

Introdução à Plataforma KNIME

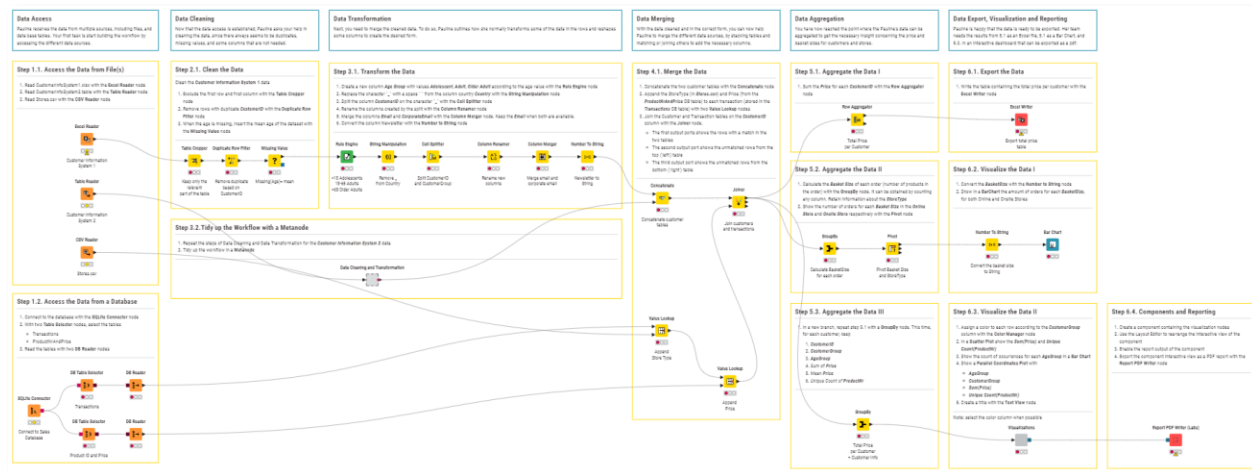
Data Access

Use Case: Customer Transactions Analysis

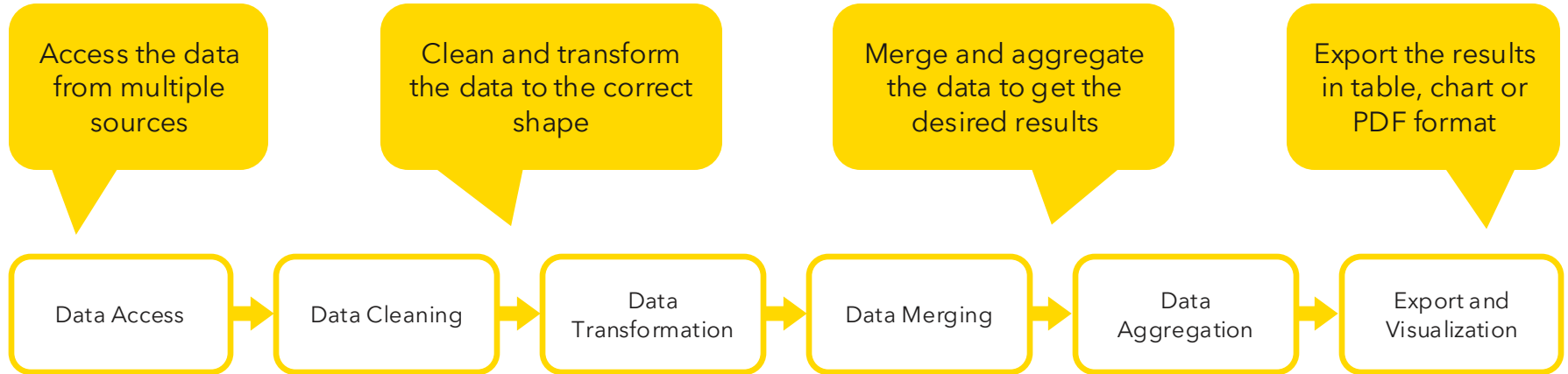
Use Case

Pauline, do setor de Vendas, gostaria de escrever um relatório e visualizar os insights dos dados de transações de clientes que sua equipe coleta todos os meses.

Ela entrou em contato com sua equipe de dados para pedir ajuda na automatização desse processo e, assim, economizar tempo.



Use Case: Customer Transactions Analysis



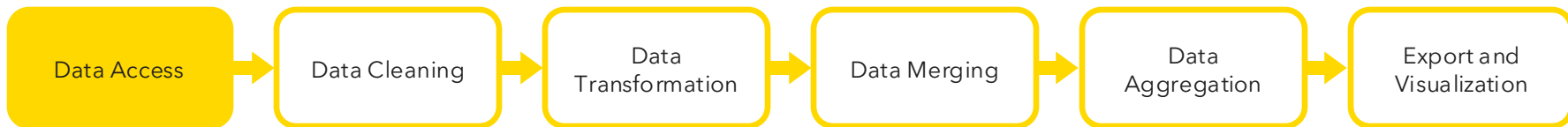
Data Access

- The beginning of every data process
- Data can be stored in many ways
 - Locally
 - In different data format (.csv, .xls...)
 - On the cloud
 - In a database
 - ...
- We need a way to access all of them

Caso de Uso

Pauline recebe os dados de várias fontes, incluindo arquivos e tabelas de banco de dados.

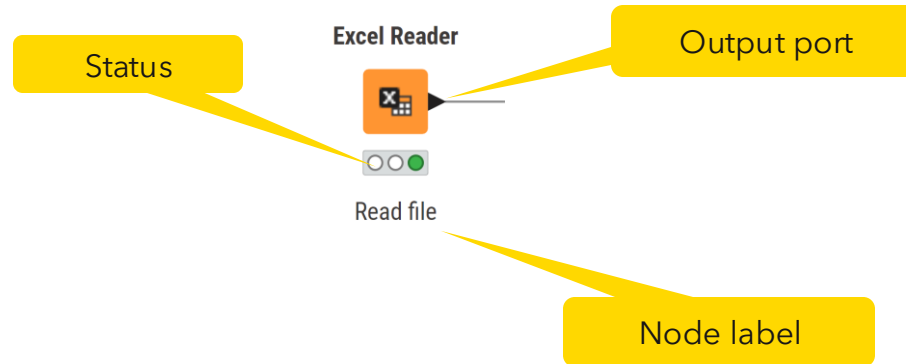
Sua primeira tarefa é começar a construir o fluxo de trabalho acessando essas diferentes fontes de dados.



Data Source Nodes

Typically characterized by:

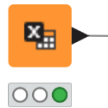
- Orange color
- By default no input ports, 1-2 output ports
- Many nodes for many data formats
- Support reading from different File Systems



Excel Reader

- Reads .xls and .xlsx file from Microsoft Excel
- Supports reading from multiple sheets

Excel Reader



Read Excel Sheet Names



Excel Reader

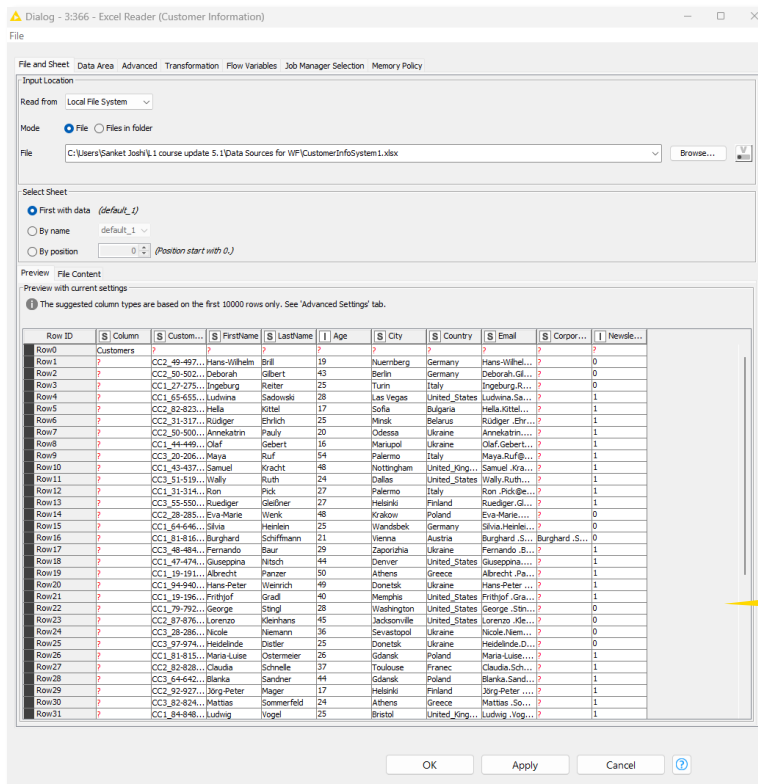
Excel Reader



File system

Sheet specific settings

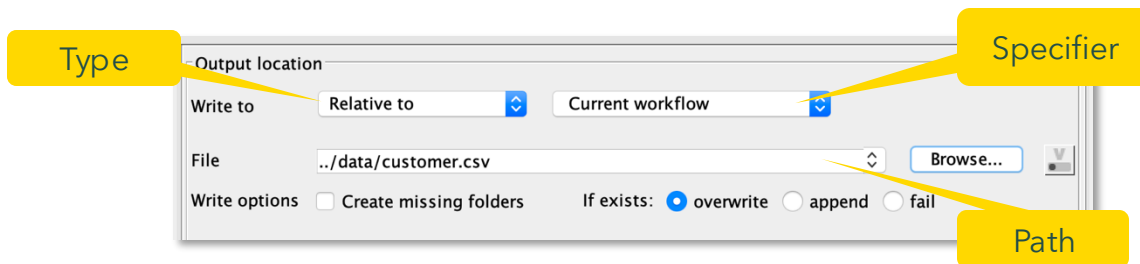
File path



Preview

Common Settings: File Path

- A path consists of three parts:
 - **Type**: Specifies the file system type - e.g., local, relative, mountpoint, custom URL or connected
 - **Specifier**: Optional string with additional file system specific information - e.g. relative to which location (knime.workflow, LOCAL mountpoint...)
 - **Path**: Specifies the location within the file system



- Examples:
 - (LOCAL, , C:\Users\username\Desktop)
 - (RELATIVE, knime.workflow, file1.csv)
 - (MOUNTPOINT, MOUNTPOINT_NAME, /path/to/file1.csv)
 - (CONNECTED, amazon-s3:eu-west-1, /mybucket/file1.csv)

Common Settings: Four Default File Systems

- Local File System

Input location

Read from: Local File System

Mode: ☒ File ☐ Files in folder

File: /Users/kathrinmelcher/Desktop/course_data.csv

Browse...

- Relative to ...

Read from: Relative to

File: Calls_data.xlsx

Browse...

Current mountpoint
Current workflow data area
Current workflow

- Mountpoint

Read from: Mountpoint

LOCAL

File: /Example Workflows/TheData/Customers/CallsData.xls

Browse...

Caminho base virtual (como
knime://knime.mountpoint/)

- Custom URL

Read from: Custom URL

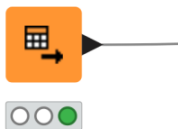
URL: knime://knime.workflow/data/Calls_data.xlsx

Browse...

CSV Reader

- Reads either one or multiple .csv and .txt files
- Further tabs to
 - Select columns
 - Limit the rows
 - Handle quotes
 - Select encoding

CSV Reader



Read data.csv

File system

File path

Preview

Advanced settings

Basic settings

Help button

Dialog - 3:367 - CSV Reader (Stores.csv)

File

Settings Transformation Advanced Settings Limit Rows Encoding Flow Variables Job Manager Selection Memory Policy

Input location

Read from: Local File System

Mode: ☒ File ☐ Files in folder

File: C:\Users\Sanket Joshi\1.1 course update 5.1\Data Sources for WF\Stores.csv

Reader options

Format

Autodetect format

Column delimiter: , Row delimiter: ☒ Line break ☐ Custom

Quote char: " Quote escape char: \"

Comment char: #

☒ Has column header ☐ Has row ID

☐ Support short data rows ☐ Prepend file index to row ID

Preview

The suggested column types are based on the first 10000 rows only. See 'Advanced Settings' tab.

Row ID	StoreID	StoreT...
Row0	Store_2348	Online Store
Row1	Store_2349	Online Store
Row2	Store_2350	Online Store
Row3	Store_2351	Online Store
Row4	Store_2352	Online Store
Row5	Store_2353	Online Store

OK Apply Cancel

CSV Reader

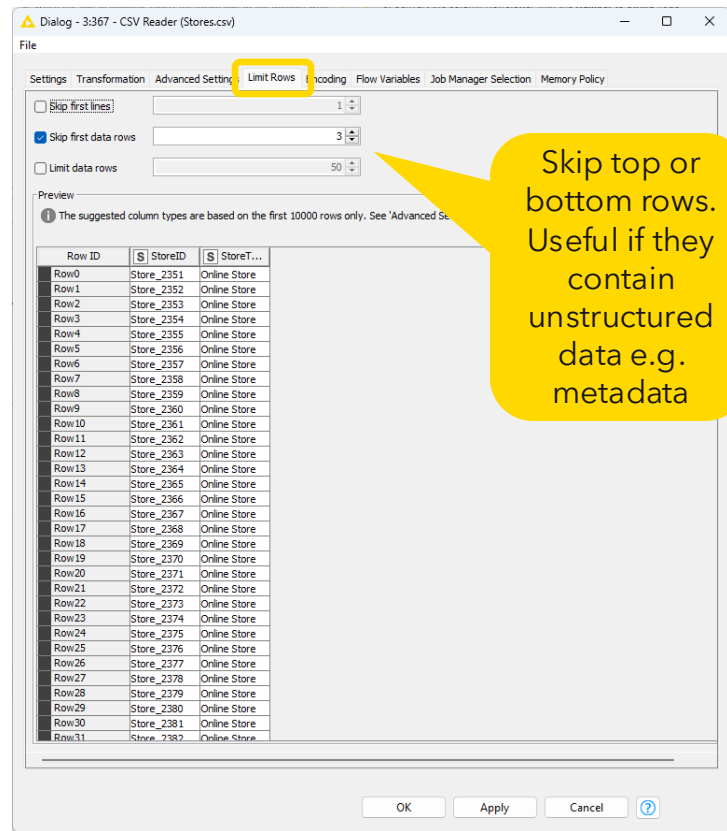
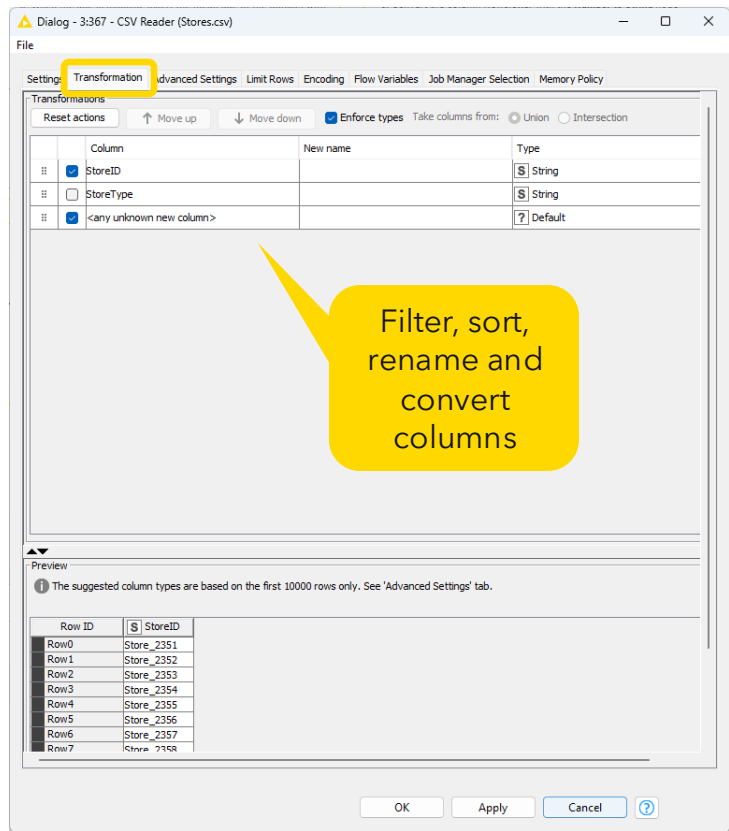
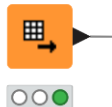


Table Reader

- Reads tables from the native KNIME Format
- Maximum performance, minimum configuration

Table Reader



File system

File path

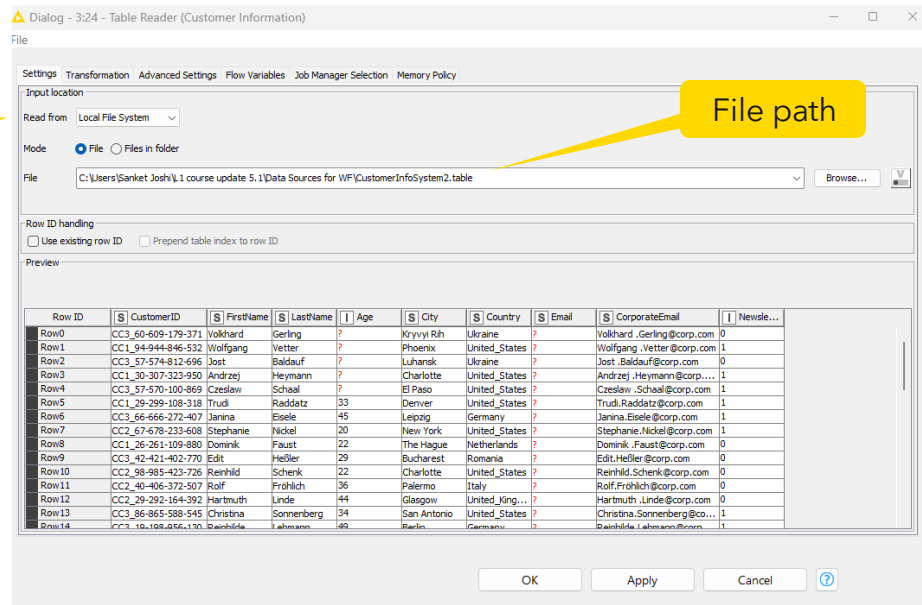
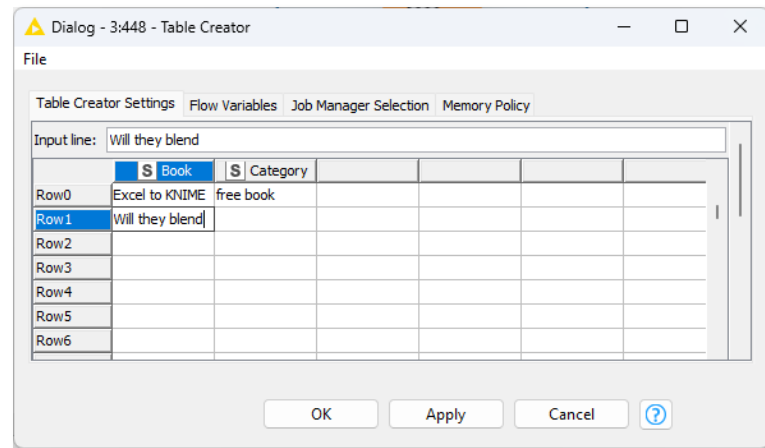


Table Creator

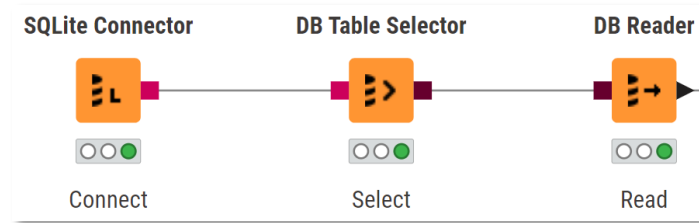
- Create data tables manually
- Enter data in a spreadsheet-like configuration window

Table Creator



Database Connectivity

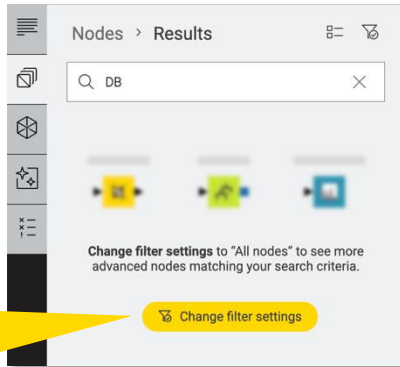
- Read data from any JDBC enabled database
- Write your own SQL or define query using dedicated nodes



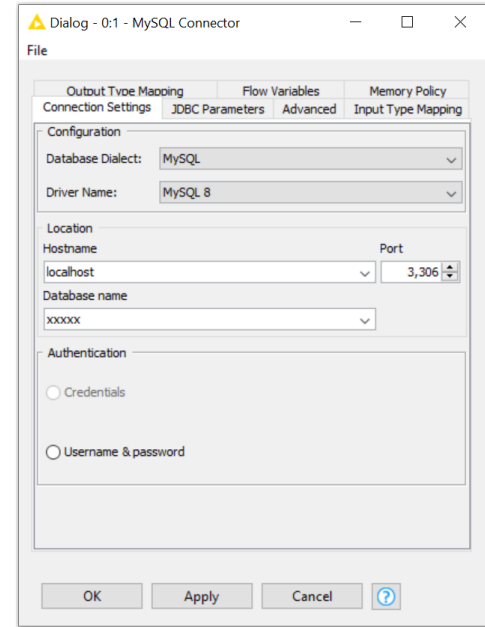
Database Connectors

- Native: PostgreSQL, MySQL, MS SQL Server, SQLite
- DB Connector (e.g., DB2, HANA)
- Big Data: HIVE and Impala

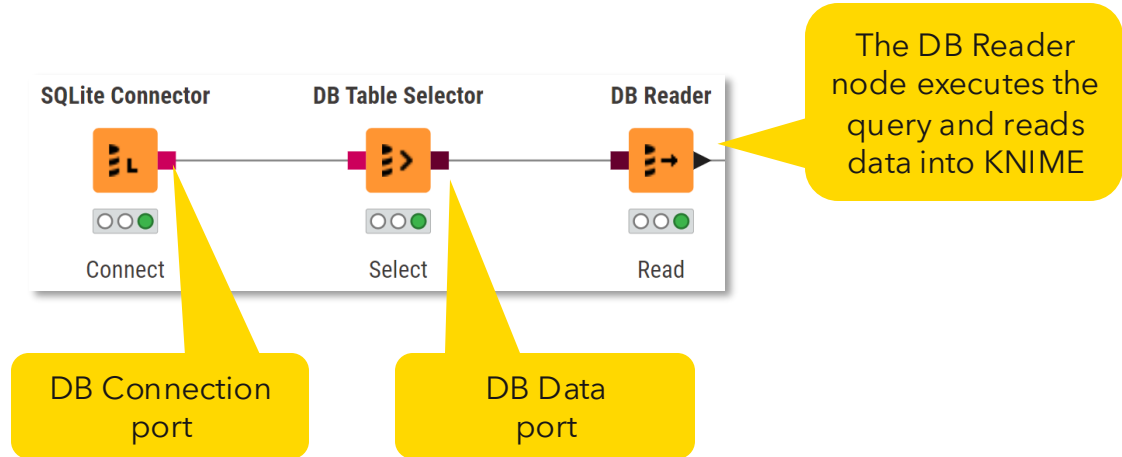
Change the filter settings to see these nodes



MySQL Connector



Database Query



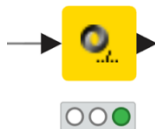
Read JSON format

- Use the JSON Reader (or GET Request) node to get a JSON cell
- Use the JSON Path node to query the JSON file and extract parameters
 - Editor window simplifies construction of JSON queries by auto-generating them

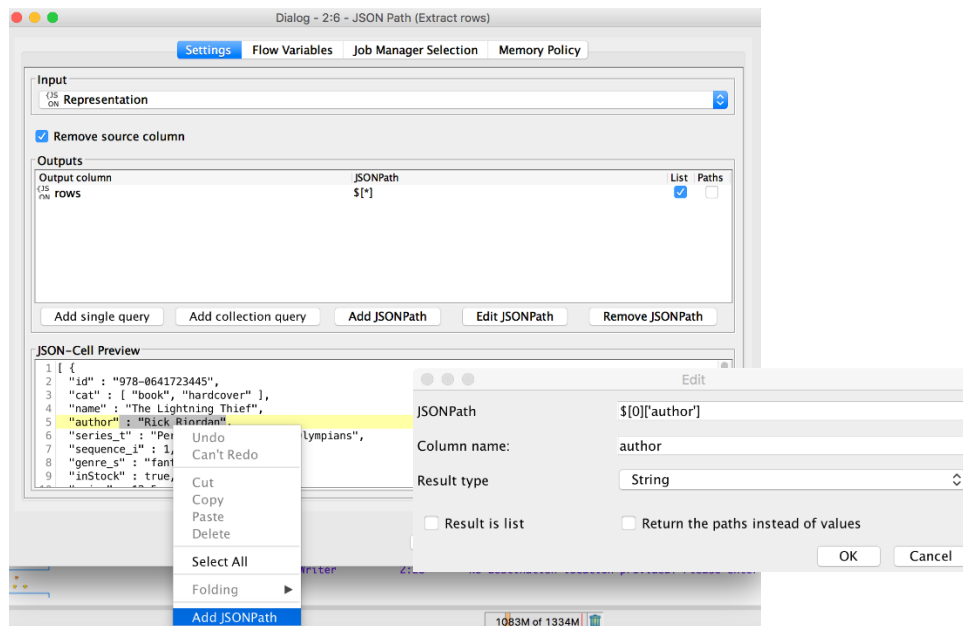
JSON Reader



JSON Path

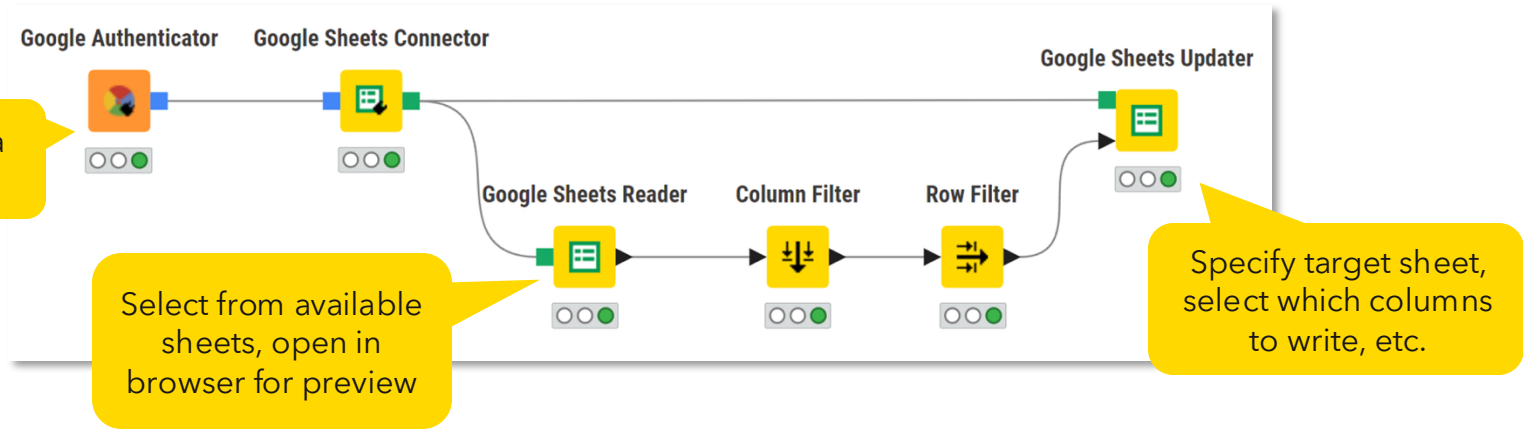


GET Request



Google Sheets

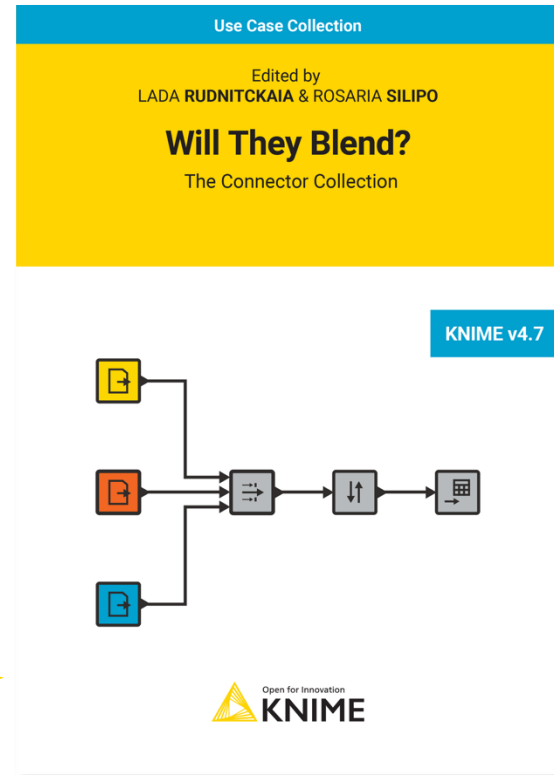
- Select from available sheets on Google Drive
- Transform data in KNIME, or enrich with new data
- Create new sheet or update existing sheets
 - Allows to read from / write to specific range of sheets (e.g. A1:G10)



Other Useful Data Sources

- KNIME Analytics Platform provides many more options to access data:
 - Azure Data Lake Storage
 - Snowflake Connector
 - SMB Connector (e.g. Samba and Windows Server)
 - Python/R Source nodes
 - Tika Parser - extracts textual data from 200+ file types

[Download the free book](#)



Exercises Session 1

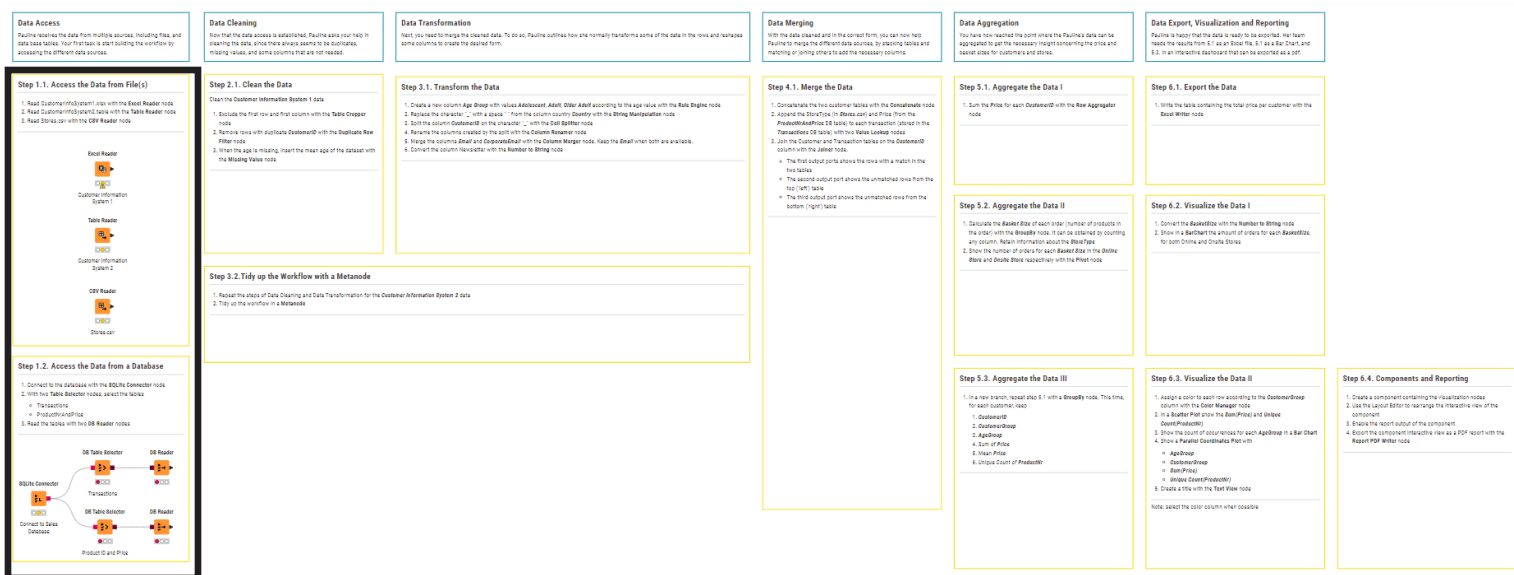
- **Step 00**

- Download and install KNIME Analytics Platform
- Get familiar with the User Interface and create your first workflow
- Download and import the training workflows for this course
- Open the **Customer Transactions Analysis - Exercise** workflow and complete today's steps

You can download the training workflows from the KNIME Community Hub
hub.knime.com/knime/spaces/Education/latest/Courses

Exercises Session 1

- **Step 1.1** – Read customers and stores data from .xlsx, .table, and .csv files
- **Step 1.2** – Read transactions and products data from database



- Compare your results with the *Customer Transactions Analysis - Solution* workflow

Session 2

Data Cleaning, Data Transformation, and Workflow Documentation

Learning Objectives

1. Filter rows and columns
2. Transform values in cells
3. Transform tables
4. List the best practices to organize and document workflows

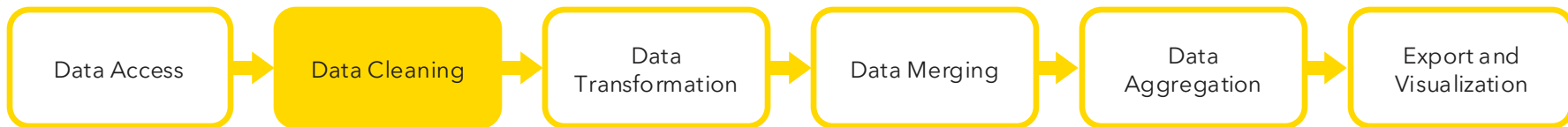
Data Cleaning

Data Cleaning

- Data are rarely clean
- Remove not useful data
- Remove repeated data
- Handle missing values

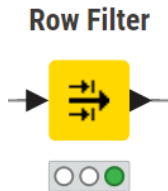
Caso de Uso

Agora que o acesso aos dados foi estabelecido, Pauline solicita sua ajuda para limpar os dados, pois há duplicatas, valores ausentes e algumas colunas que não são necessárias.



Row Filter

- Row filtering with include and exclude options according to certain criteria
 - Certain values or patterns in the selected column
 - Row number
 - Row ID



Dialog - 4:395 - Row Filter

Filter

Criterion 1 ↑ ↓ ✕

Filter column	Operator
Age ▼	> ▼

Value

25 ▲ ▼

⊕ Add criterion

Output

Column domains

Retain Compute

Filter behavior ⚠ ?

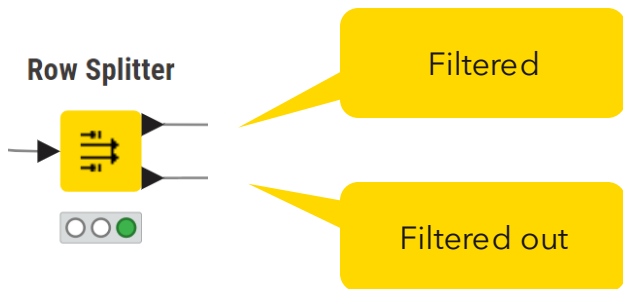
Output matching rows Output non-matching rows

Output non-matching rows

Cancel Ok

Row Splitter

- Same configuration as Row Filter
- Filtered out rows are available in second output port



Dialog - 4:395 - Row Splitter

Filter

Criterion 1

Filter column: Newsletter Operator: =

Value: 1

+ Add criterion

Output

Column domains

Retain Compute

Splitting behavior

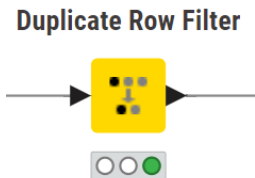
☒ Matching rows at first output, non-matching at second output

☐ Non-matching rows at first output, matching at second output

Cancel Ok

Duplicate Row Filter

- Detect duplicate rows and apply a selected treatment
 - Select columns to check for duplicates
 - Provide options for treating duplicated values



Flag or Remove Duplicates

Select criteria to keep row

Dialog - 3:316 - Duplicate Row Filter (Remove duplicate)

Duplicate detection

Choose columns for duplicates detection

Manual Wildcard Regex Type

Search Aa

Excludes

- FirstName
- LastName
- Age
- City
- Country
- Email

Any unknown columns

Includes

- CustomerID

Duplicate handling

Duplicate rows

☒ Remove duplicate rows

☐ Keep duplicate rows

Row chosen in case of duplicate

☒ First

☐ Last

☐ Minimum of

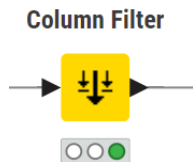
☐ Maximum of

[Show advanced settings](#)

Cancel Ok

Column Filter

- Remove columns from table



CustomerID	FirstName	LastName	Age	Newsletter
?	?	?	?	?
6589	Peter	Parker	31	yes
6768	Bruce	Banner	32	no
6925	Natasha	Romanoff	34	no

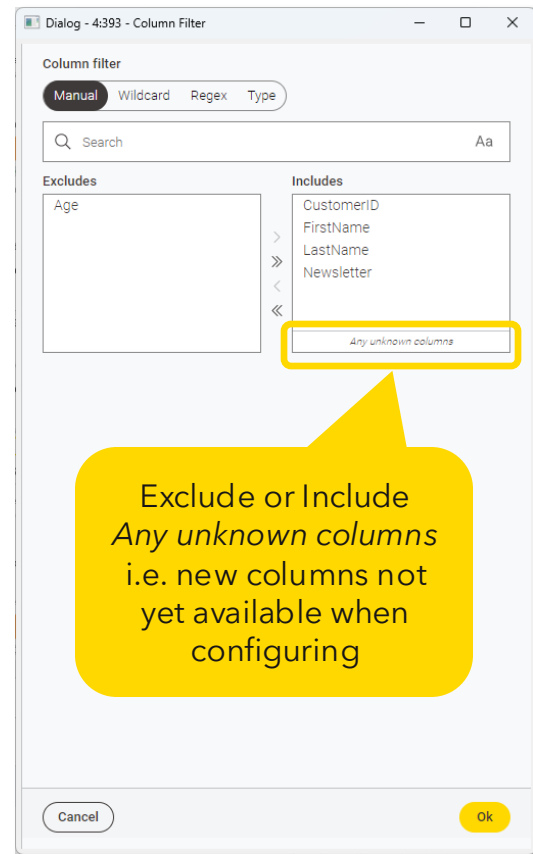
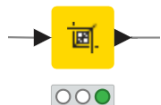


Table Cropper (recortador)

- Retain a selected, contiguous portion of a table
 - Select starting and ending rows and columns

Table Cropper



CustomerID	FirstName	LastName	Age	Newsletter
?	?	?	?	?
6589	Peter	Parker	31	yes
6768	Bruce	Banner	32	no
6925	Natasha	Romanoff	34	no

Dialog - 4:394 - Table Cropper

Columns

Column range mode
☒ By name ☐ By number

Start column
FirstName

End column (inclusive)
Newsletter

Rows

Start row number
2

☐ Start counting rows from the end of the table

End row number (inclusive)
1

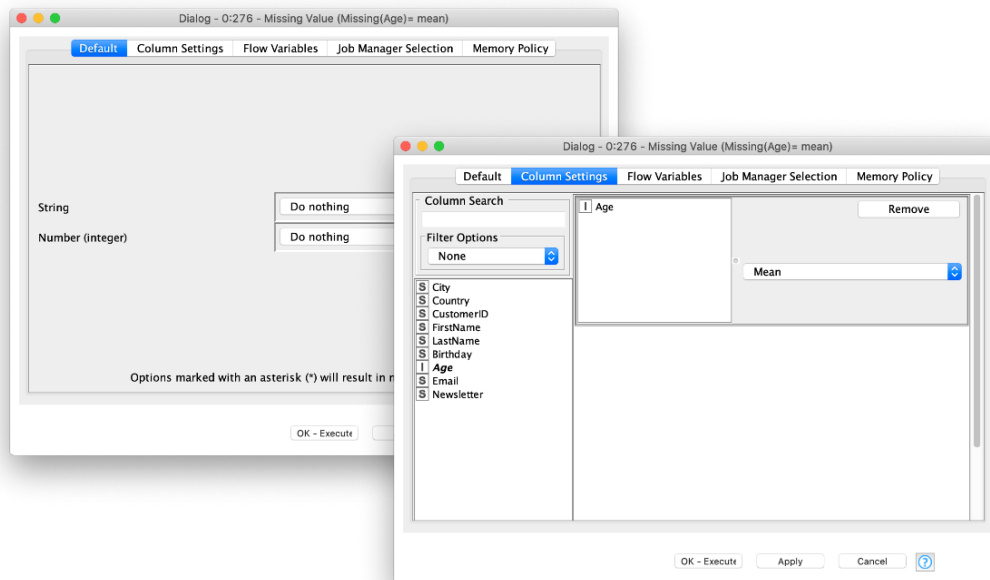
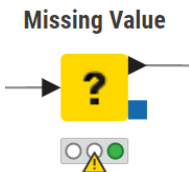
☒ Start counting rows from the end of the table

[Show advanced settings](#)

Cancel Ok

Missing Value

- Define how to handle missing values for all columns of a given type
 - Affect all columns that are not explicitly mentioned in the second tab
- Define how to handle missing values for each available column



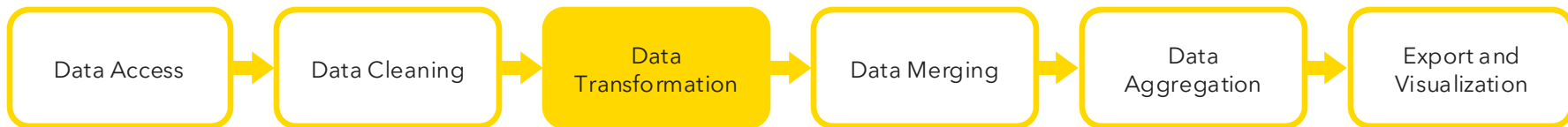
Data Transformation

Data Transformation

- The core of the data pipeline
- Extract more information
 - Define rules and mathematical operations
 - Transform at cell or row level
- Transform the data to the desired shape
 - Rename and resort table columns
 - Split and merge columns
 - Convert data types

Caso de Uso

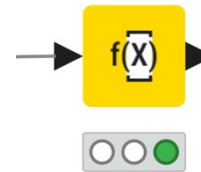
Em seguida, você precisa mesclar os dados limpos. Para isso, Pauline descreve como normalmente transforma alguns dos dados nas linhas e reorganiza algumas colunas para criar o formato desejado.



Expression

- String operations
 - Capitalize
 - Join strings
 - ...
- Math operations
 - Round
 - Sum
 - Max
 - ...
- Custom logic with if-clause

Expression



Dialog - 4:393 - Expression (Remove...)

Expression editor

```
1 replace(${ "Country" }, "- ", " ")
```

Append Replace Country

Ask K-AI Evaluate first 10 rows

String - Extract & Replace

- first_chars(string, n)
- last_chars(string, n)
- substring(string, start, ...)
- regex_extract(string, pattern, ...)
- replace(string, pattern, ...)**
- regex_replace(string, pattern, ...)
- find(string, search, ...)
- find_chars(string, chars, ...)
- count(string, search, ...)
- count_chars(string, search, ...)

replace(string, pattern, replace, modifiers)

Arguments

- string: Input string to perform replacements on
- pattern: Pattern to search for
- replace: Replacement text
- modifiers: (optional), "i" for case-insensitive matching (root locale), "w" to match whole words only

Console Output table

Preview computed on first 10 rows of 2374 rows

RowID	Customer...	FirstName	LastName	Age	City	Country	Email
	String	String	String	Number (dou...	String	String	String
Row3	CC1_27-275-...	Ingeburg	Reiter	25	Turin	Italy	Ingeburg.Reit
Row4	CC1_65-655-...	Ludwina	Sadowski	28	Las Vegas	United States	Ludwina.Sad
Row5	CC2_82-823-...	Hella	Kittel	17	Sofia	Bulgaria	Hella.Kittel@

Cancel Ok

Expression

Select columns from input table

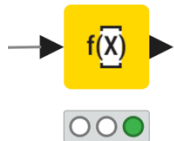
Edit the expression

Select a function from the list

Read the documentation

Preview of the output

Expression



Dialog - 4:393 - Exp

Input table

- CustomerID STRING
- FirstName STRING
- LastName STRING
- Age FLOAT
- City STRING
- Country STRING
- Email STRING
- CorporateEmail STRING
- Newsletter INTEGER
- AgeGroup STRING

Flow variables

- knime.workspace STRING

Output table

Expression editor

```
1 replace(${Country}, "-", " ")
```

Search

String - Extract & Replace

- first_chars(string, n)
- last_chars(string, n)
- substring(string, start, ...)
- regex_extract(string, pattern, ...)
- replace(string, pattern, ...)**
- regex_replace(string, pattern, ...)
- find(string, search, ...)
- find_chars(string, chars, ...)
- count(string, search, ...)
- count_chars(string, search, ...)

replace(string, pattern, replace, modifiers)

Arguments

- string: Input string to perform replacements on
- pattern: Pattern to search for
- replace: Replacement text
- modifiers: (optional), "i" for case-insensitive matching (root locale), "w" to match whole words only

Append Replace Country

Ask K-AI Evaluate first 10 rows

Console Output table

Preview computed on first 10 rows of 2374 rows

RowID	Customer... String	FirstName String	LastName String	Age Number (dou...	City String	Country String	Email String
Row3	CC1_27-275-...	Ingeburg	Reiter	25	Turin	Italy	Ingeburg.Rei...
Row4	CC1_65-655-...	Ludwina	Sadowski	28	Las Vegas	United States	Ludwina.Sad...
Row5	CC2_82-823-...	Hella	Kittel	17	Sofia	Bulgaria	Hella.Kittel@...

Cancel Ok

Expression with K-AI Copilot

Input table

- CustomerID
- FirstName
- LastName
- Age
- City
- Country
- Email
- CorporateEmail
- Newsletter

Flow variables

- knime.workspace

Output table

Dialog - 4:394 - Expression (<18 Adolescents)

```
1 if(condition_1, value_1, additional_conditions, value_if_all_false)
2 if($["Age"] < 18, "Adolescent", $["Age"] <= 65, "Adult", "Older Adult")
```

Callouts:

- K-AI provides edits (green) to the existing expression (red)
- Accept the edits replace the expression in the editor
- Write a prompt to describe what you expect
- Install the [AI Assistant extension](#)

AI Assistant Interface:

Create a new column AgeGroup with values Adolescent, Adult, Older Adult according to the age value: <18 Adolescents, 18-65 Adults, >65 Older Adults

Type your prompt

Buttons: Insert in editor, Ask K-AI, Evaluate first 10 rows

Output column: AgeGroup

Console / Output table:

Cell Splitter

- Split the content of one column into many columns based on a delimiter

Filtered table - 2:311 - Column Filter

File Hilite Navigation View

Table "default" - Rows: 66401

Row ID	S ProductNr
Row0	J-241-1982
Row11	F-19-1987
Row22	Z-160-1990
Row23	H-16-2005
Row33	A-106-1988
Row34	V-208-1988
Row35	Q-130-1989
Row36	Q-136-1998

Cell Splitter



Output Table - 2:161 - Cell Splitter (Split on "-")

File Hilite Navigation View

Table "default" - Rows: 66401

Row ID	S	ProductNr	S ...	I P...	I Pro...
Row0	J	241	1982		
Row11	F	19	1987		
Row22	Z	160	1990		
Row23	H	16	2005		
Row33	A	106	1988		
Row34	V	208	1988		
Row35	Q	130	1989		
Row36	Q	136	1998		

Dialog - 0:218 - Cell Splitter

Settings Flow Variables Job Manager Selection Memory Policy

Column to split

Select a column: ☒ Remove input column

Settings

Enter a delimiter: ☐ Use \ as escape character

Enter a quotation character: (leave empty for none.)

☒ Remove leading and trailing white space chars (trim)

Output

☐ As list ☐ As set (remove duplicates) ☒ As new columns

☐ Split input column name for output column names

☐ Set array size

☒ Guess size and column types (requires additional data table scan)

☐ Scan limit (number of lines to guess on)

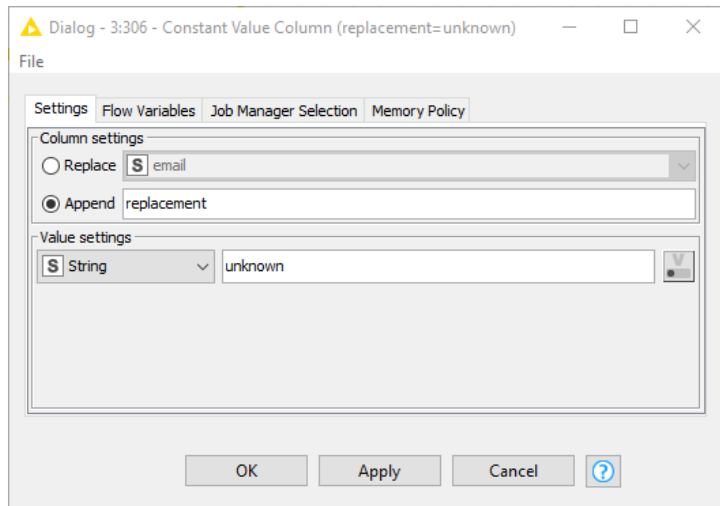
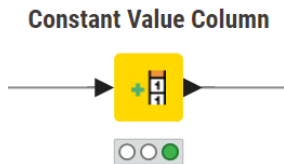
Missing Value Handling

☐ Create empty string cells instead of missing string cells

OK - Execute Apply Cancel ?

Constant Value Column

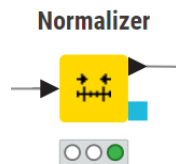
- Add or replace a column with a single constant value
- Can be used to add an empty column



Normalizer

- Normalize values of numeric columns
 - Min-Max Normalization
 - Z-Score (Gaussian) Normalization
 - Decimal Scaling

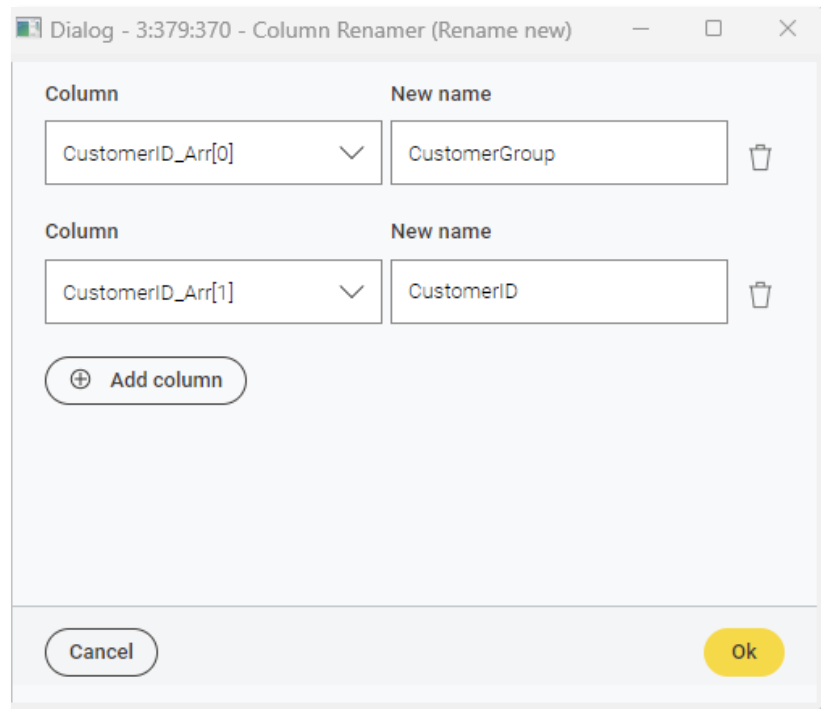
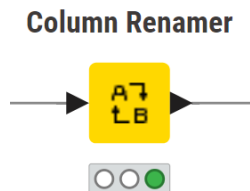
User	Age	Income
user1	34	30000
user2	20	45000
user3	59	20000
user4	60	100000



User	Age	Income
user1	0.35	0.125
user2	0	0.312
user3	0.975	0
user4	1	1

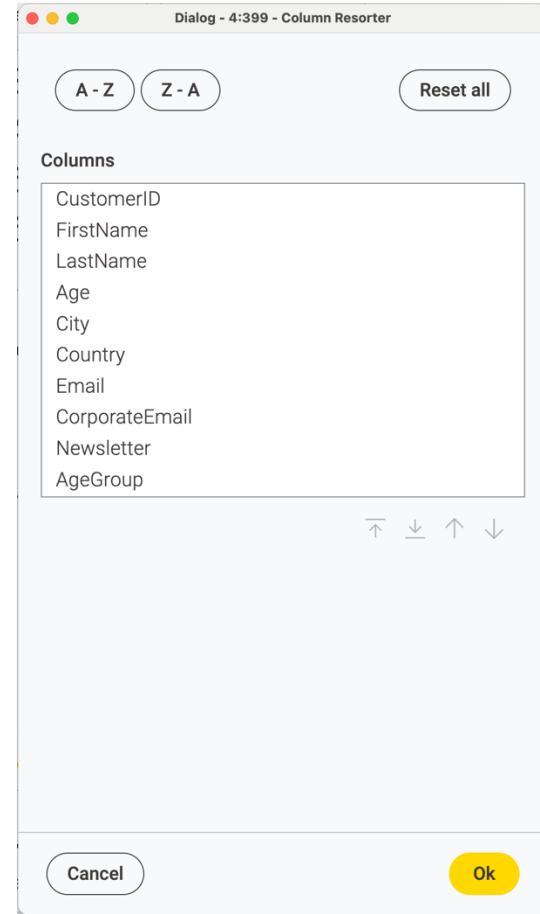
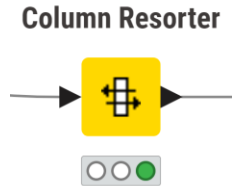
Column Renamer

- Change name of one or more columns



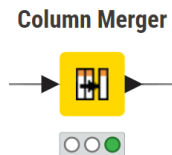
Column Resorter

- Change the order of the columns in a table



Column Merger

- Merge two columns into one by choosing the cell that is not missing



Dialog - 4:321 - Column Merger (Merge email and)

Primary column
Email

Secondary column
CorporateEmail

Replace/append columns

☒ Replace primary and delete secondary

☐ Replace primary

☐ Replace secondary

☐ Append as new column

Cancel Ok

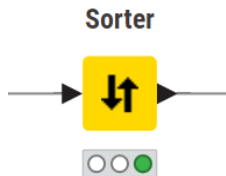
Email	Corporate Email
peter.parker@gmail.com	?
?	blackw@marvel.com
bbanner@gmail.com	thehulk@marvel.com



Email
peter.parker@gmail.com
blackw@marvel.com
bbanner@gmail.com

Sorter

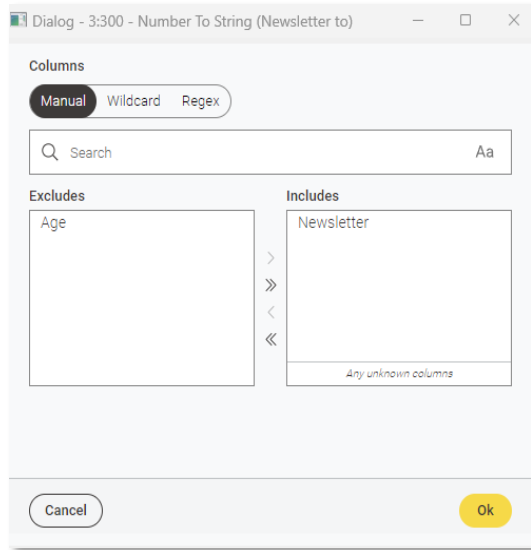
- Sort the rows based on the values of the selected column(s), either
 - ascending or
 - descending



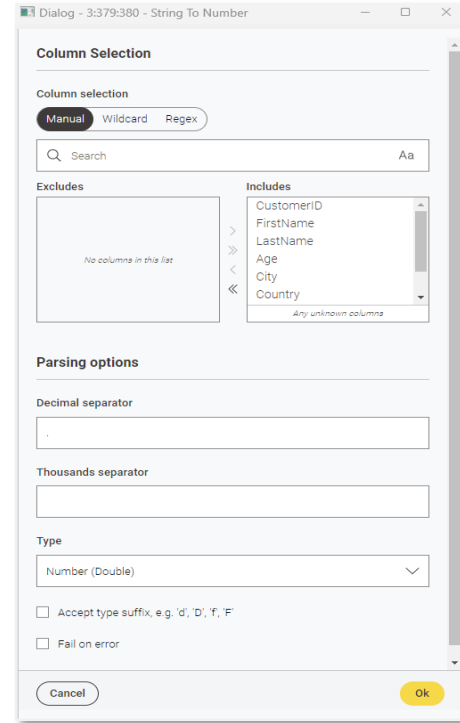
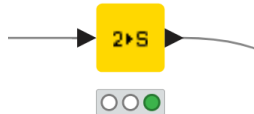
The screenshot shows the 'Dialog - 4:399 - Sorter' window. The title bar has three colored circles (red, yellow, green) on the left. The main area is titled 'Sorting'. It contains two criteria sections. Each section has a 'Criterion' label, a 'Column' dropdown menu, and an 'Order' section with 'Ascending' and 'Descending' buttons. The first criterion is set to 'Age' and 'Ascending'. The second criterion is set to 'FirstName' and 'Ascending'. Below these sections is a button labeled '+ Add sorting criterion'. At the bottom, there is a 'Show advanced settings' link, a 'Cancel' button, and an 'Ok' button.

Type Conversion

- Change the data type of the selected columns



Number To String



String To Number

