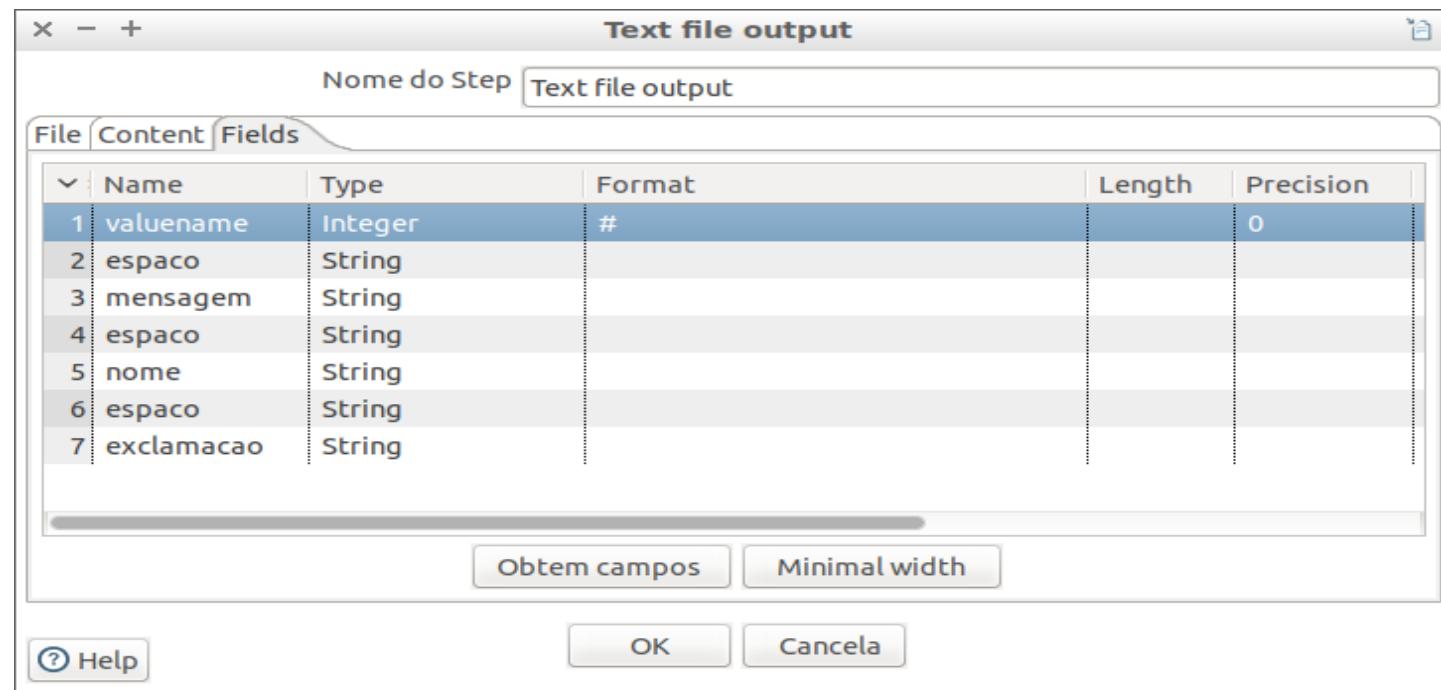
Exercício 2 – Criando a transformação “Hello World Fulano”

Na aba Fields, clique no botão *Obtém campos*. Se tudo foi feito corretamente até agora, serão mostrados os campos de acordo com a figura abaixo. Observe que os campos foram gerados pelos steps anteriores.



Exercício 2 – Criando a transformação “Hello World Fulano”

Queremos que seja gravada no arquivo a mensagem “<valuename> <mensagem><espaço> <nome><exclamacao>”. Para que isso ocorra devemos modificar a ordem dos campos. Clique com o botão direito em cima do campo *valuename* e escolha a opção *move up*. Repita a operação até que o campo *valuename* seja o primeiro da lista. Repita a operação com os demais campos. A figura abaixo mostra a disposição final dos campo



Exercício 2 – Criando a transformação “Hello World Fulano”

8 – Salve a transformação e execute em seguida. O resultado gerado é um arquivo texto com as seguintes linhas:

1 Hello World Fulano!

2 Hello World Sicrano!

3 Hello World Beltrano!



Exercício 3

Criando uma conexão com um banco de dados.

Exercício 3 – Criando uma conexão com um banco de dados.

- 1- Nos exercícios anteriores mostramos como obter dados a partir de uma fonte de dados baseada em arquivos texto. Para os exercícios futuros precisaremos extrair, transformar e carregar dados em tabelas de um banco. Para facilitar o processo de aprendizagem, vamos utilizar o SGBD MySQL. Faça o download da última versão seguindo o tutorial do arquivo “Instalação MySQL”
- 2- Não esqueça de colocar o Driver do MySQL dentro da instalação do PDI. Copie o arquivo mysql-connector-java-5.1 para a pasta /lib/ de sua instalação do PDI.
- 3- Abra o Spoon e carregue a transformação do exercício anterior. Em seguida, clique no botão View na barra lateral. Expanda a aba Conexões.

Exercício 3 – Criando uma conexão com um banco de dados.

4- Com o botão direito, clique em cima da opção *Conexões* e escolha *Novo*.

5- Na aba General, digite os seguintes parâmetros:

Connection name: pdi_mysql_conn

Connection type: MySQL

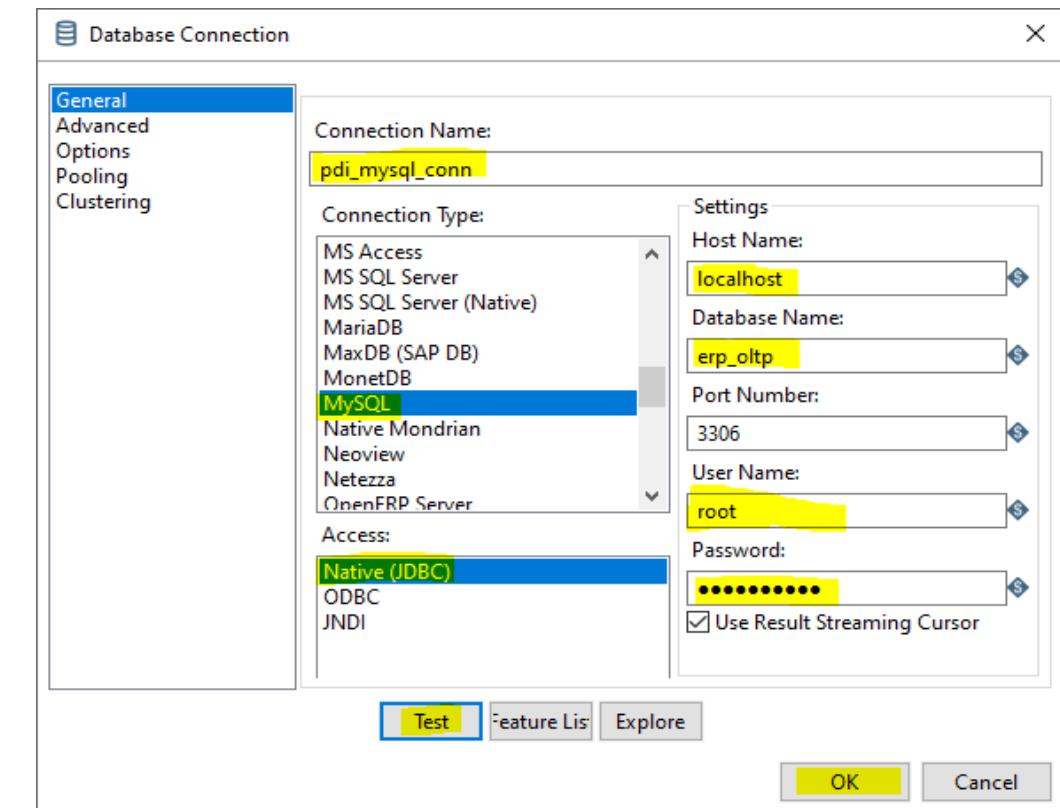
Access: Native (JDBC)

Host: localhost

Port:3306

User: root

Password: 123456

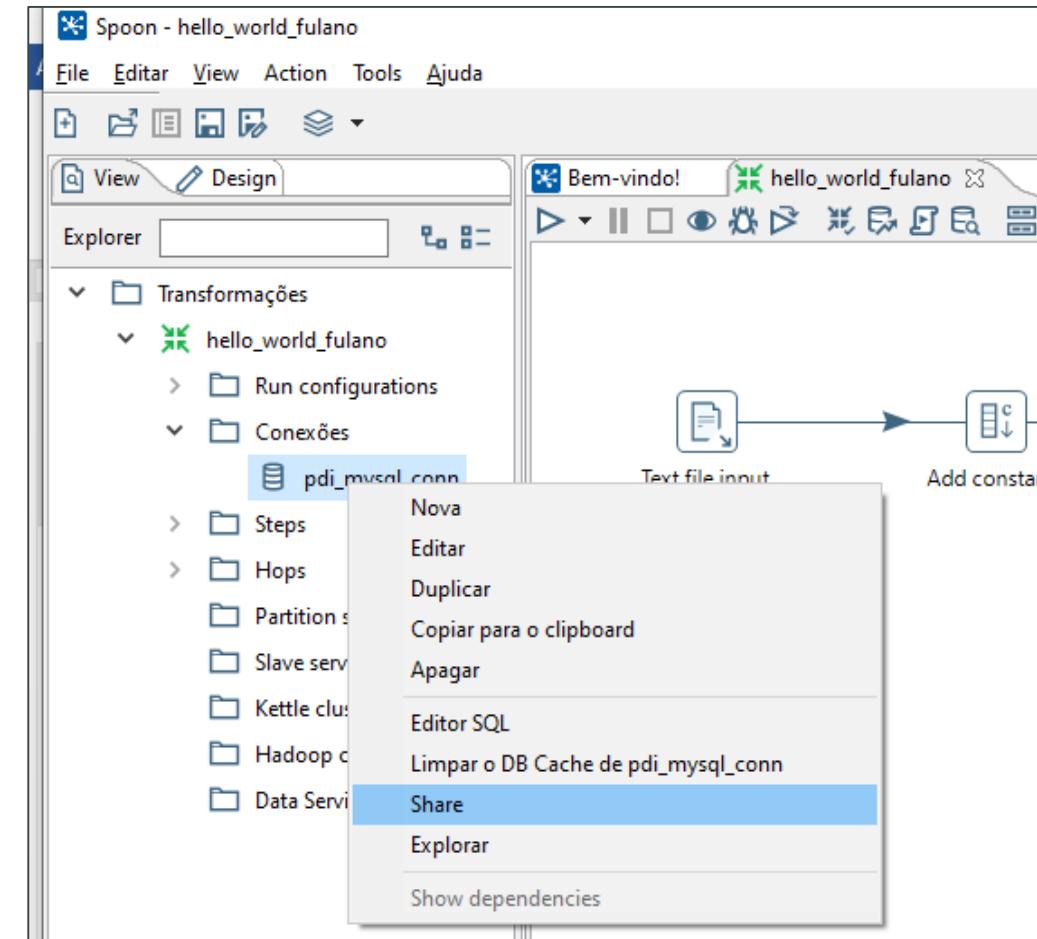


Exercício 3 – Criando uma conexão com um banco de dados.

– Clique no botão Test para verificar se a conexão com o banco está OK. Clique em OK.

6 - Podemos tornar a conexão visível para todas as transformações e Jobs de nossa instalação do PDI. Para isso, clique com o botão direito em cima da conexão criada e escolha a opção *Share*. Note que o texto da conexão está em negrito agora.

7- Mais adiante vamos utilizar essa e outras conexões para a modelagem dimensional e a execução do processo de ETL.



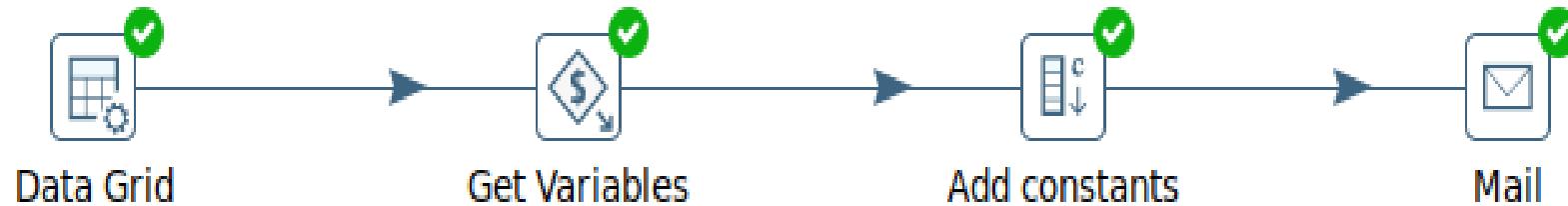
Exercício 4

Criando uma transformação de envio de e-mail.

Exercício 4 – Criando uma transformação de envio de e-mail.

Para este exercício vamos criar uma Transformação de envio de e-mail com passagem de parâmetros entre eles.

1 - Criar uma transformação chamada *call_trans.ktr*. Essa transformação irá enviar um e-mail. Em seguida adicione os seguintes *steps* conforme figura abaixo:



Exercício 4 – Criando uma transformação de envio de e-mail.

Data Grid: configurar mensagens que serão enviadas. Na aba meta criar os campos conforme figurar abaixo. Na aba Data adicionar dados de teste.

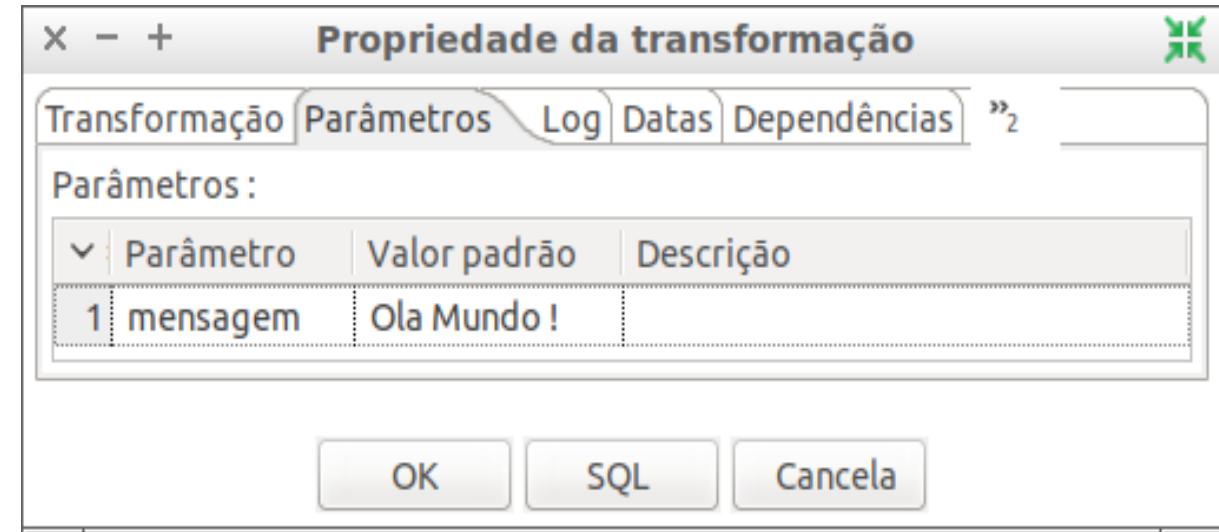
The screenshot shows a software interface for configuring a data grid. At the top, there are buttons for 'x', '−', '+', and 'Add constant rows'. Below this, the 'Nome do Step' (Step Name) is set to 'Data Grid'. The interface has two tabs: 'Meta' and 'Data', with 'Data' currently selected. A table is displayed under the 'Data' tab, showing four columns: 'Name', 'Type', and 'Format' (which is empty). The rows are numbered 1 to 4, and the 'Name' column contains 'nome', 'titulo', 'para', and 'de' respectively, all of which are of type 'String'.

	Name	Type	Format
1	nome	String	
2	titulo	String	
3	para	String	
4	de	String	

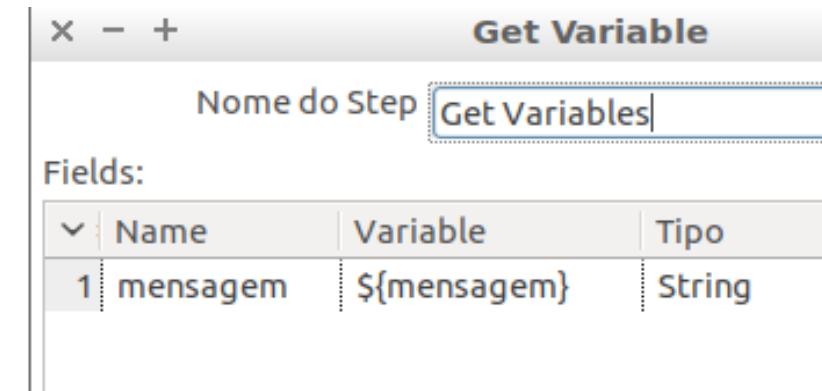
Exercício 4 – Criando uma transformação de envio de e-mail.

Get Variables: capturar as variáveis. Primeiro criar o parâmetro para a transformação:

- clique duplo em uma área em branco da sua transformação.
- Na aba Parâmetros adicionar “mensagem” como parâmetro e um valor padrão:



- Em seguida adicionar o step *Get variables*



Exercício 4 – Criando uma transformação de envio de e-mail.

Add constants: adicionar a configuração SMTP de envio de e-mail.

Configuração SMTP:

Host: smtp.mailtrap.io

Port:25 or 465 or 587 or 2525

Username:389e79a81b7a2b

Password:be34fd4b0b7661

Adiciona valores constantes

Nome do Step

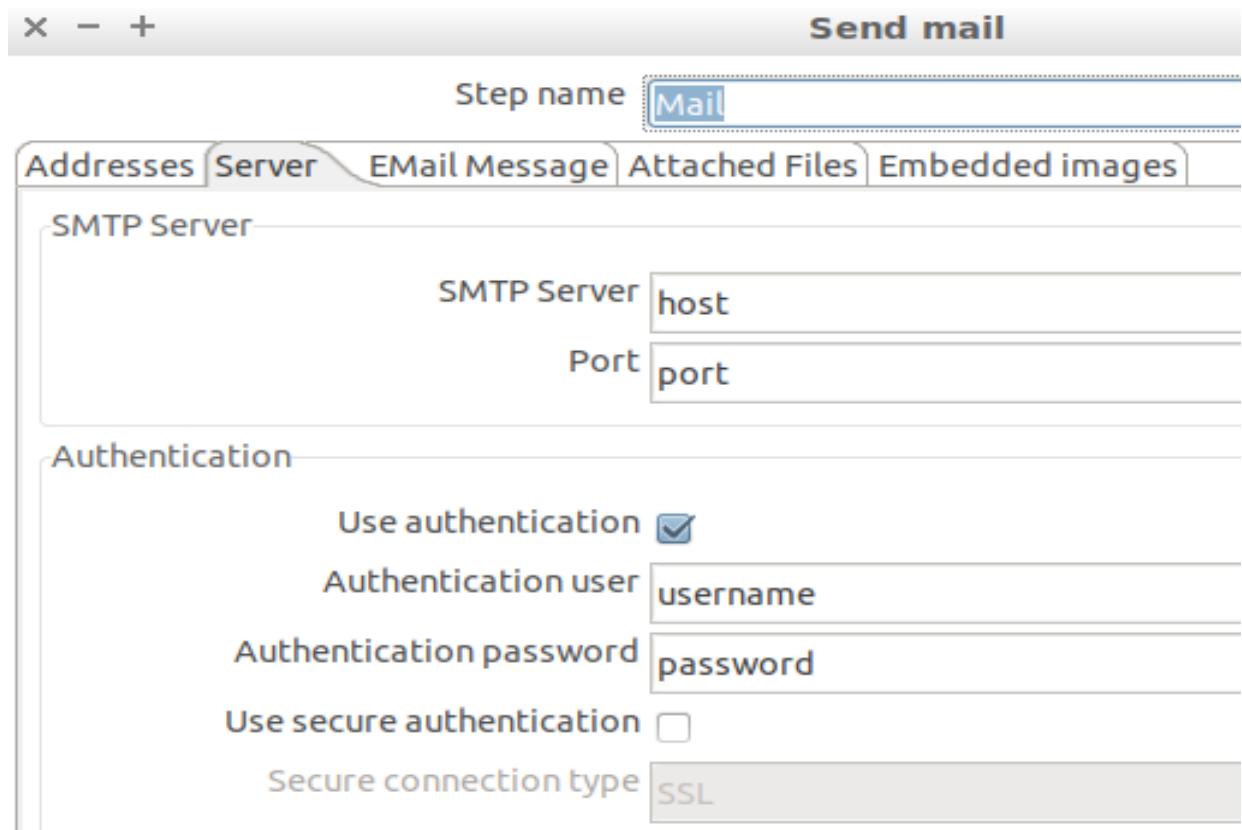
Campos :

#	Nome	Tipo	Formato	Precisão	Decimal	Valor
1	host	String				smtp.mailtrap.io
4	password	String				be34fd4b0b7661
3	username	String				389e79a81b7a2b
2	port	Number				2525

Exercício 4 – Criando uma transformação de envio de e-mail.

Mail: configurar e enviar o e-mail para cada mensagem. Na aba *Addresses* configurar “De e Para” do e-mail. Na aba *Server* configurar o SMTP Server. E finalmente, na Aba *Email Message* o conteúdo da mensagem

Então finalmente executar a transformação !



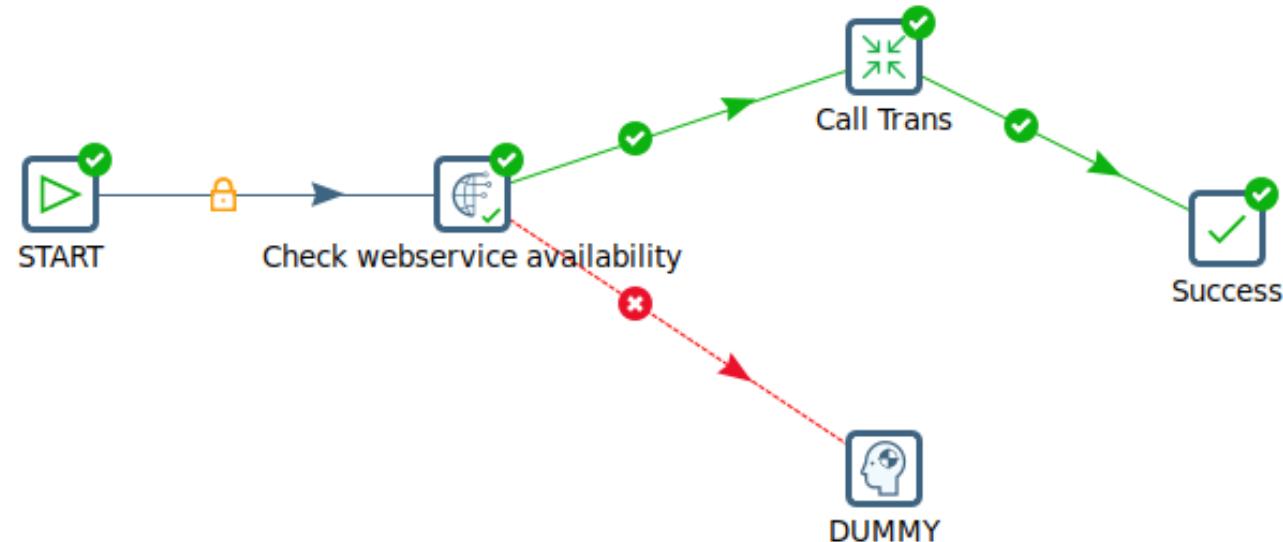
Exercício 5

Criando um Job.

Exercício 5 – Criando um Job.

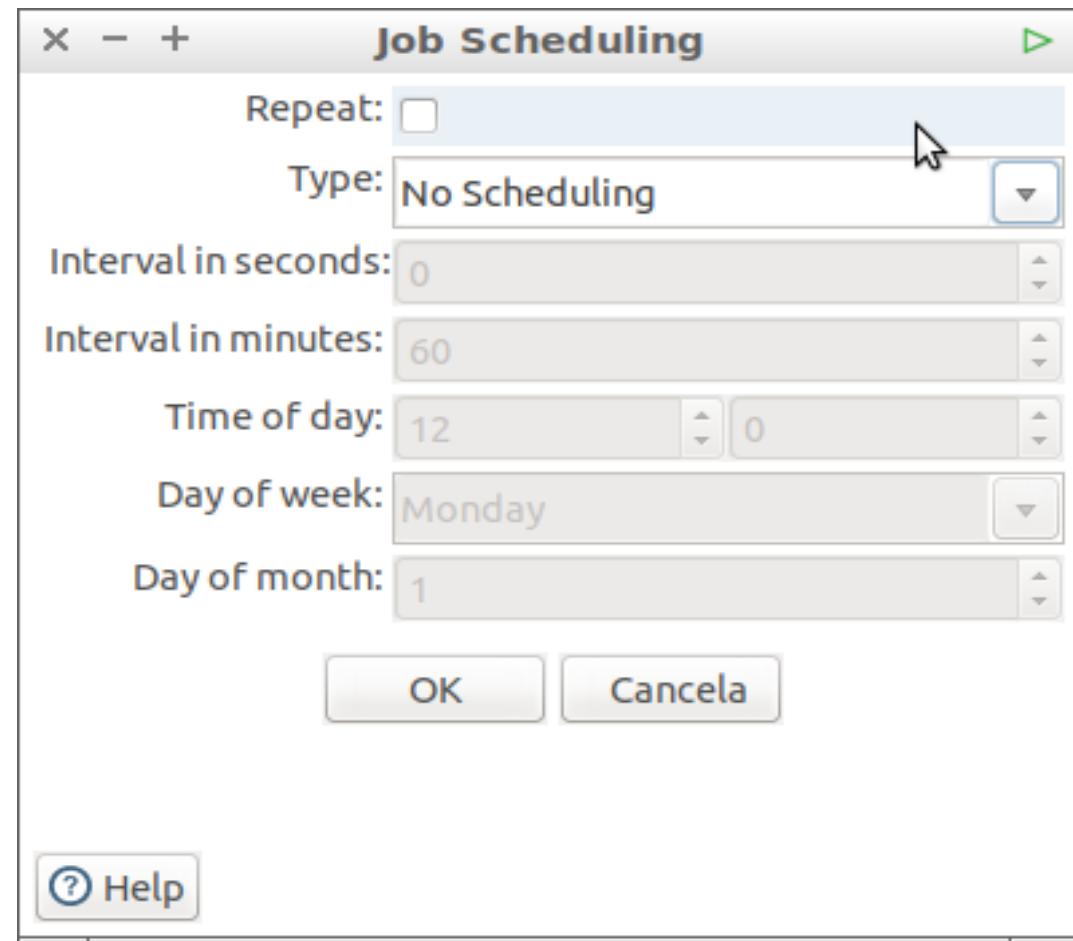
Neste exercício será criado um Job chamando uma transformação com passagem de parâmetros!

1 - Criar uma Job chamado *call_job.ktr*. Esse Job chamará a transformação de enviar e-mail do exercício anterior. Adicionar os seguintes steps:



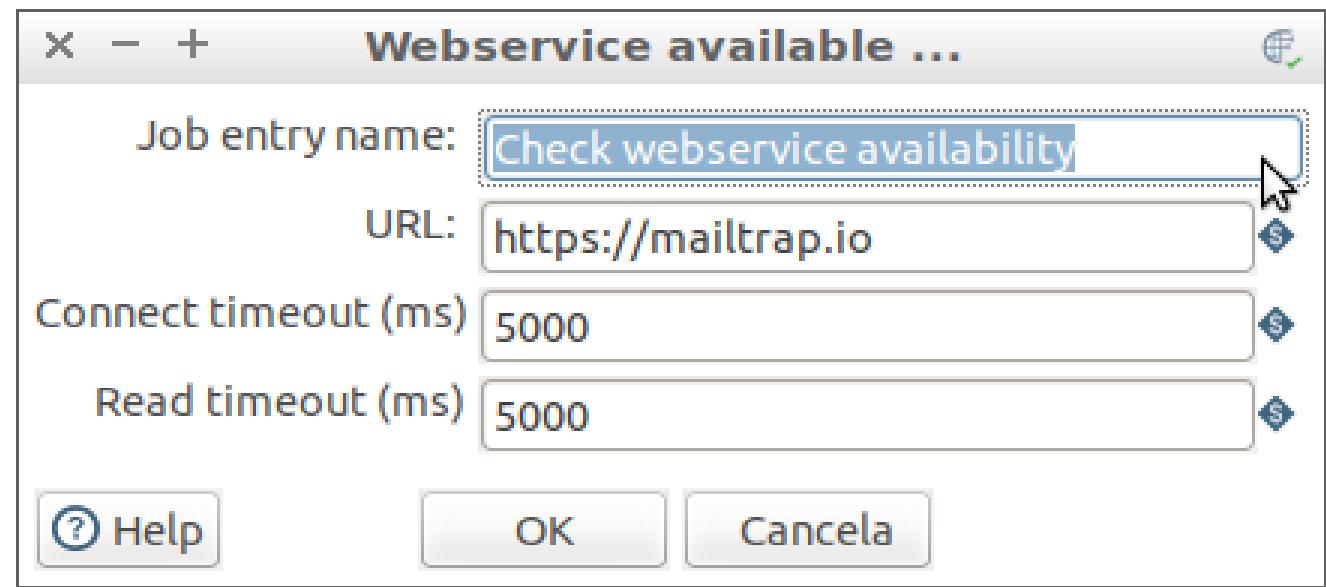
Exercício 5 – Criando um Job.

START: define o ponto de partida para a execução do Job. Todo Job deve ter um (e apenas um). Configura propriedades de repetição e agendamento.



Exercício 5 – Criando um Job.

Check Webservice Availability: verifica a disponibilidade de um serviço web. Possível estabelecer a condicional caso o serviço esteja disponível. Adicionar configuração conforme figura abaixo:

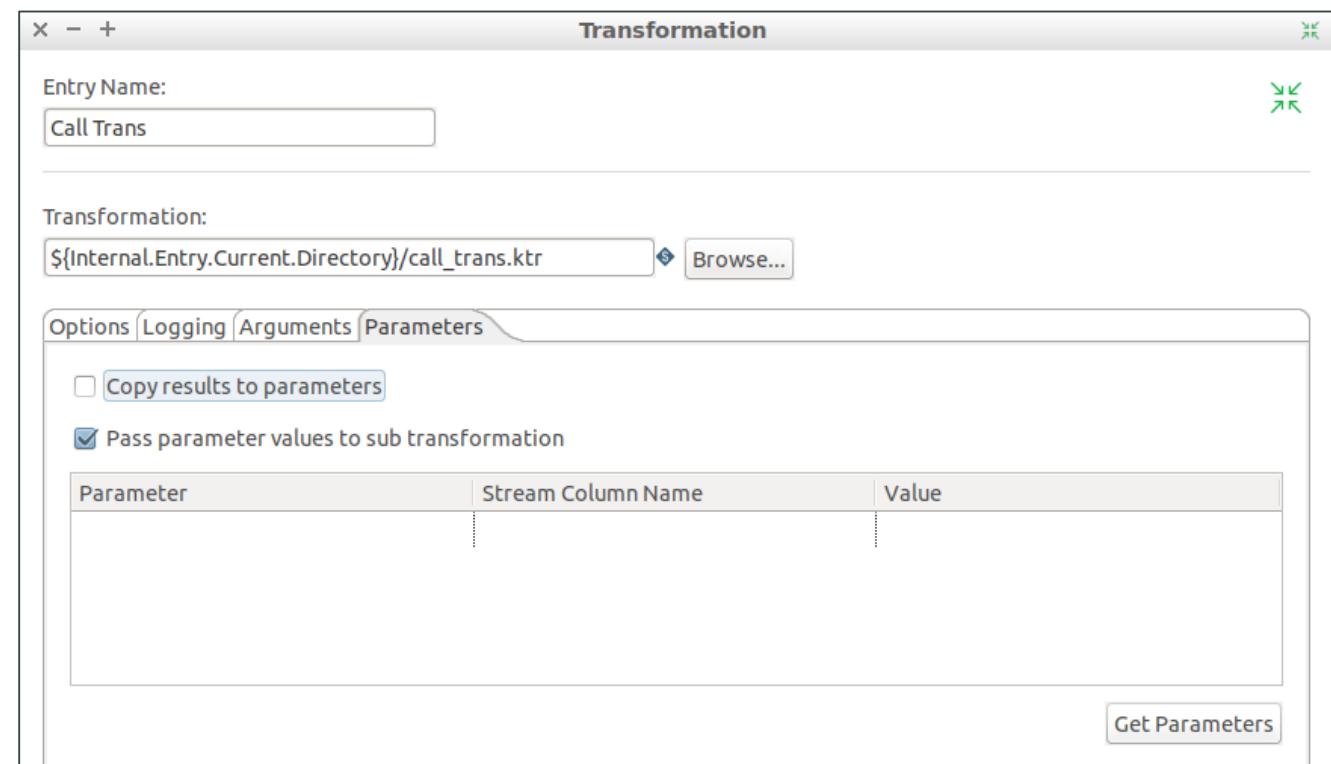


Exercício 5 – Criando um Job.

Transformation: Chama a execução de uma transformação. Um dos steps mais utilizados em Jobs. Pode ser chamado a cada tupla de dados ou depois da execução de todo um *rowsets*. Também é possível de condição em caso de erro.

Vamos configurar a passagens de parâmetros na transformação:

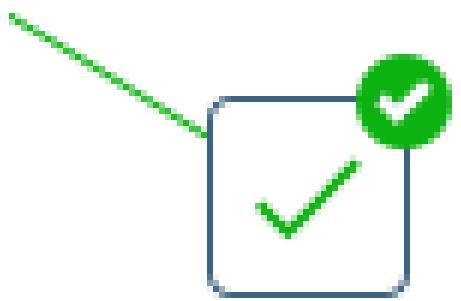
1. Primeiro adicione o parâmetro ao Job:
2. Segundo nas propriedades do step de transformação na aba *Parameters* marcar a opção “Pass parameter values to sub transformation”



Exercício 5 – Criando um Job.

Success: Step de sucesso da execução para indicar o final. Podem existir mais de um ponto de Success.

Não esqueça de executar!



Exercício 6

Executando um Job e Transformação através da linha de comando.

Exercício 6 - Executando um Job e Transformação através da linha de comando.

O objetivo do exercício é automatizar o processo ETL em batch (lote). Possibilitando formas para serem agendados ou em espera. As ferramentas de linha de comando PDI executam o conteúdo PDI de fora do cliente PDI (Spoon). Normalmente, você usaria essas ferramentas no contexto da criação de um script ou de uma tarefa *cron* para executar a tarefa ou transformação com base em alguma condição fora do domínio do software Pentaho.

- ❖ **Pan** é a ferramenta de linha de comando PDI para executar transformações.
- ❖ **Kitchen** é a ferramenta de linha de comando PDI para a execução de tarefas.

Exercício 6 - Executando um Job e Transformação através da linha de comando.

Os sistemas Windows usam sintaxe com a barra ("/") e dois pontos (""). Se houver espaços nos valores da opção, use aspas simples ("") e aspas duplas ("") para manter os espaços juntos, por exemplo, " - param:MASTER_HOST=192.168.1.3" "-param:MASTER_PORT=8181 "

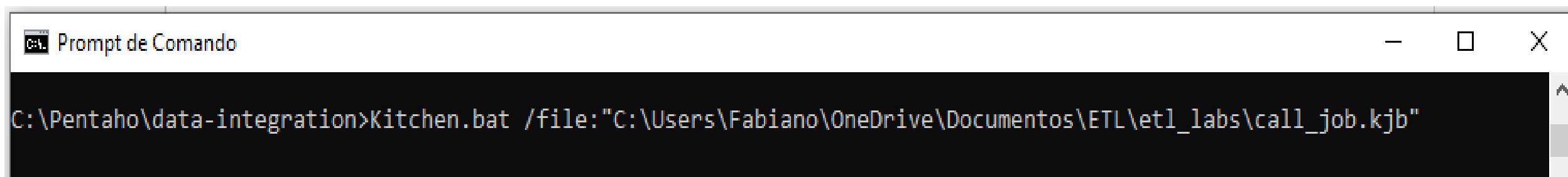
A sintaxe padrão é:

1. Windows: pan.bat /option: valor arg1 arg2
2. Linux: sh pan.sh –option=valor arg1 arg2

Exercício 6 - Executando um Job e Transformação através da linha de comando.

Executar o Job/Transformação:

1. Primeiro abrimos o terminal
2. Entraremos na pasta data-integration
3. Chamar o script pan.bat/pan.sh da linha de comando passando o local do arquivo conforme figura abaixo.
Para o arquivo será utilizado /option ou –option



```
C:\Pentaho\data-integration>Kitchen.bat /file:"C:\Users\Fabiano\OneDrive\Documentos\ETL\etl_labs\call_job.kjb"
```

Exercício 6 - Executando um Job e Transformação através da linha de comando.

4 - Como no item anterior criamos o parâmetro mensagem, então iremos adicionar uma mensagem personalizada para envio de e-mail.

C:\ Prompt de Comando

```
C:\Pentaho\data-integration>Kitchen.bat /file:"C:\Users\Fabiano\OneDrive\Documentos\ETL\etl_labs\call_job.kjb" /param:"mensagem=Mensagem de email via PDI ! "
```

5 - Caso esteja utilizando um repositório do PDI será necessário passar nome do repositório, usuário, senha e nome do job. Conforme a seguir:

```
kitchen.bat /rep:meu_pdi_repo /user:admin /pass:passadmin /job:call_job
```

Exercício 6 - Executando um Job e Transformação através da linha de comando.

6 - Caso seja uma transformação será semelhante os comandos para Job, exceto pelo fato de o script de execução ser pan.bat/pan.sh. Conforme a seguir:

 Prompt de Comando

```
C:\Pentaho\data-integration>Pan.bat /file:"C:\Users\Fabiano\OneDrive\Documentos\ETL\etl_labs\call_trans.ktr" /param:"mensagem=Mensagem de email via PDI 2! "
```

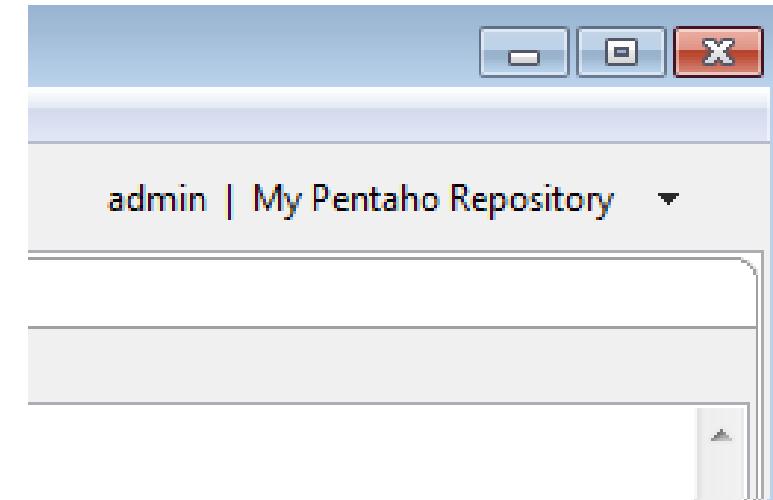
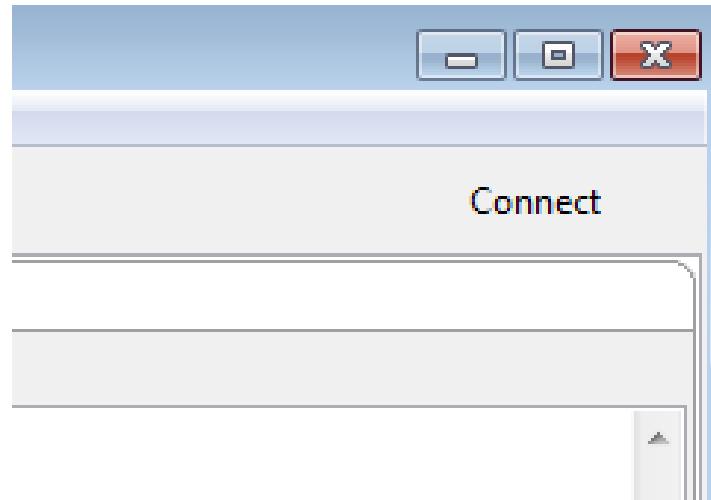
Exercício 7

Criando repositório no PDI

Exercício 7 - Criando repositório no PDI

O PDI Cliente (*Spoon*) oferece vários tipos diferentes de armazenamento de arquivos. Se sua equipe precisar de um ambiente colaborativo ETL (Extrair, Transformar e Carregar), recomendamos o uso de um Repositório Pentaho.

1 - Selecione “*Repository Manager*” no menu Connect.



Exercício 7 - Criando repositório no PDI

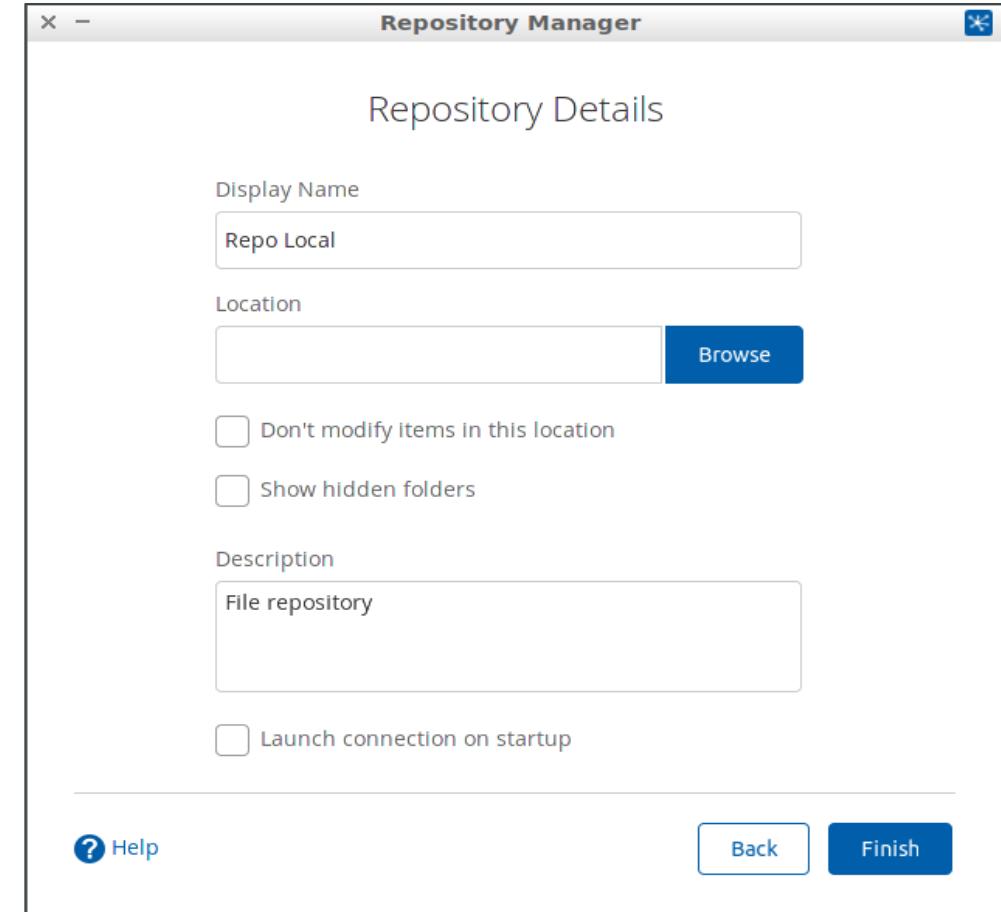
2 - Existem três opções de repositório:

- Pentaho Repository (BI Server)
- Database Repository
- File Repository
- Não usar repositório

3 - Clique “Other Repositories”

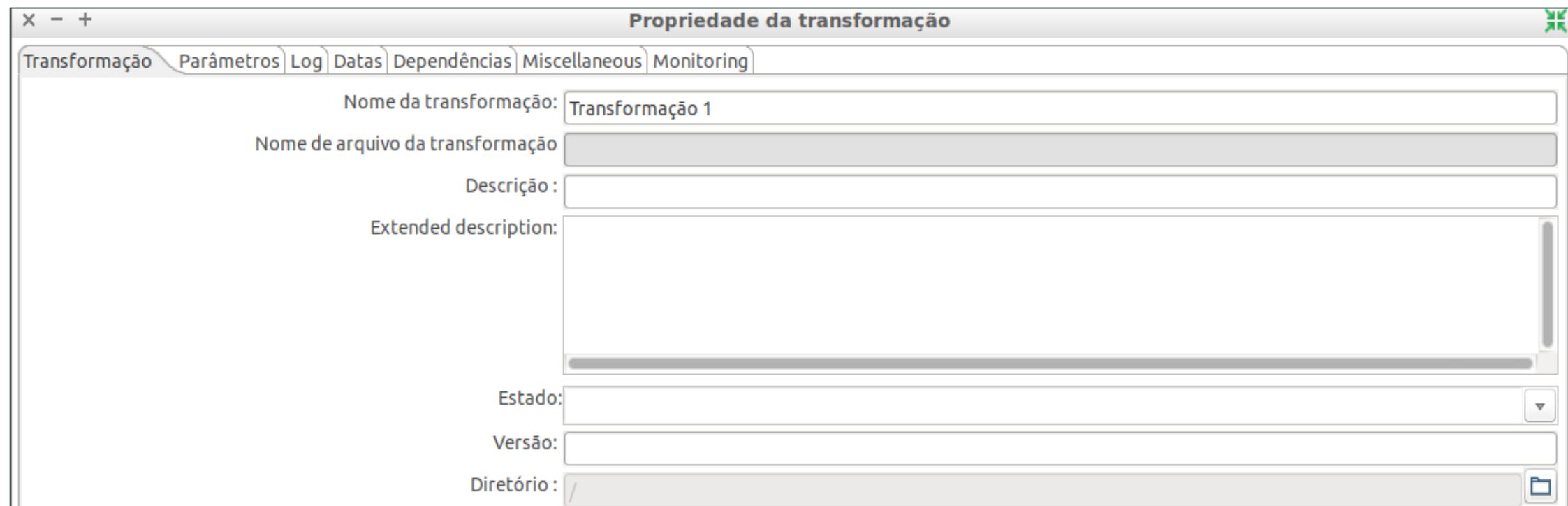
4 - Escolha “File Repository”.

5 - Em “*Location*” escolha a pasta raiz do seu repositório:



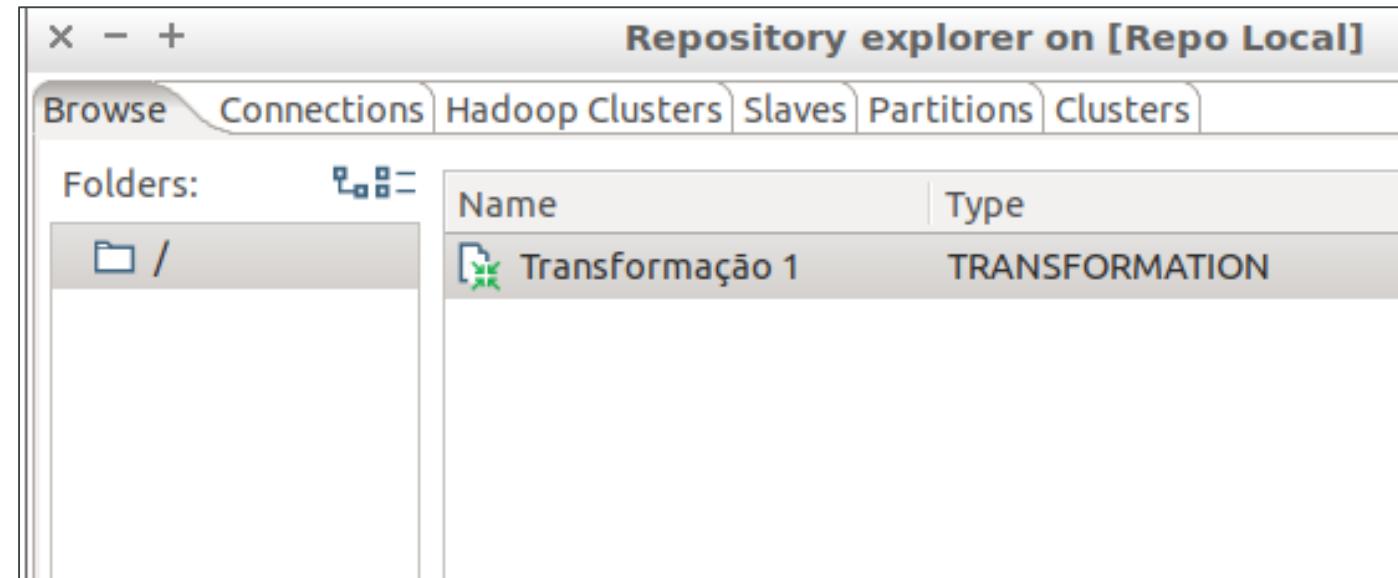
Exercício 7 - Criando repositório no PDI

6 - Agora crie uma nova transformação (CTRL+N) e salve no seu repositório (CTRL+S). Conforme figura abaixo. Escolha um nome para transformação e o diretório relativo que será salvo e então clique em OK.



Exercício 7 - Criando repositório no PDI

7 - Agora abra o “Repository Explorer” (CTRL+E) e confira a transformação.



Importante que a estrutura de diretório seja criada com objetivo de organizar, estruturar e classificar seus mapas ETL. É possível “versionar” seu repositório.

Exercício 8

Obtendo informações do Sistema

Exercício 8 - Obtendo informações do Sistema

Em alguma situações é necessário obter dados do sistema para complementar informações.

Leia os dados do sistema e salve em um arquivo de texto. Escolha uma das informações seguintes:

1. Datas
2. Sistema de arquivos
3. Rede

Step: Get System Info



[Get System Info](#)

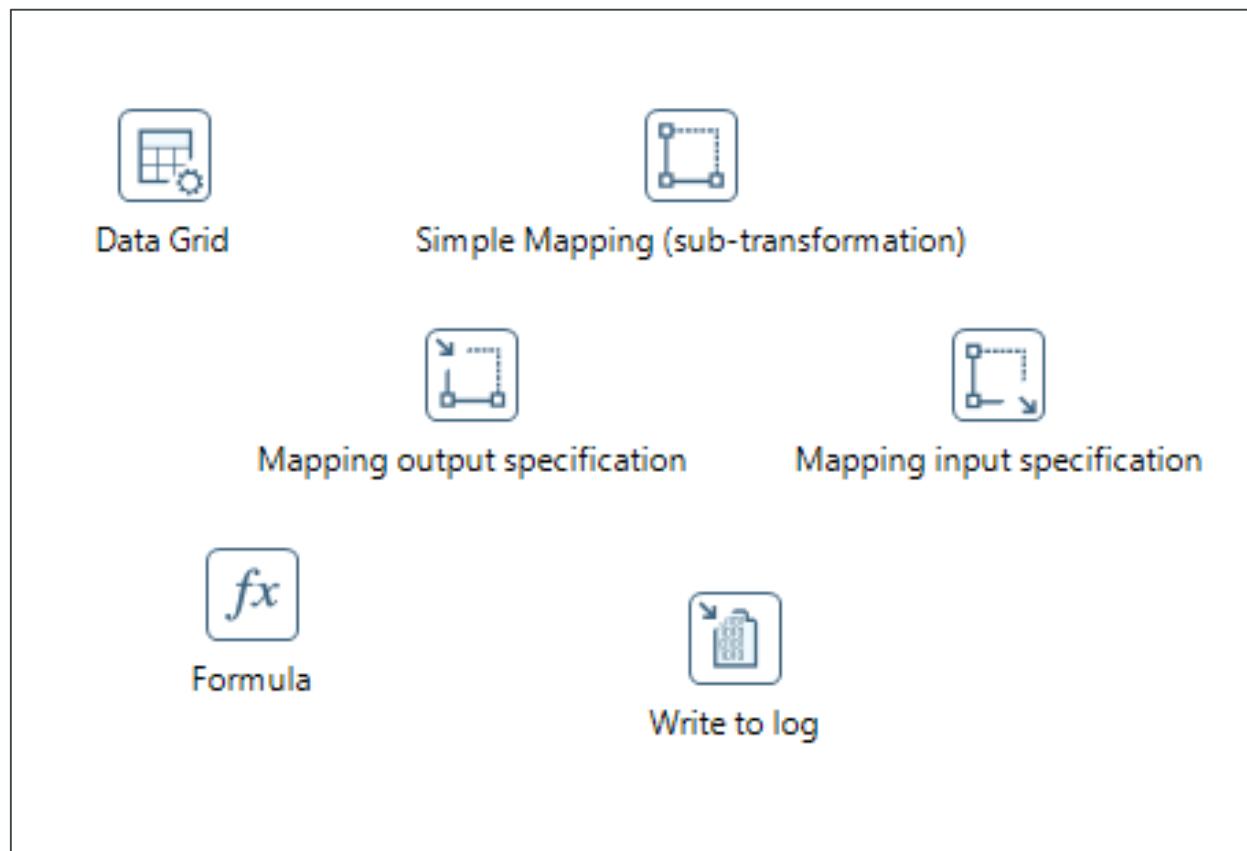
Exercício 9

Crie um módulo com transformações

Exercício 9 - Crie um módulo com transformações

Modularizar é necessário! Crie um módulo de soma passando valores em colunas em A e B.

Use os seguintes Steps:



Exercício 10

Crie uma transformação.

Exercício 10

Escolha alguns *steps* e crie uma transformação. Quantos mais steps melhor para o treinamento. Use a criatividade !



