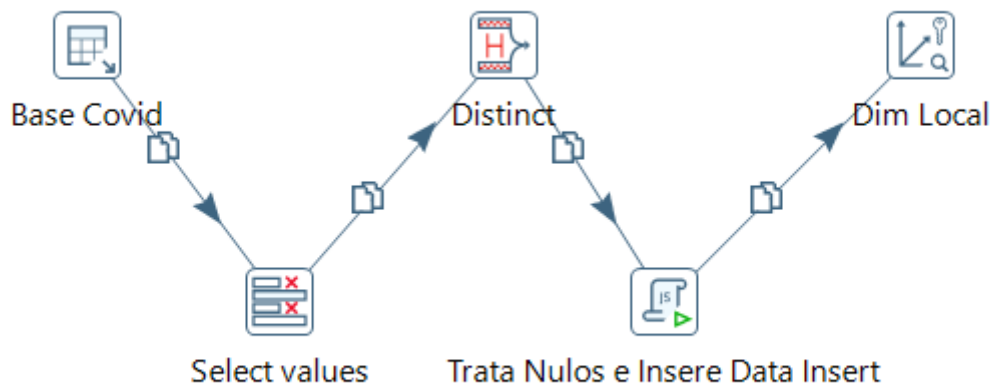


Exercício 01 – Dimensão Local

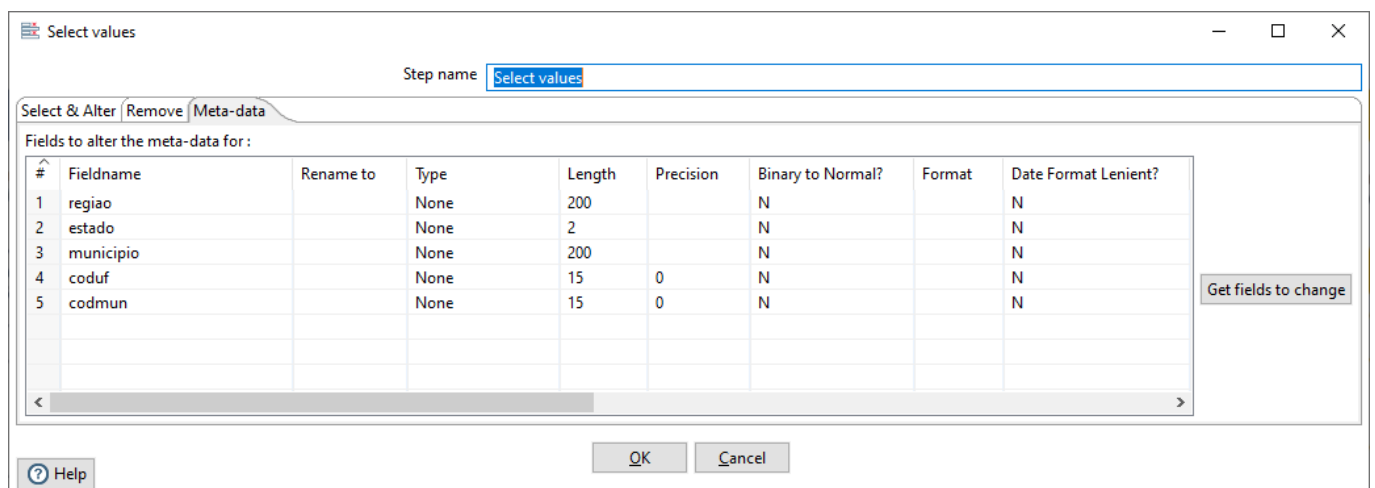
1.1 Crie uma transformação chamada “01 – DIM LOCAL”, e salve em uma nova pasta, “LAB06” em seu repositório. Traga os steps a seguir:

- Table Input
- Select Values
- Unique Rows (HashSet)
- Modified JavaScript value
- Combination Lookup update



Data flow final da transformação

1.2 Em “Select values”, selecione apenas os campos “região”, “estado”, “município”, “coduf”, “codmun”, alterando os tipos de dados conforme orientado na imagem a seguir:



#	Fieldname	Rename to	Type	Length	Precision	Binary to Normal?	Format	Date Format Lenient?
1	regiao		None	200		N		N
2	estado		None	2		N		N
3	municipio		None	200		N		N
4	coduf		None	15	0	N		N
5	codmun		None	15	0	N		N

1.3 Elimine as repetições:

Unique rows (HashSet)

Step name:

Settings

Compare using stored row values? ☐

Redirect duplicate row ☐ Error description

Fields to compare (no entries means: compare complete row)
Note: values for the selected fields are stored in memory for every record.

#	Fieldname
1	codmun
2	coduf
3	regiao
4	estado
5	municipio

Help OK Cancel Get

1.4 Utilizando um step Javascript (ou outro à sua escolha), realize o tratamento de nulos:

Modified JavaScript value

Step name:

Java script functions:

- Transform Scripts
- Transform Constants
- Transform Functions
- Input fields
 - codmun
 - coduf
 - regiao
 - estado
 - municipio
- Output fields
 - Please use the 'Replace value' field

Java script:

```
//Script here
var v_dat_insert= new Date();

if (estado == null)
{
  estado = substr(regiao,0,2);
}

if (codmun == null)
{
  codmun = coduf;
}

if (municipio == null)
{
  municipio = estado;
}
```

Linenn: 0

Compatibility mode? ☐ Optimization level

#	Fieldname	Rename to	Type	Length	Precision	Replace value 'Fieldname' or 'Rename to'
1	v_dat_insert		Date			N
2	codmun		Integer			Y
3	estado		String			Y
4	municipio		String			Y

Help OK Cancel Get variables Test script

Observações:

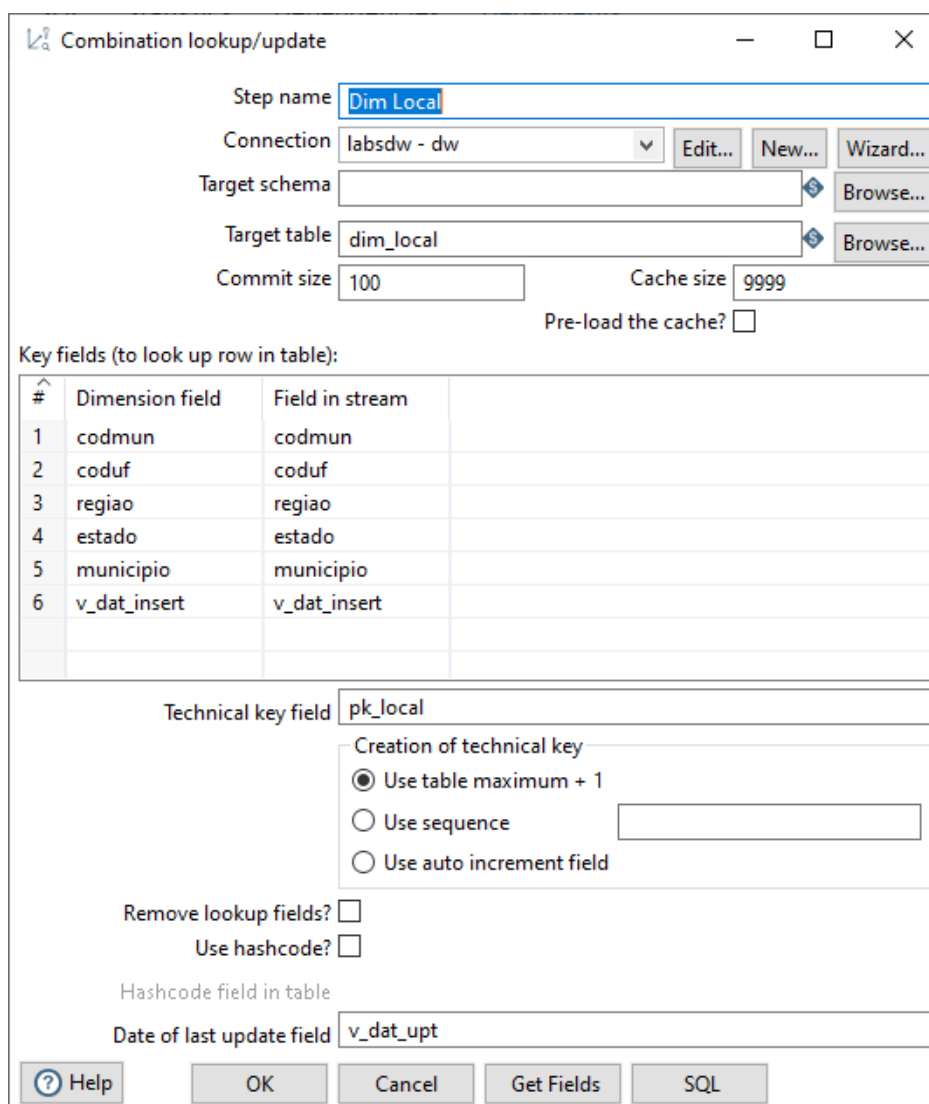
Documentação sobre a Transformação "Select Values"

<https://help.pentaho.com/Documentation/9.0/Products/Select Values>

Tutorial sobre funções JavaScript

<https://www.tutorialsteacher.com/javascript/javascript-tutorials>

- 1.5** No step “Combination lookup/update”, configure a chave técnica como “pk_local”, adicione ao campo “Date of last update field” o valor “v_dat_updt”, crie a tabela utilizando o botão “SQL” e execute a transformação:



Combination lookup/update

Step name:

Connection:

Target schema:

Target table:

Commit size: Cache size:

Pre-load the cache? ☐

Key fields (to look up row in table):

#	Dimension field	Field in stream
1	codmun	codmun
2	coduf	coduf
3	regiao	regiao
4	estado	estado
5	municipio	municipio
6	v_dat_insert	v_dat_insert

Technical key field:

Creation of technical key

☒ Use table maximum + 1

☐ Use sequence

☐ Use auto increment field

Remove lookup fields? ☐

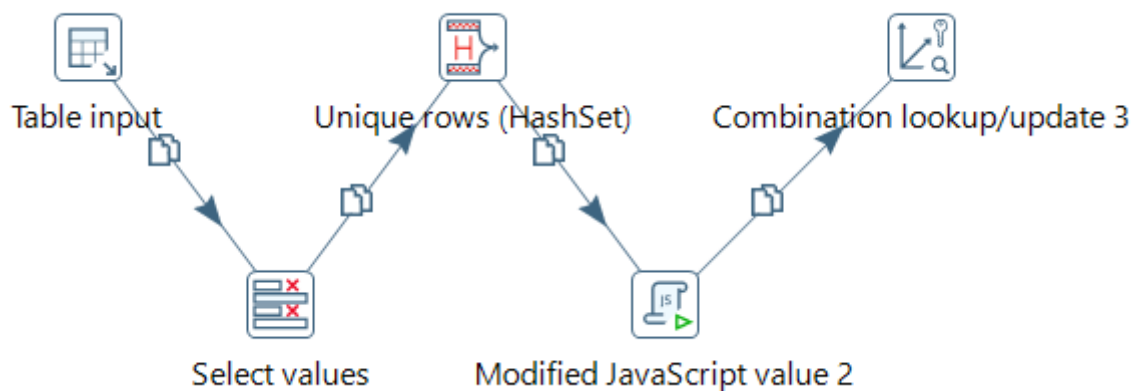
Use hashcode? ☐

Hashcode field in table

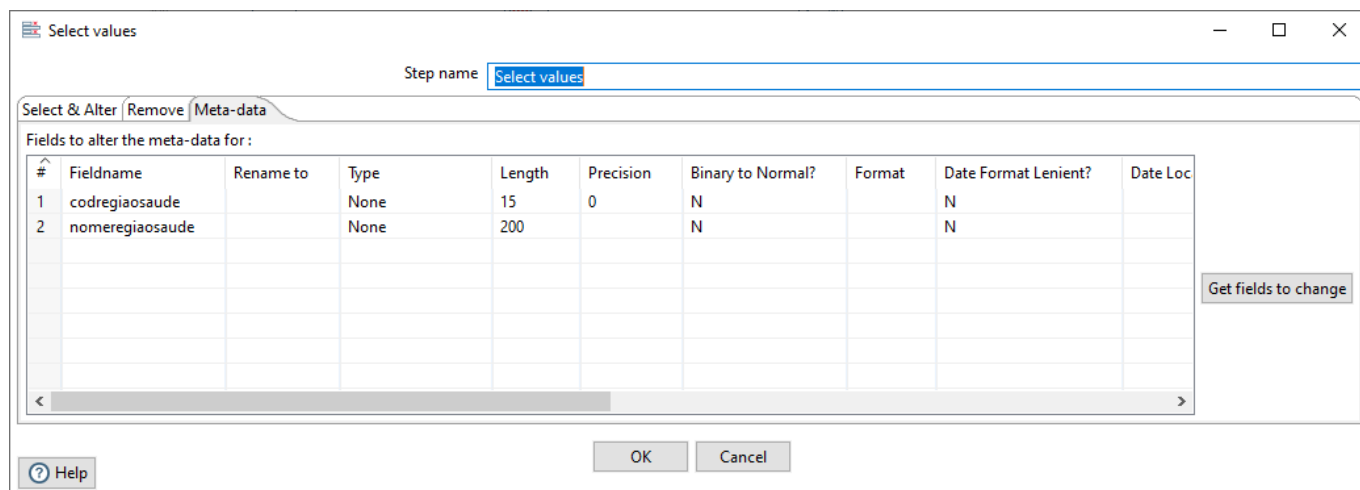
Date of last update field:

Exercício 02 – Dimensão Reg Saude

2.1 Crie uma transformação chamada “02 – DIM REG SAUDE”, e a salve na pasta LAB06. Traga os steps a seguir:



2.2 Traga apenas as colunas “codigoregiaosaude” e “nomeregiaoasaude”, verifique se suas precisões estão de acordo com a imagem a seguir:

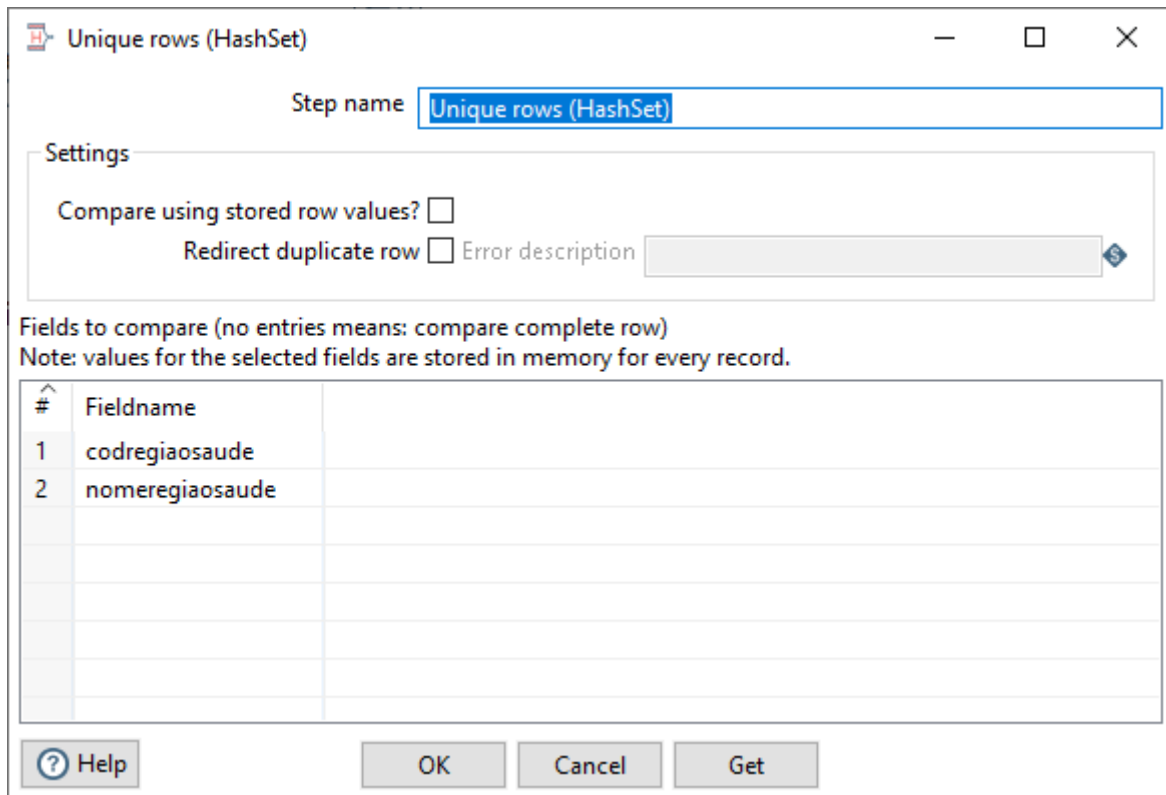


#	Fieldname	Rename to	Type	Length	Precision	Binary to Normal?	Format	Date Format Lenient?	Date Loc
1	codigoregiaosaude		None	15	0	N		N	
2	nomeregiaoasaude		None	200		N		N	

Get fields to change

OK Cancel

2.3 Realize o “Distinct”:



Unique rows (HashSet)

Step name: Unique rows (HashSet)

Settings

Compare using stored row values? ☐

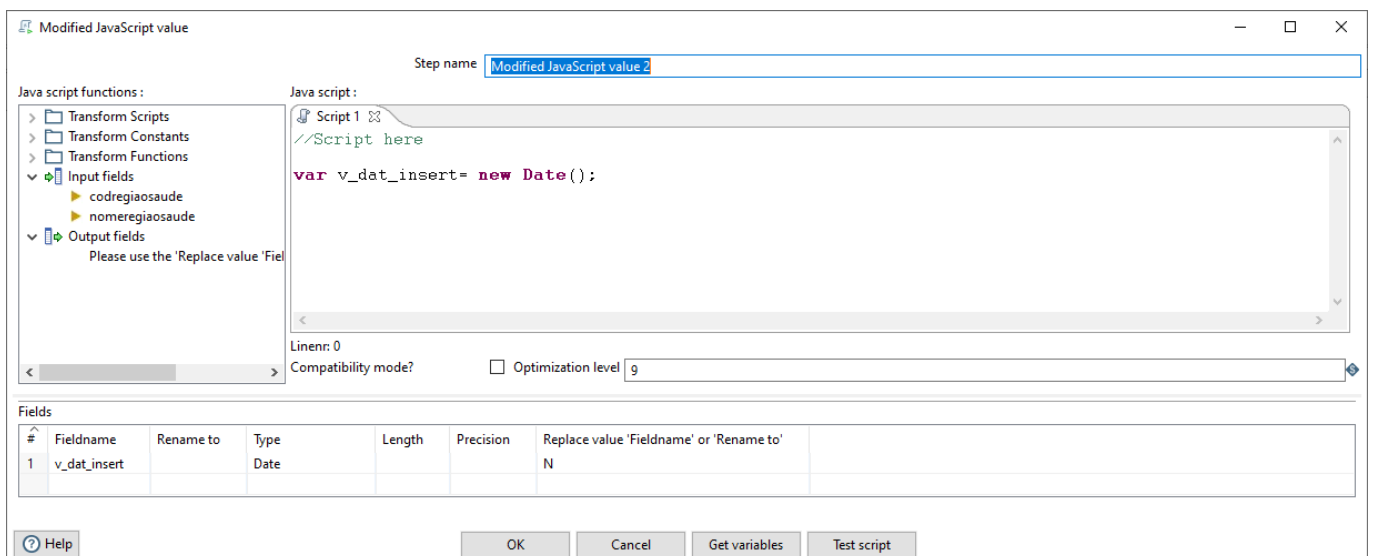
Redirect duplicate row ☐ Error description

Fields to compare (no entries means: compare complete row)
Note: values for the selected fields are stored in memory for every record.

#	Fieldname
1	codregiaoasaude
2	nomeregiaoasaude

Help OK Cancel Get

2.4 Adicione um campo novo para gravar na tabela a data de inserção do registro:



Modified JavaScript value

Step name: Modified JavaScript value 2

Java script functions:

- Transform Scripts
- Transform Constants
- Transform Functions
- Input fields
 - codregiaoasaude
 - nomeregiaoasaude
- Output fields
 - Please use the 'Replace value' field

Java script:

```
//Script here
var v_dat_insert= new Date();
```

Linelnr: 0

Compatibility mode? ☐ Optimization level 9

Fields

#	Fieldname	Rename to	Type	Length	Precision	Replace value 'Fieldname' or 'Rename to'
1	v_dat_insert		Date			N

Help OK Cancel Get variables Test script

2.5 Defina a chave técnica, para compor a pk da tabela, e a data de última atualização, crie a tabela e execute a transformação:

Combination lookup/update

Step name: Combination lookup/update 3

Connection: labsdw - dw [Edit...] [New...] [Wizard...]

Target schema: [Browse...]

Target table: dim_reg_saude [Browse...]

Commit size: 100 Cache size: 9999

Pre-load the cache? ☐

Key fields (to look up row in table):

#	Dimension field	Field in stream
1	codregiaosaude	codregiaosaude
2	nomeregiaosaude	nomeregiaosaude
3	v_dat_insert	v_dat_insert

Technical key field: pk_reg_saude

Creation of technical key

☒ Use table maximum + 1

☐ Use sequence []

☐ Use auto increment field

Remove lookup fields? ☐

Use hashcode? ☐

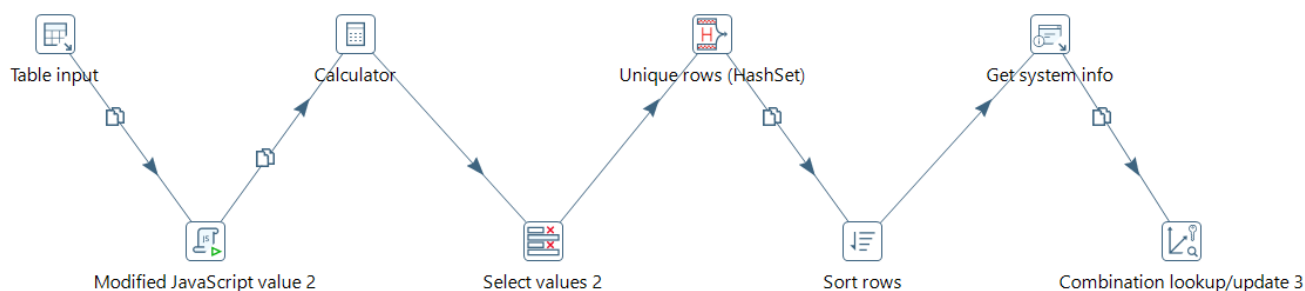
Hashcode field in table

Date of last update field: v_dat_upt

[?] Help [OK] [Cancel] [Get Fields] [SQL]

Exercício 03 – Dimensão Tempo

3.1 Crie uma transformação chamada “03 – DIM TEMPO”, e adicione os steps listado na imagem a seguir:



3.2 Utilize o JavaScript (ou outro step que você queira) para tratar dos nulos:

Modified JavaScript value

Step name: **Modified JavaScript value 2**

JavaScript functions:

- Transform Scripts
- Transform Constants
- Transform Functions
- Input fields
 - regiao
 - estado
 - municipio
 - coduf
 - codmun
 - codregiao
 - nomeregiao
 - data
 - semanaepi
 - populacaotcu2019
 - casosacumulado
 - casosnovos
 - obitosacumulado
 - obitosnovos
 - recuperadosnovos
 - emacompanhamentonovos
 - interior_metropolitana

JavaScript:

```

//Script here

if (estado == null)
{
  estado = substr(regiao,1,2);
}

if (codmun == null)
{
  codmun = coduf;
}

if (municipio == null)
{
  municipio = estado;
}
  
```

Fields

#	Fieldname	Rename to	Type	Length	Precision	Replace value 'Fieldname' or 'Rename to'
1	codmun		Integer			Y
2	estado		String			Y
3	municipio		String			Y

OK Cancel Get variables Test script

Código JavaScript para tratar dos nulos:

```
if (estado == null)
{
    estado = substr(regiao,1,2)
}

if (codmun == null)
{
    codmun = coduf; }

if (municipio == null)
{
    municipio = estado; }
```

3.3 Extraia do campo “data” as partes “ano”, “mês” e “dia”, utilizando o step “Calculator”:

Calculator

Step name

Calculator

☒ Throw an error on non existing files

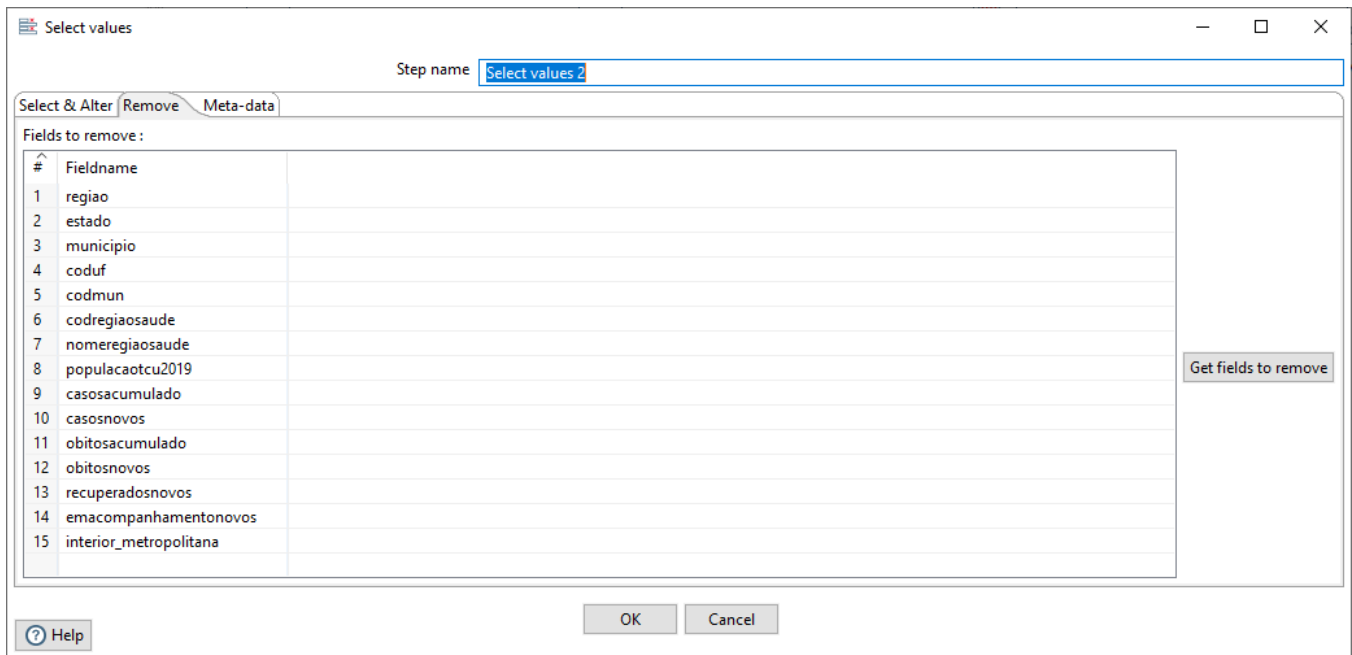
Fields:

#	New field	Calculation	Field A	Field B	Field C	Value type	Length	Precision	Remove	Conversion mask
1	dia	Day of month of date A	data			Integer			N	0
2	mes	Month of date A	data			Integer			N	0
3	ano	Year of date A	data			Integer			N	0

Help

OK Cancel

3.4 Remova os campos que não serão necessários à dimensão tempo:



Step name:

Select & Alter Remove Meta-data

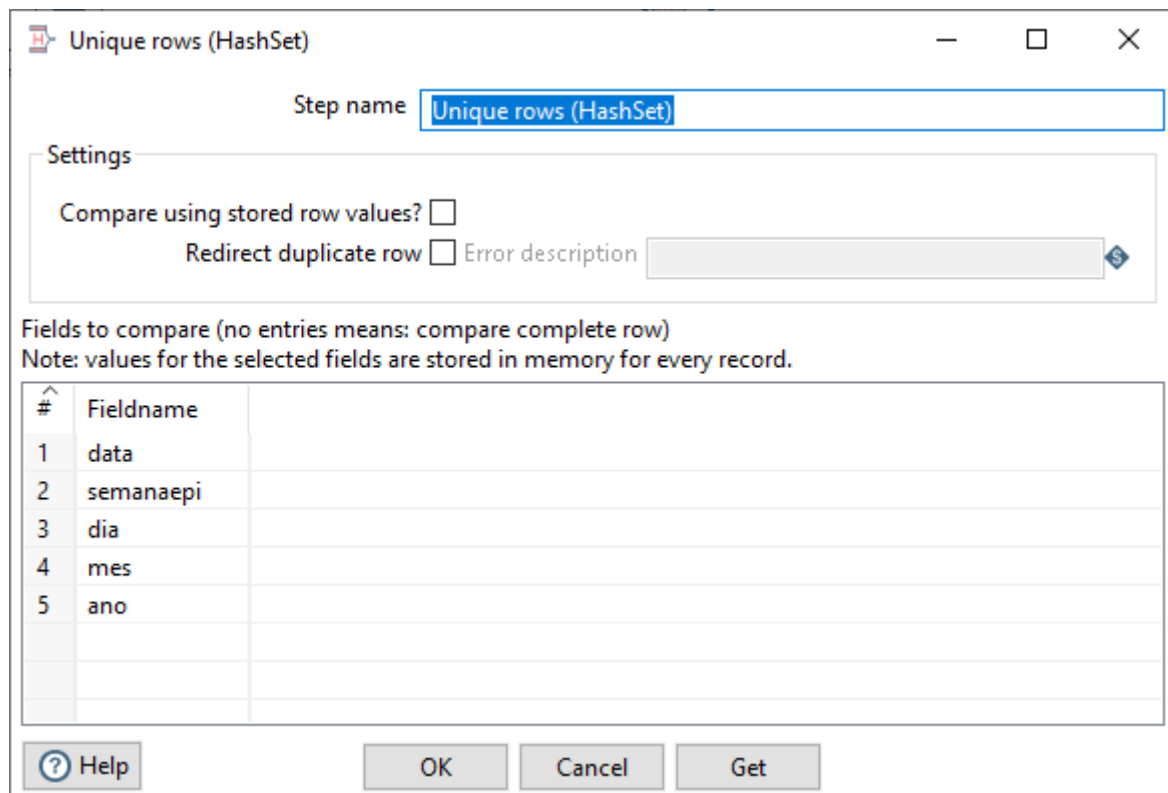
Fields to remove:

#	Fieldname
1	regiao
2	estado
3	municipio
4	coduf
5	codmun
6	codregiaosaude
7	nomeregiaosaude
8	populacaotcu2019
9	casosacumulado
10	casosnovos
11	obitosacumulado
12	obitosnovos
13	recuperadosnovos
14	emacompanhamentonovos
15	interior_metropolitana

Get fields to remove

Help OK Cancel

3.5 Remova as repetições:



Step name:

Settings

Compare using stored row values? ☐

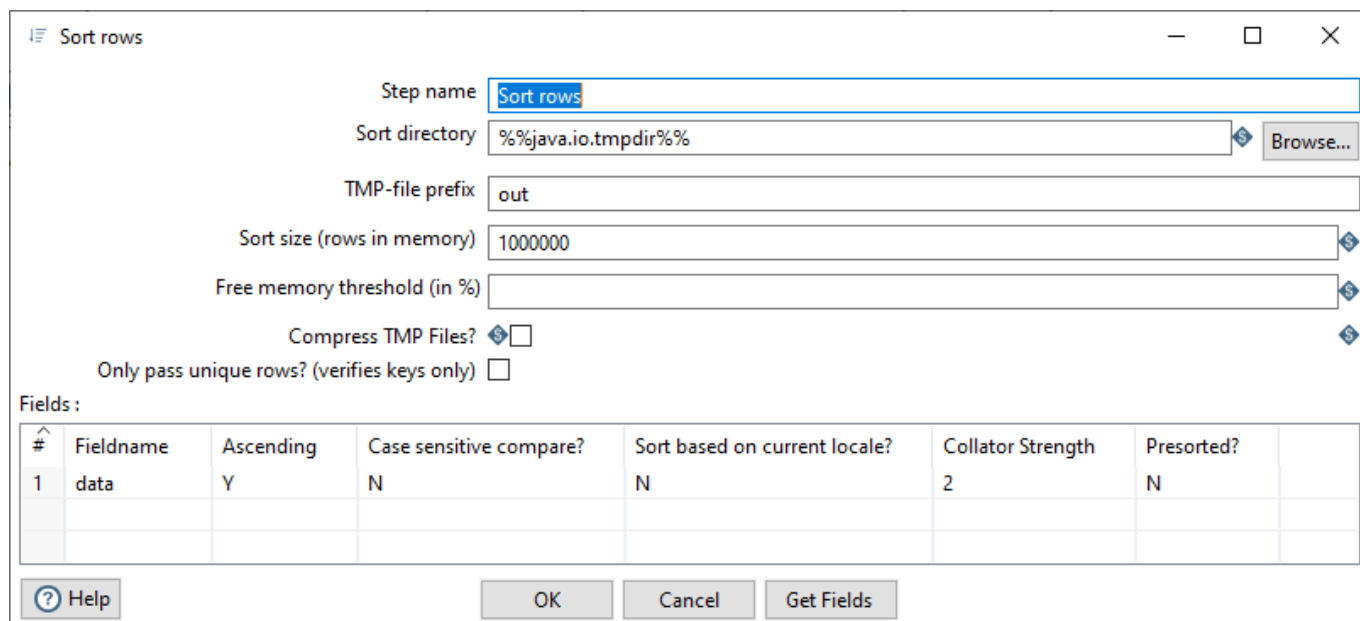
Redirect duplicate row ☐ Error description

Fields to compare (no entries means: compare complete row)
Note: values for the selected fields are stored in memory for every record.

#	Fieldname
1	data
2	semanaepi
3	dia
4	mes
5	ano

Help OK Cancel Get

3.6 Ordene os dados em ordem crescente a partir do campo “data”:



Sort rows

Step name:

Sort directory:

TMP-file prefix:

Sort size (rows in memory):

Free memory threshold (in %):

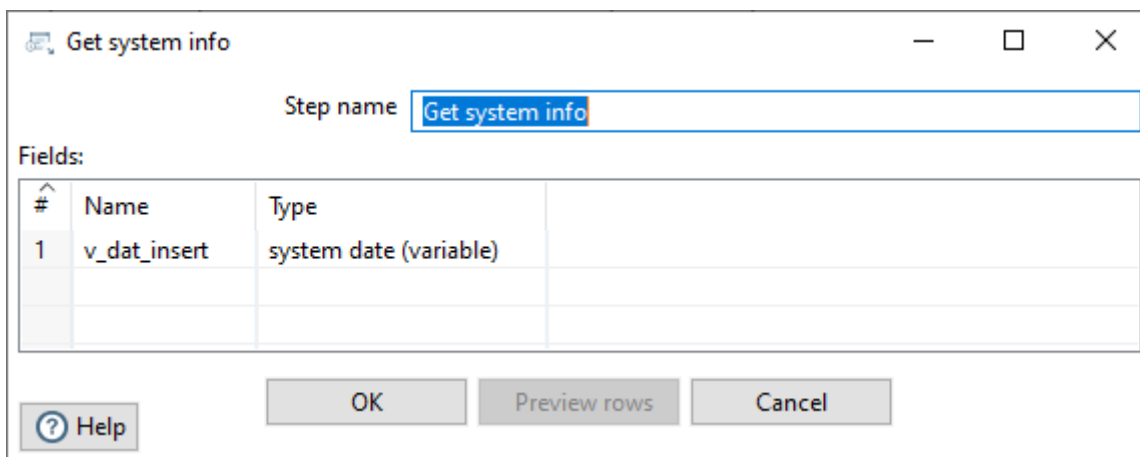
Compress TMP Files? ☐

Only pass unique rows? (verifies keys only) ☐

Fields:

#	Fieldname	Ascending	Case sensitive compare?	Sort based on current locale?	Collator Strength	Presorted?
1	data	Y	N	N	2	N

3.7 Utilizando o step “Get system info”, traga a data e hora do sistema, e a armazene em uma nova variável, que será adiciona ao fluxo:



Get system info

Step name:

Fields:

#	Name	Type
1	v_dat_insert	system date (variable)

3.8 Configure a pk da tabela, sua chave para update, crie e tabela e execute a transformação:

Combination lookup/update

Step name:

Connection:

Target schema:

Target table:

Commit size: Cache size:

Pre-load the cache? ☐

Key fields (to look up row in table):

#	Dimension field	Field in stream
1	data	data
2	semanaepi	semanaepi
3	dia	dia
4	mes	mes
5	ano	ano
6	v_dat_insert	v_dat_insert

Technical key field:

Creation of technical key

☒ Use table maximum + 1

☐ Use sequence

☐ Use auto increment field

Remove lookup fields? ☐

Use hashcode? ☐

Hashcode field in table

Date of last update field: