



University of Minho
School of Engineering



Machine Learning with Knime

Similarity Based Systems

Perfil ML:FA@MiEI/4º ano - 1º Semestre

@MES/2º ano - 1º Semestre

Bruno Fernandes, Paulo Novais

17/10/2019

Contents

2

Flow Variables

Metanodes

Data Viz

PMML

Hands On

- Flow Variables
- Metanode Templates and Components
- Data Visualization
 - JavaScript (and Java) integration
 - Composite Views
- PMML
- Hands On

Flow Variables

3

FLOW VARIABLES

Metanodes

Data Viz

PMML

Hands On

Flow Variables are **variables** used inside a workflow to **dynamically overwrite** node settings, i.e., to **parameterize workflows** when node settings need to be determined dynamically

Two of the most important applications for flow variables are the configuration of **Loops** and **Metanodes**

Two ways of creating workflow variables:

- At **Workflow Level** (visible to all nodes)
- or **Inside the Workflow** (visible to the nodes that appear after the variable creation)

Without Flow Variables

4

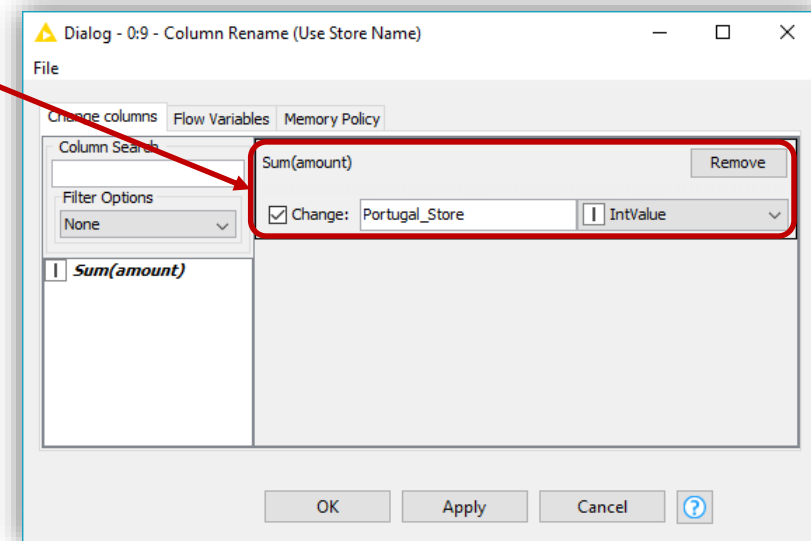
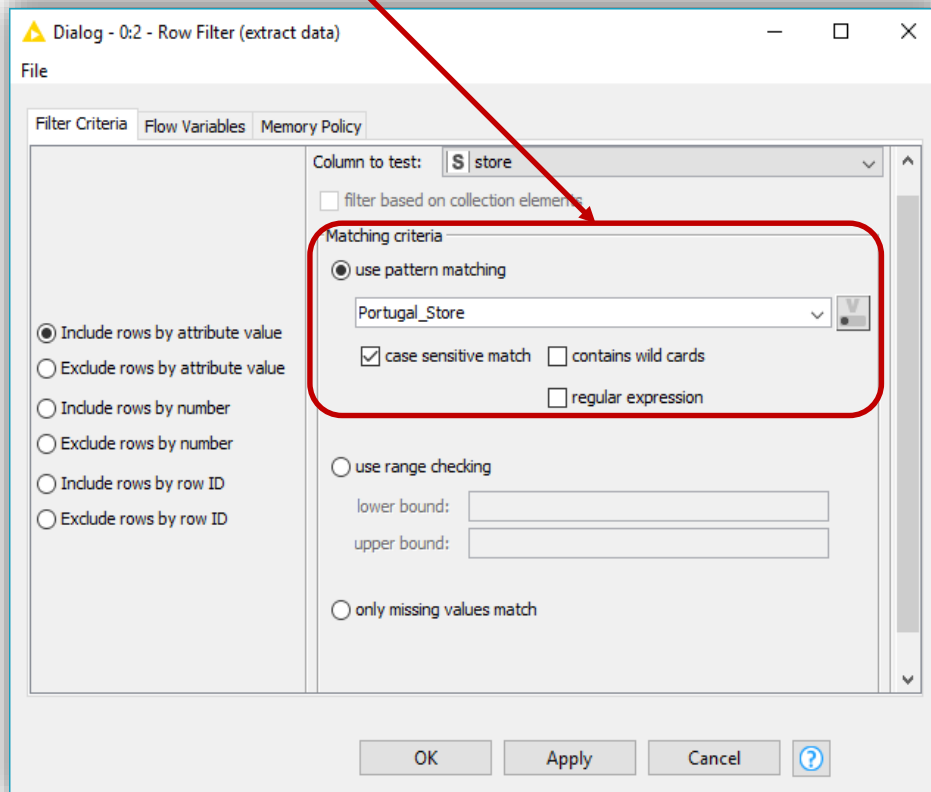
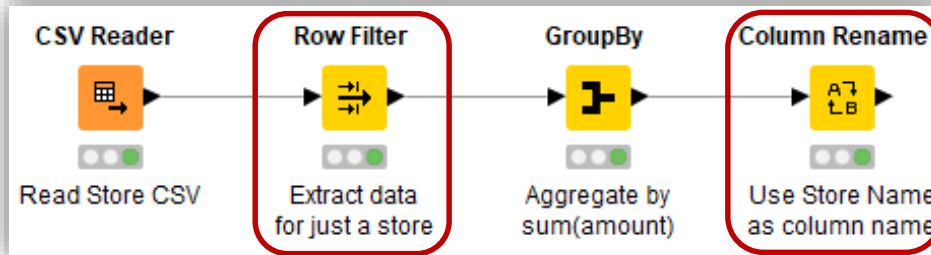
FLOW VARIABLES

Metanodes

Data Viz

PMML

Hands On



And this is a simple flow... A bigger one with more nodes would imply a lot of work just to rename all parameters!

Flow Variables at Workflow Level

5

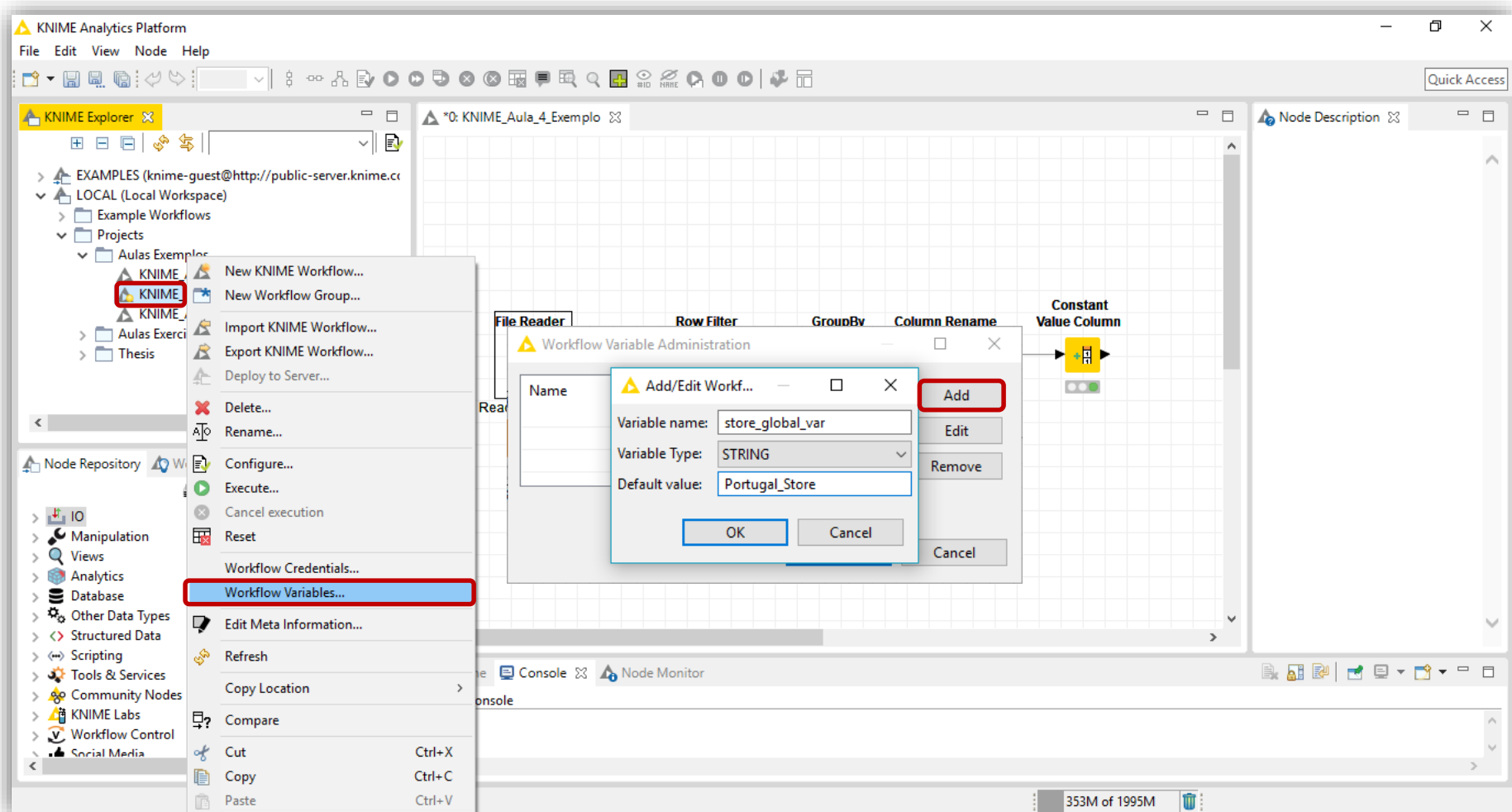
FLOW VARIABLES

Metanodes

Data Viz

PMML

Hands On



Flow Variables at Workflow Level

6

FLOW VARIABLES

Metanodes

Data Viz

PMML

Hands On

The screenshot displays the KNIME Analytics Platform interface. On the left, the 'KNIME Explorer' pane shows a tree view of the workspace. A right-click context menu is open over the 'KNIME' folder, with the 'Workflow Variables...' option highlighted in blue. The main workspace shows a workflow titled '*0: KNIME_Aula_4_Exemplo' with nodes including 'File Reader', 'Row Filter', 'GroupBy', 'Column Rename', and 'Constant Value Column'. A 'Workflow Variable Administration' dialog box is open in the center, showing a table with the following data:

Name	Type	Value
store_global_var	STRING	Portugal_Store

The dialog box includes 'Add', 'Edit', and 'Remove' buttons for the table, and 'OK' and 'Cancel' buttons at the bottom.

Flow Variables Inside the Workflow

7

FLOW VARIABLES

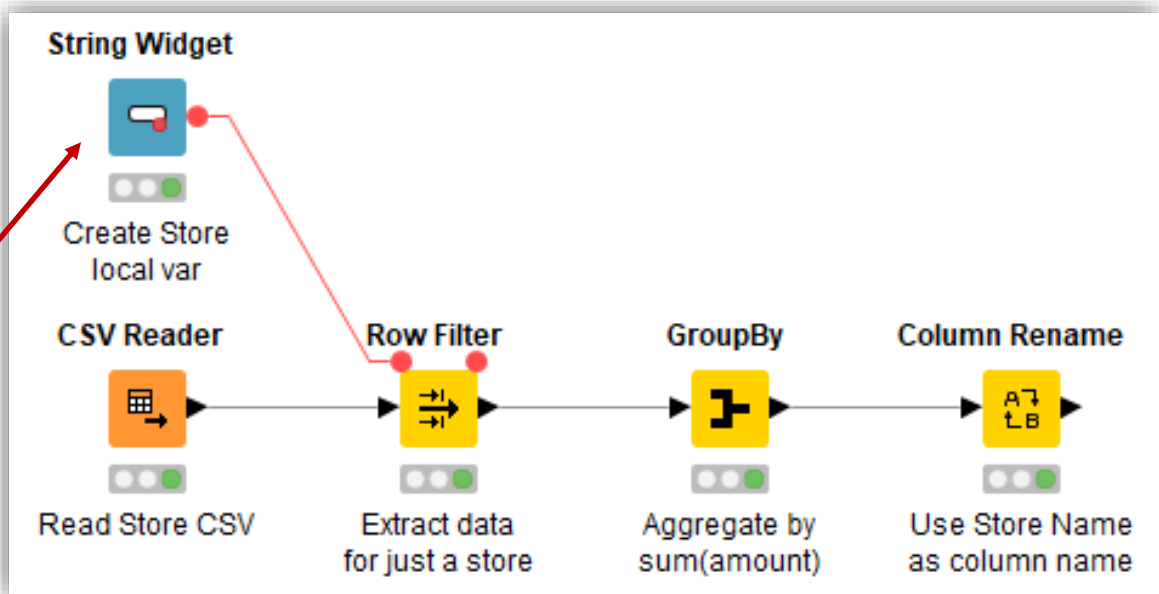
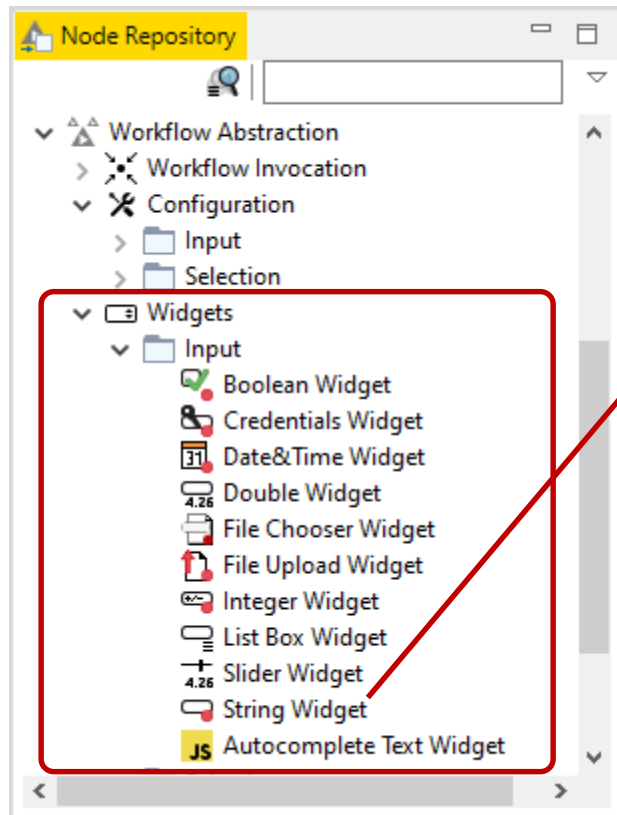
Metanodes

Data Viz

PMML

Hands On

To create a flow variable inside the workflow we should use **Widgets**!



Flow Variables Inside the Workflow

8

FLOW VARIABLES

Metanodes

Data Viz

PMML

Hands On

String Widget



Create Store
local var

Indicates a **Flow Variable**!

Defines the variable's name

Defines the default value

Dialog - 2:31 - String Widget (Create Store)

File

Control Flow Variables Memory Policy

Label: Store

Description: The store from where to extract data.

Variable Name: store_local_var

Editor type: ☒ Single-line ☐ Multi-line

Multi-line editor width: 60

Multi-line editor height: 5

Regular Expression:

Validation Error Message:

Common Regular Expressions: Assign

Default Value: Portugal_Store

OK Apply Cancel ?

Flow Variables Inside the Workflow

9

FLOW VARIABLES

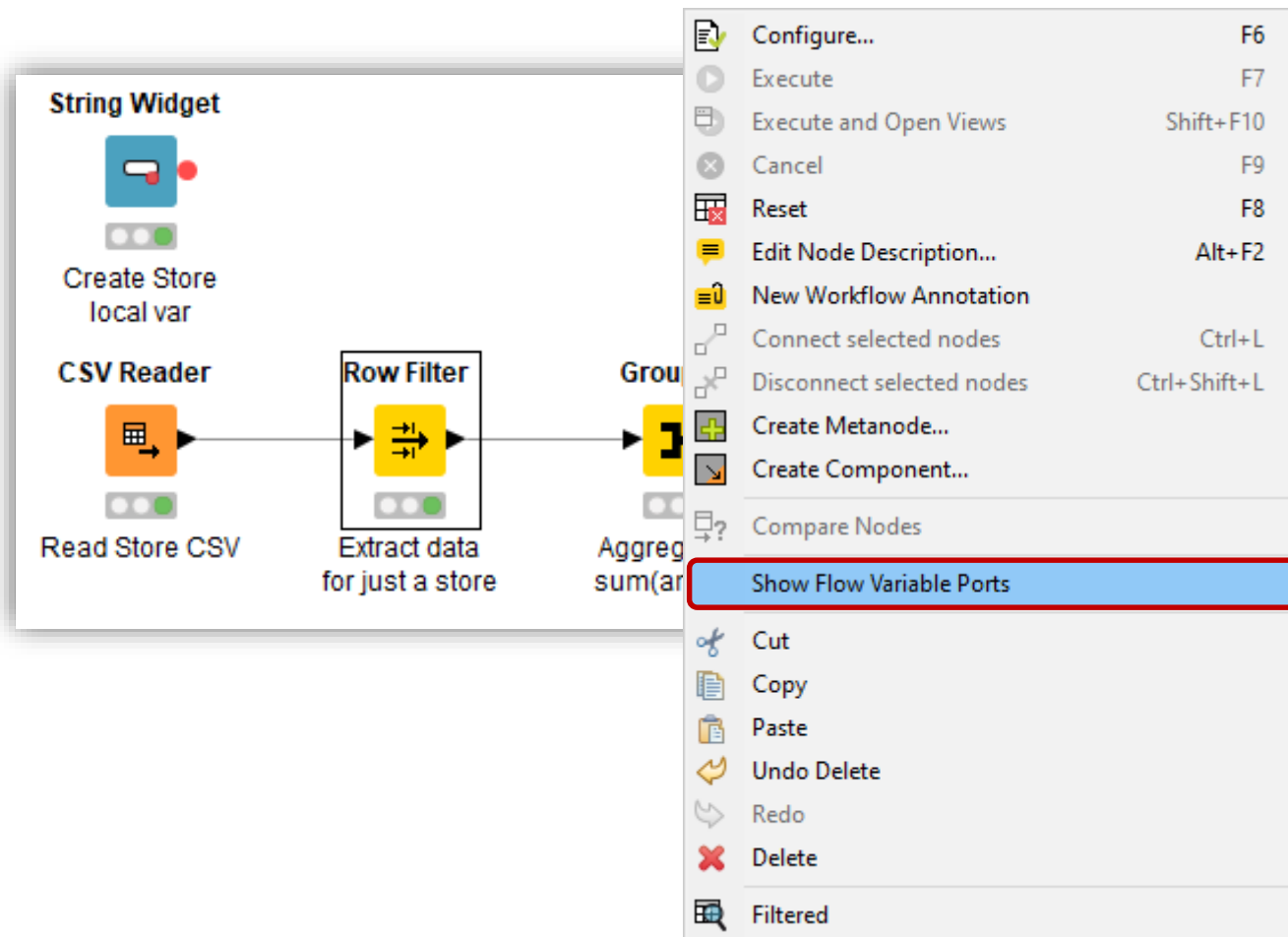
Metanodes

Data Viz

PMML

Hands On

But... How to **connect** the **flow variables** output of the widget to a node's input?



Flow Variables Inside the Workflow

10

FLOW VARIABLES

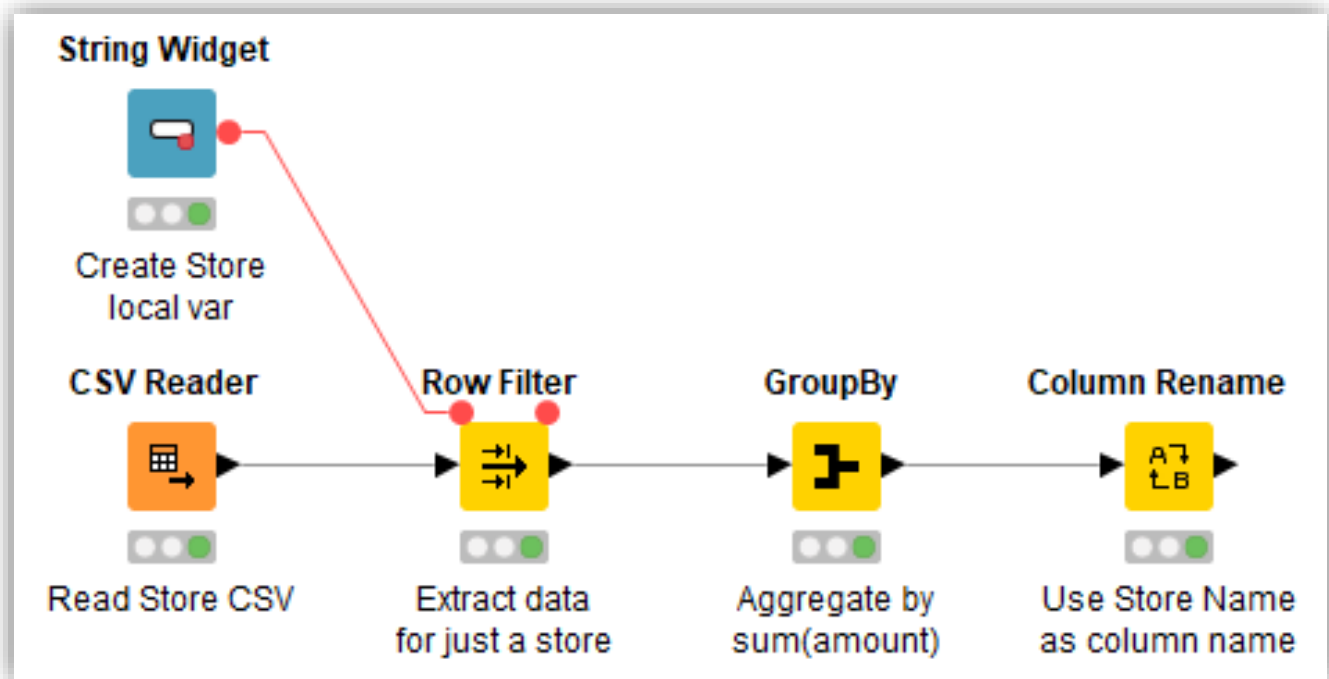
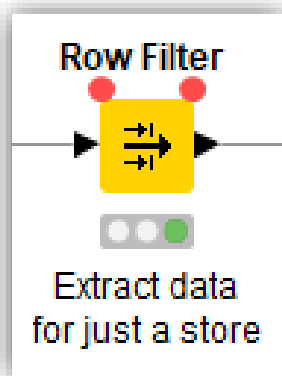
Metanodes

Data Viz

PMML

Hands On

The **variables' input** and **output** of the selected node appears! Then, just connect.



How to use the created Flow Variables

11

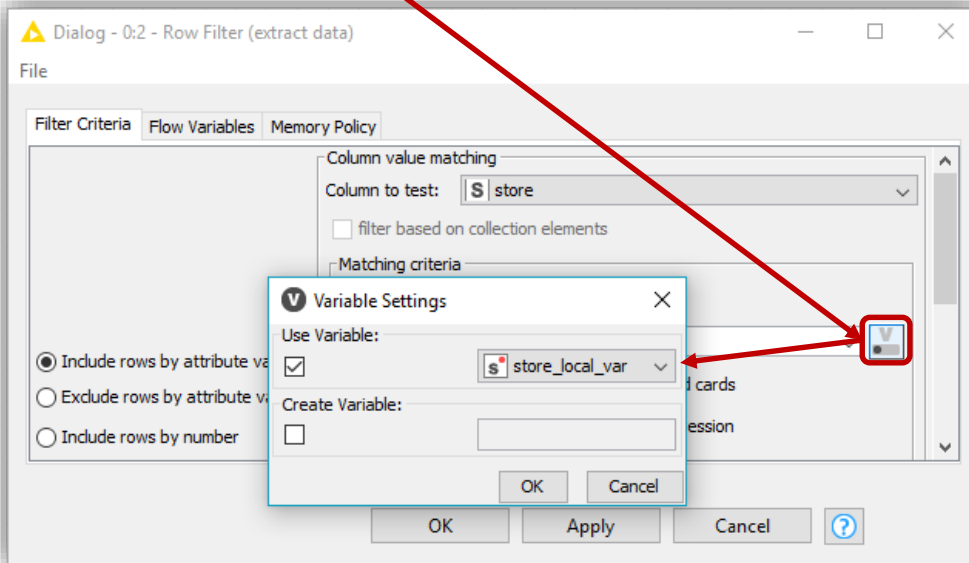
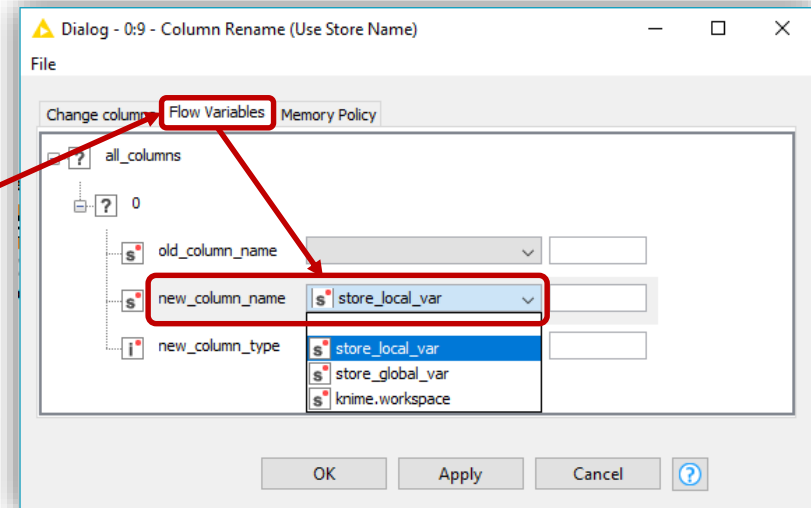
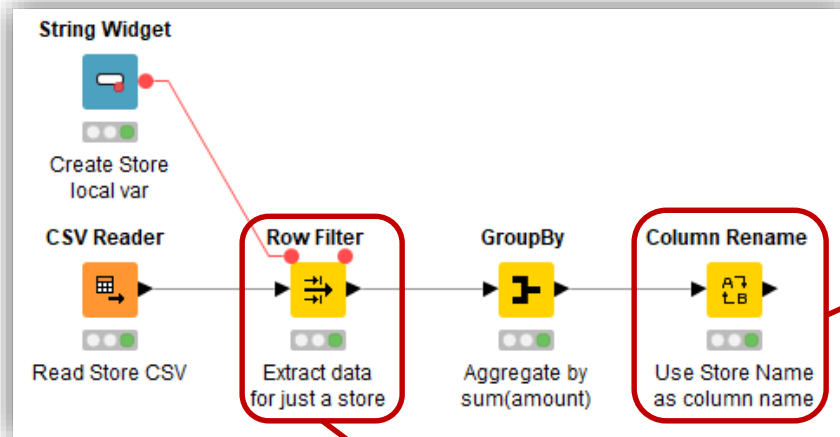
FLOW VARIABLES

Metanodes

Data Viz

PMML

Hands On



Everything is now parameterizable using **Flow Variables**! We can now execute the flow and gather results for the variable's value!

Create More Flow Variables

12

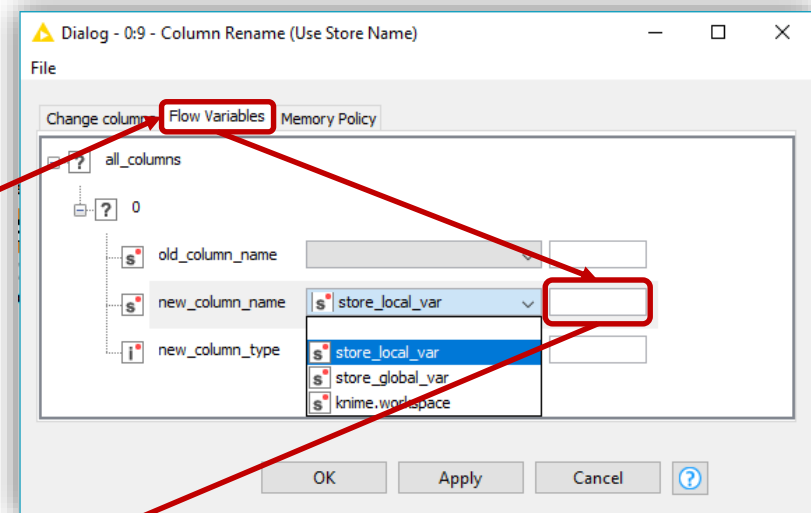
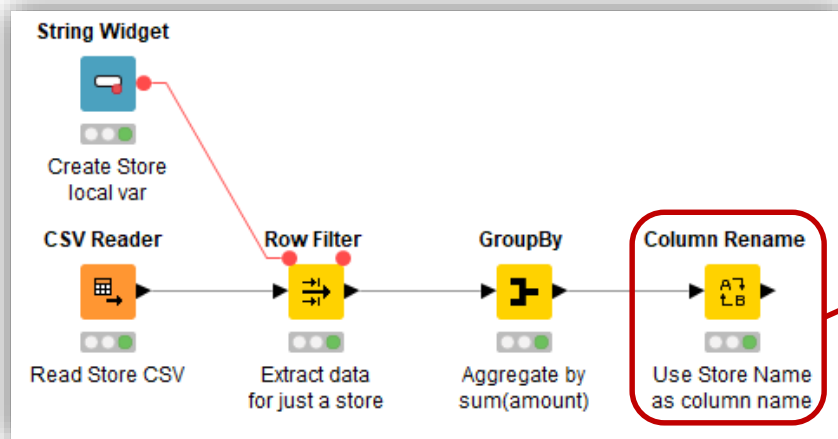
FLOW VARIABLES

Metanodes

Data Viz

PMML

Hands On



We can also create new Flow Variables with the value of a specific node setting. For that just define the name of the new flow variable in the input text near to the desired setting!

Flow Variables available to a Node

13

FLOW VARIABLES

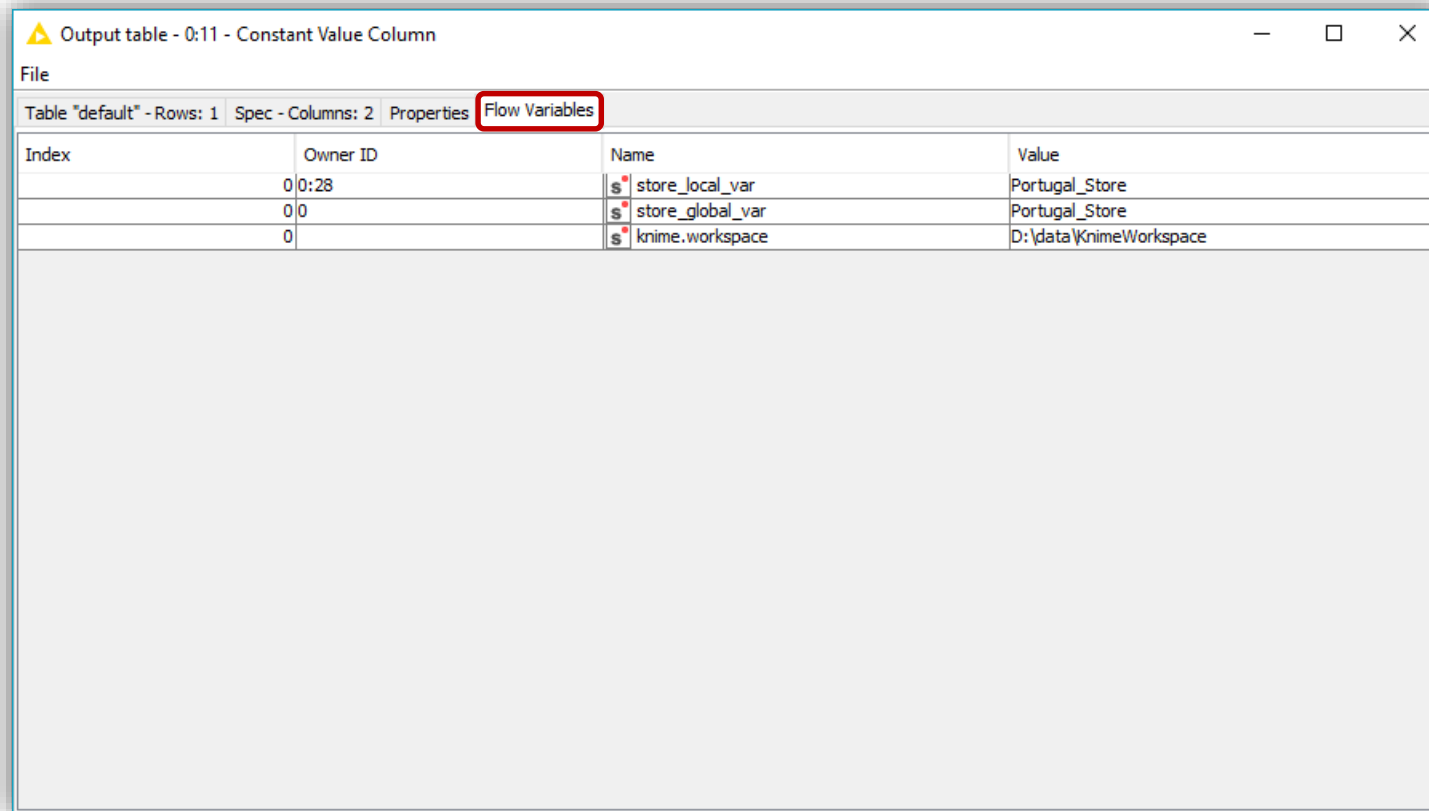
Metanodes

Data Viz

PMML

Hands On




Last but not the least, we can check which **flow variables a node sees** (and may use) in the **table view** of **the node**!



Output table - 0:11 - Constant Value Column

File

Table "default" - Rows: 1 Spec - Columns: 2 Properties **Flow Variables**

Index	Owner ID	Name	Value
0	0:28	 store_local_var	Portugal_Store
	0	 store_global_var	Portugal_Store
	0	 knime.workspace	D:\data\KnimeWorkspace

Data Cells into Variables

14

FLOW VARIABLES

Metanodes

Data Viz

PMML

Hands On

These nodes allows us to **convert values from a table into flow variables** with the row IDs or column names as variable names.

The screenshot displays the KNIME Analytics Platform interface. The main workspace shows a workflow with four nodes: 'Table Column to Variable', 'Table Row to Variable' (highlighted with a red border), 'Variable to Table Row', and 'Create File Name'. The left sidebar contains the 'KNIME Explorer' and 'Node Repository' panels. The 'Node Repository' is filtered to show 'Variables' nodes, with 'Table Row to Variable' selected. The right sidebar shows the 'Node Description' for 'Table Row to Variable', which explains that the node uses a single row of a data table to define new flow variables, with column names as variable names and row values as variable assignments. The 'Dialog Options' section includes a 'Fail' option and 'Defaults' for handling missing values.

Table Row to Variable

This node uses a single row of a data table to define new flow variables. The names of the variables are defined by the column names and the variable assignments (i.e. the values) are given by the values in the row. The variables are exposed using a variables out connection.

Dialog Options

Fail

If selected the node fails with an error on Missing Values or empty tables.

Defaults

If selected the node replaces Missing Values or empty table columns by either integer, double, or String values as defined.

Metanode Templates

15

Flow Variables

METANODES

Data Viz

PMML

Hands On

We have seen that a **Metanode** is a node with other nodes inside! We use them for a **Tidy Workflow**!

However...

We can, and should (!), have metanodes promoting **reusability**!! If a metanode holds a self-contained logical block of nodes, it can be reused for the same task in other workflows by defining a **Metanode Template**.

Metanode Templates

16

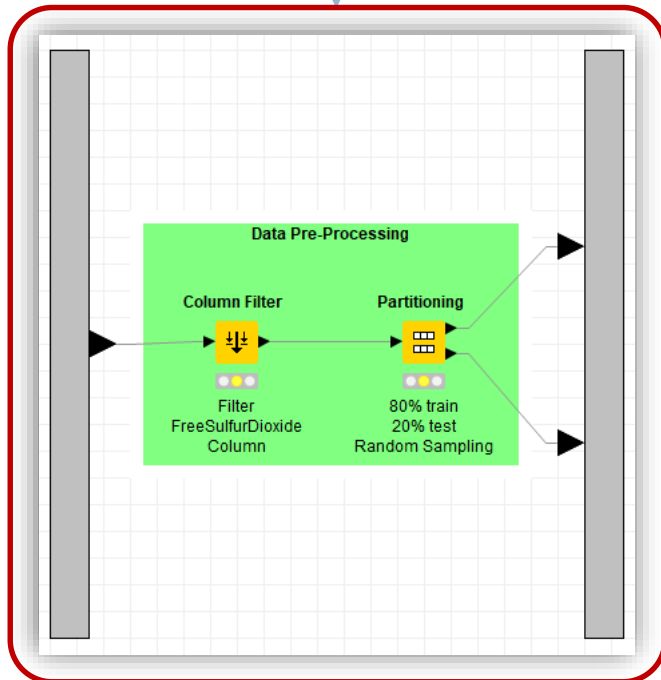
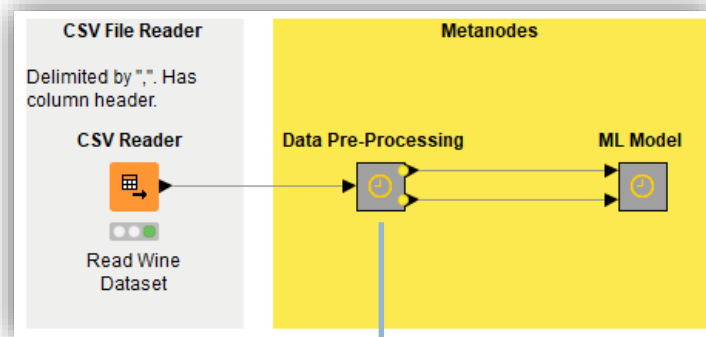
Flow Variables

METANODES

Data Viz

PMML

Hands On



We may want to have this logic replicated in more workflows. Therefore, we should create a **Metanode Template!**

Metanode Templates

17

Flow Variables

METANODES

Data Viz

PMML

Hands On

The screenshot displays a workflow editor interface. On the left, a 'CSV File Reader' node is shown with the description 'Delimited by ";". Has column header.' and a 'CSV Reader' icon. It is connected to a 'Data Pre-Processing' metanode, which is highlighted with a red rectangle. The metanode is part of a yellow 'Metanodes' palette. A context menu is open over the 'Data Pre-Processing' node, listing various actions:

- Configure... (F6)
- Execute (F7)
- Execute and Open Views (Shift+F10)
- Cancel (F9)
- Reset (F8)
- Edit Node Description... (Alt+F2)
- New Workflow Annotation
- Connect selected nodes (Ctrl+L)
- Disconnect selected nodes (Ctrl+Shift+L)
- Create Metanode...
- Create Component...
- Metanode** (highlighted)
- Compare Nodes
- Cut
- Copy
- Paste
- Undo
- Redo
- Delete
- Connected to: First partition (as defined in dialog)
- Connected to: Second partition (remaining rows)

The 'Metanode' option is expanded, showing a sub-menu with the following actions:

- Open
- Expand
- Reconfigure...
- Convert to Component
- Share...** (highlighted)
- Update Link (Ctrl+Alt+U)
- Disconnect Link
- Change Link Type...
- Select in Explorer

Metanode Templates

18

Flow Variables

METANODES

Data Viz

PMML

Hands On

CSV File Reader
Delimited by ";". Has column header.

CSV Reader
Read Wine Dataset

Metanodes
Data Pre-Processing

Save As Metanode Template
Select destination workflow group for metanode template

- LOCAL (Local Workspace)
 - Aulas Ejemplos
 - Aulas Ejercicios**
 - Example Workflows
 - ML Projects

knime://LOCAL/Aulas Ejercicios

Link Shared Metanode
Link local instance to the shared metanode?

Select the type of link to be created:

- ☒ Create absolute link
- ☐ Create mountpoint-relative link
If you move the workflow to a new location it will always link back to this template
- ☐ Create workflow-relative link
- ☐ Don't create link with saved template

OK Cancel

Components (~~Wrapped Metanodes~~)

19

Flow Variables

METANODES

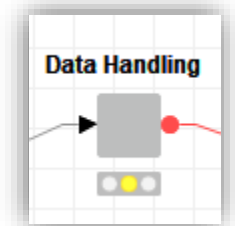
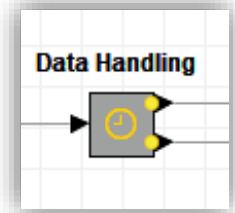
Data Viz

PMML

Hands On

Components are new to Knime and were added in Knime's last major release (4.0.0, June 2019). They were introduced to enhance and replace old Wrapped Metanodes, which have evolved from simple metanodes! Components can:

- Acquire a **configuration window** via Widget nodes
- Have their **own dialog** and their own **interactive views**
- **Represent web pages** in an Analytical Application
- They can **filter flow variables**, i.e., flow variables don't enter or exit a component automatically! We need to allow the flow variable to enter and exit the component!



Components

20

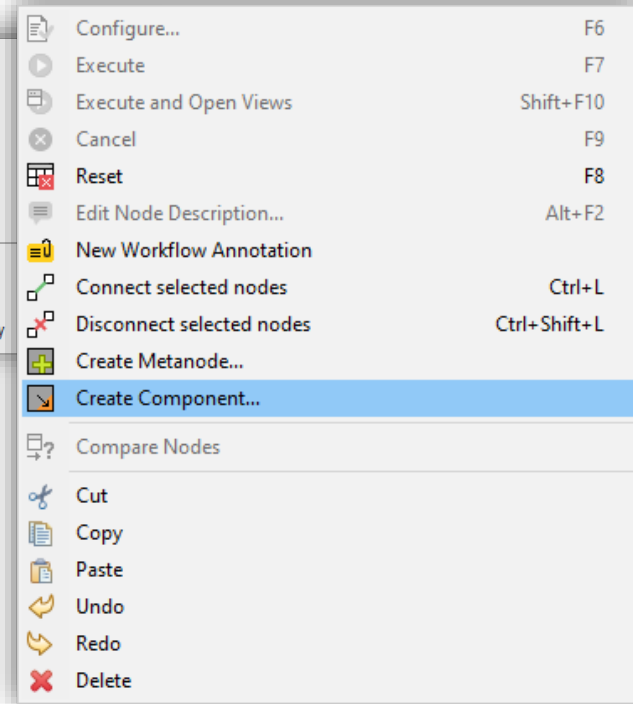
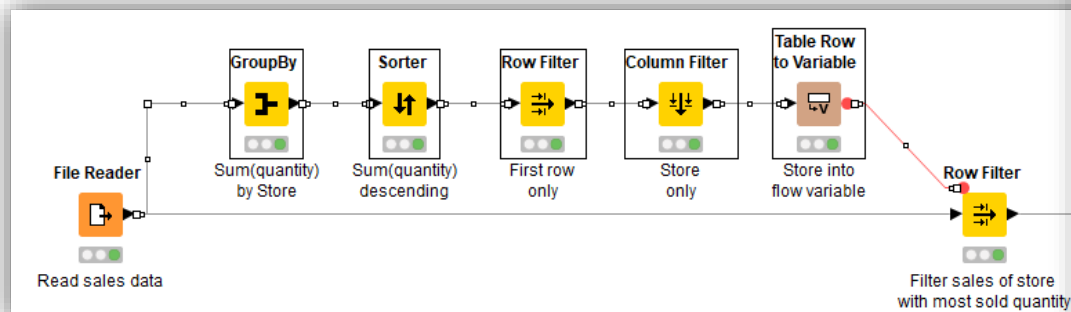
Flow Variables

METANODES

Data Viz

PMML

Hands On



Data Viz component will not execute... Why?

Components

21

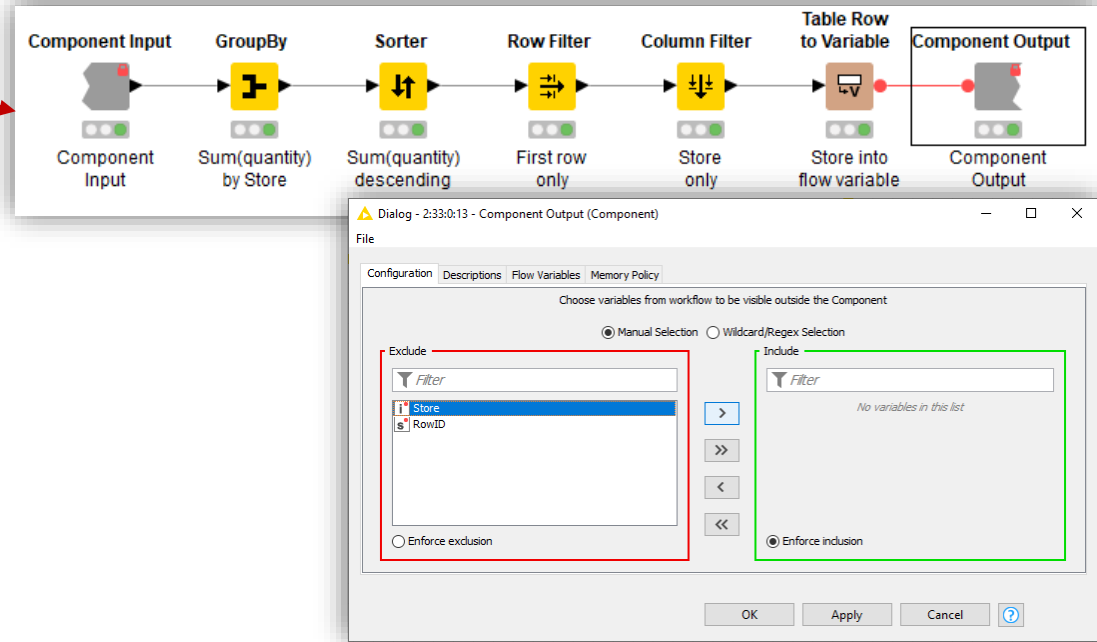
Flow Variables

METANODES

Data Viz

PMML

Hands On



We have to **allow the flow variable** created in the Data Handling component to **exit such component** and then allow it to **enter the Data Viz one**.

Components

22

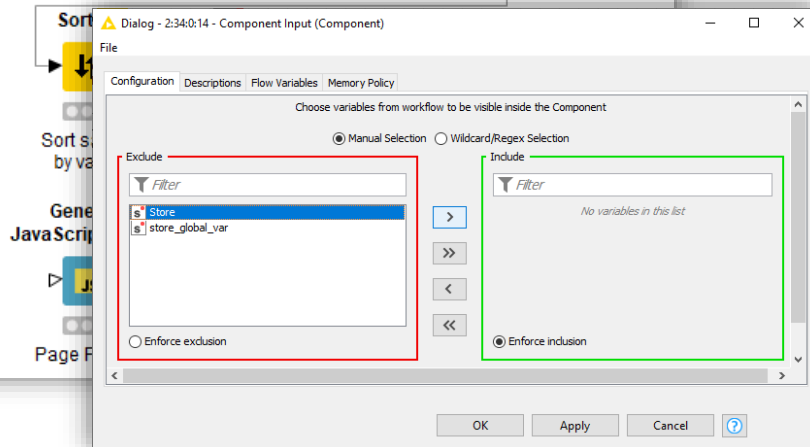
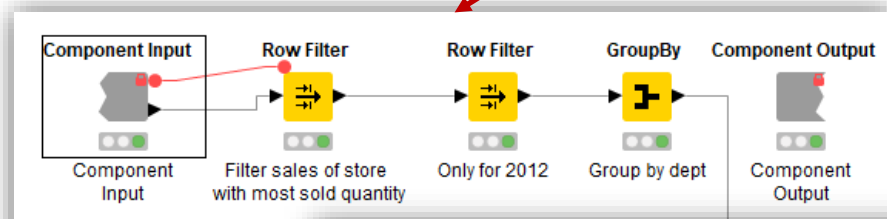
Flow Variables

METANODES

Data Viz

PMML

Hands On



We have to **allow the flow variable** created in the Data Handling component to **exit such component** and then allow it to **enter the Data Viz one**.

Components

23

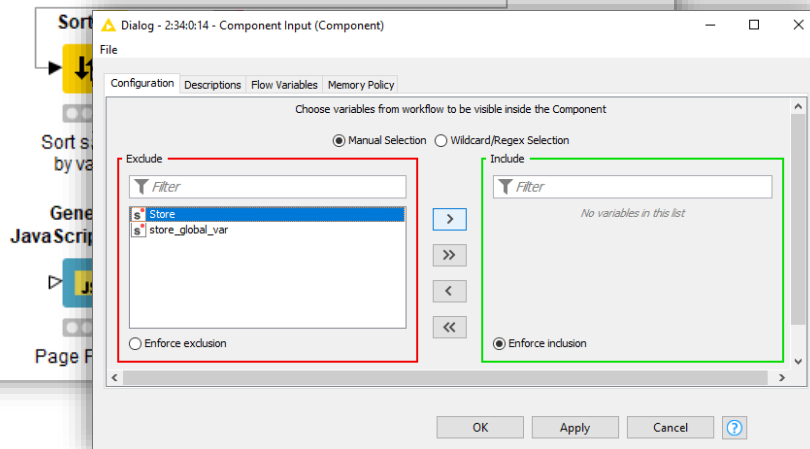
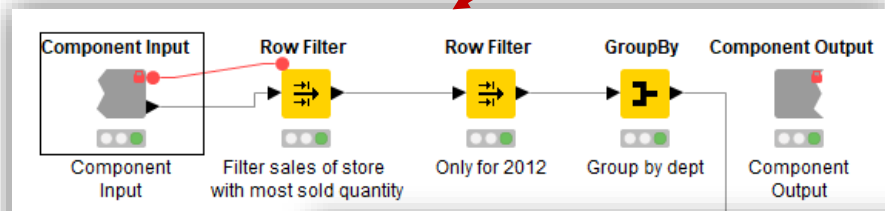
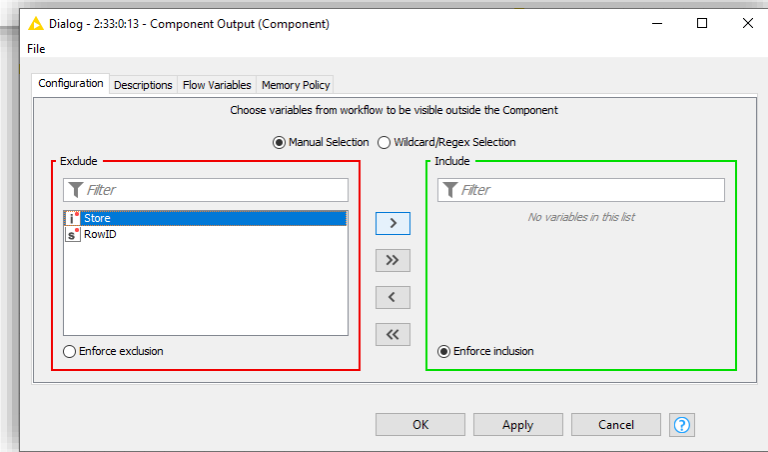
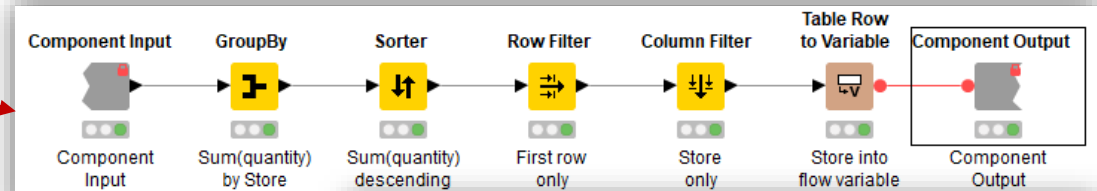
Flow Variables

METANODES

Data Viz

PMML

Hands On



We have to allow the flow variable created in the Data Handling component to exit such component and then allow it to enter the Data Viz one.

Components and Data Viz

24

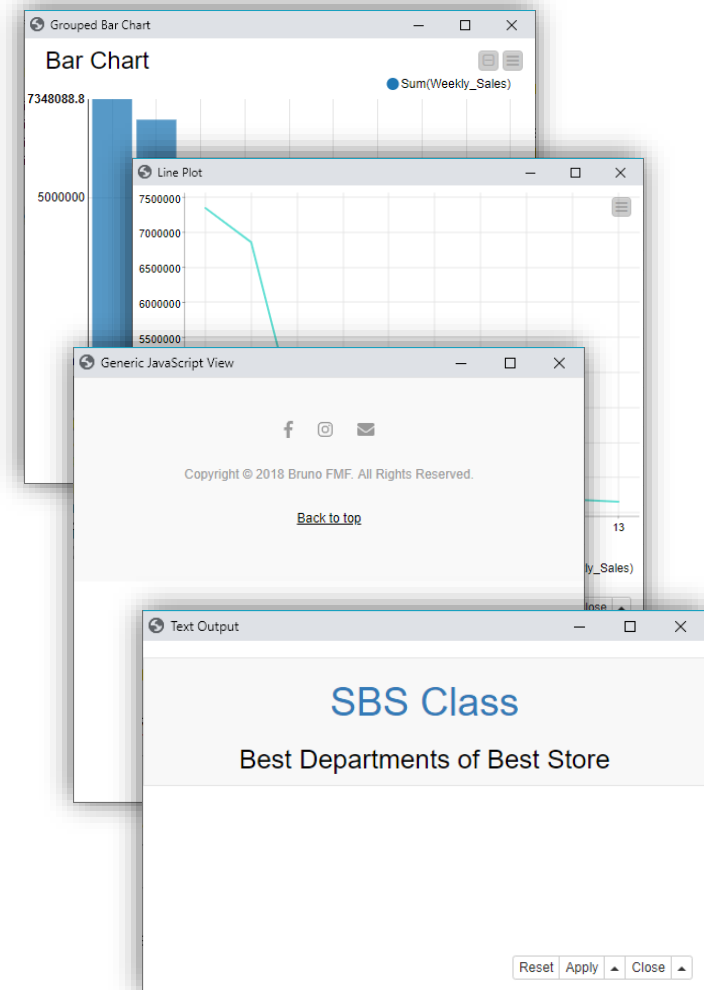
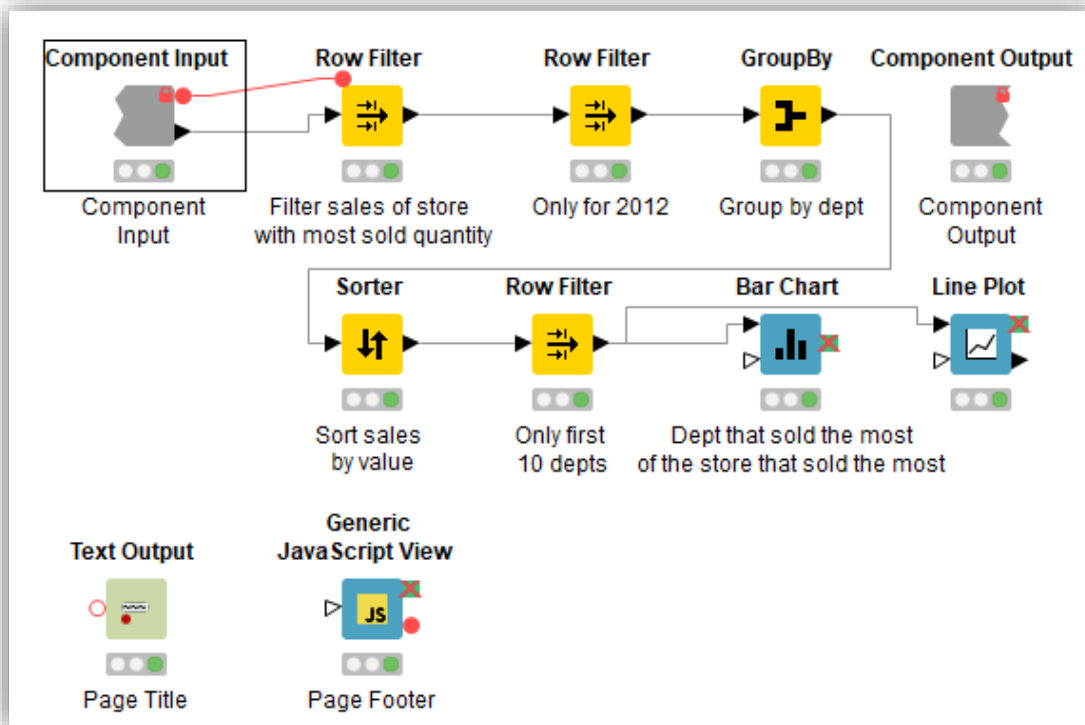
Flow Variables

Metanodes

DATA VIZ

PMML

Hands On



Each plot, chart and javascript widget has its own view.
However...

Components and Data Viz

25

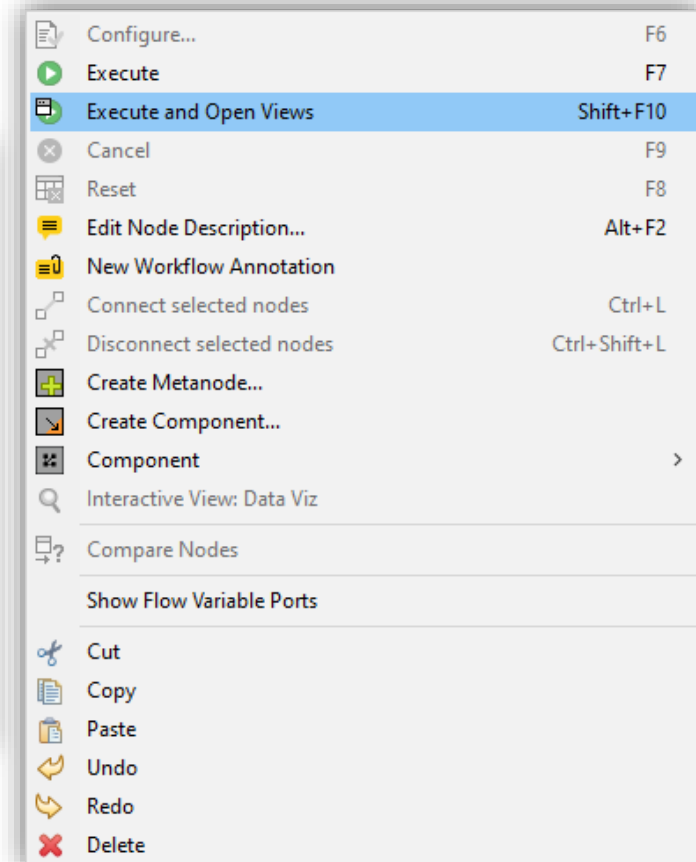
Flow Variables

Metanodes

DATA VIZ

PMML

Hands On



Wrapping the plots/charts/widgets into a Component and executing it...

Components and Data Viz

26

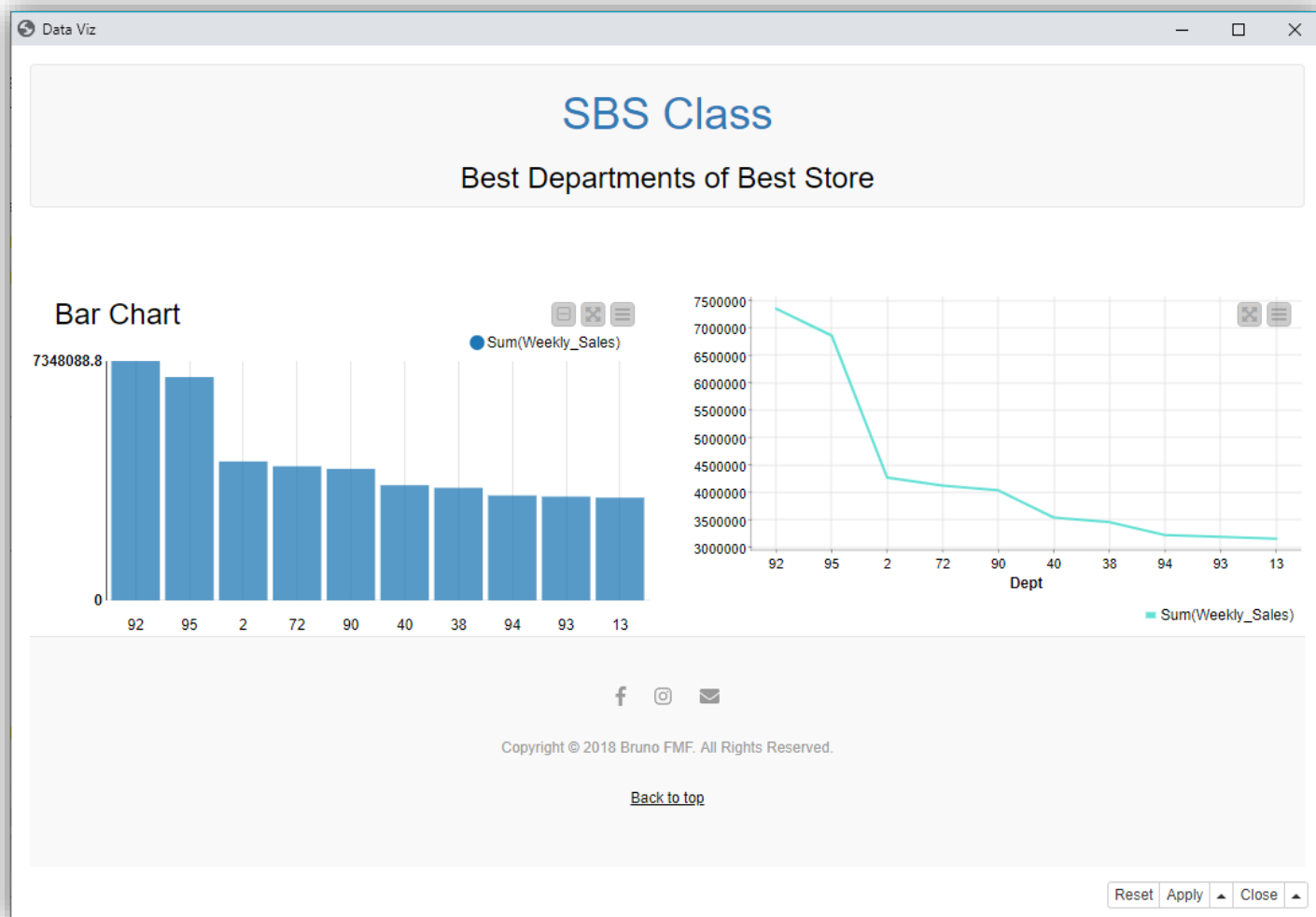
Flow Variables

Metanodes

DATA VIZ

PMML

Hands On



JavaScript Integration



27

Flow Variables

Metanodes

DATA VIZ

PMML

Hands On

JavaScript Charts

Table View
(JavaScript)



Scatter Plot
(JavaScript)



Pie/Donut Chart
(JavaScript)



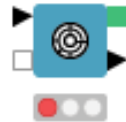
Bar Chart
(JavaScript)



Decision Tree
View (JavaScript)



Sunburst Chart
(JavaScript)



JavaScript Data Handling

String Widget



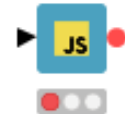
Integer Widget



Slider Widget



Autocomplete
Text Widget



Date&Time Widget



Double Widget



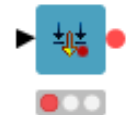
Credentials Widget



Generic
JavaScript View



Column Selection
Widget



Multiple
Selection Widget



Single Selection
Widget



JavaScript Integration



28

Flow Variables

Metanodes

DATA VIZ

PMML

Hands On

Header & Footer

Text Output



Title

Generic JavaScript View



Footer

Dialog - 0:142:0:197 - Text Output (Title)

File

Control Flow Variables Memory Policy

Label: Page Title

Description: Enter Description

Text format: Html

Text:

knime.workspace

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com<br><link rel="stylesheet" href="https://cdnjs.cloudflare.com<br><br><nav class="navbar navbar-default navbar-expand-md fixed-  
<h1 style="text-align: center; color:8FDEFF;">  
  <a href="#">SBS Recommender System</a>  
</h1>  
<h3 style="text-align: center;">Clustering Movies</h3>  
</nav>
```

OK Apply Cancel ?

JavaScript Integration



29

Flow Variables

Metanodes

DATA VIZ

PMML

Hands On

Header & Footer

Text Output



Title

Generic
JavaScript View



Footer

Dialog - 0:142:0:179 - Generic JavaScript View (Footer)

File

JavaScript View | Image Generation | Flow Variables | Memory Policy

Maximum number of rows: 2.500

Flow Variables

- knime.workspace

Dependencies

- ☐ D3 - Version 4.2.6
- ☐ JSFreeChart - Version 0.5.0
- ☐ jQuery - Version 3.1.1
- ☐ jQuery UI - Version 1.12.1

CSS

```
18 /*
19  * Footer
20  */
21 .my-footer {
22   font-family: 'Raleway', Helvetica, Arial, sans-
23   padding: 40px 0;
24   color: #999;
25   text-align: center;
26   font-size: 12px;
27   background-color: #f9f9f9;
28   border-top: 1px solid #e5e5e5;
29 }
30
31 .my-footer .socialMedia {
32   padding: 10px;
33   color: #999;
34   transition: all .5s ease;
35 }
```

JavaScript

```
1 var body = document.getElementsByTagName('body')[
2 var html = '';
3
4 //Load External Scripts - Font Awesome
5 var external_script = document.createElement('scr
6 external_script.setAttribute('src', 'https://use.f
7 document.head.appendChild(external_script);
8
9 //knimeDataTable.getNumRows() or knimeDataTable.g
10 html += '<footer class="my-footer"> <div> <
11
12 body.innerHTML = html;
```

OK Apply Cancel ?

Composite Views



30

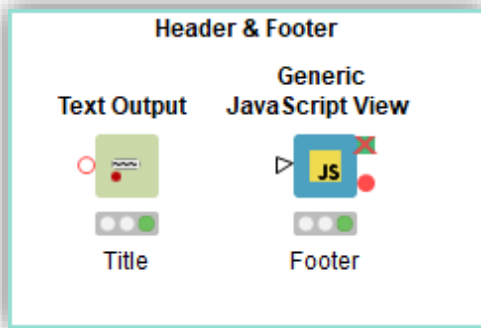
Flow Variables

Metanodes

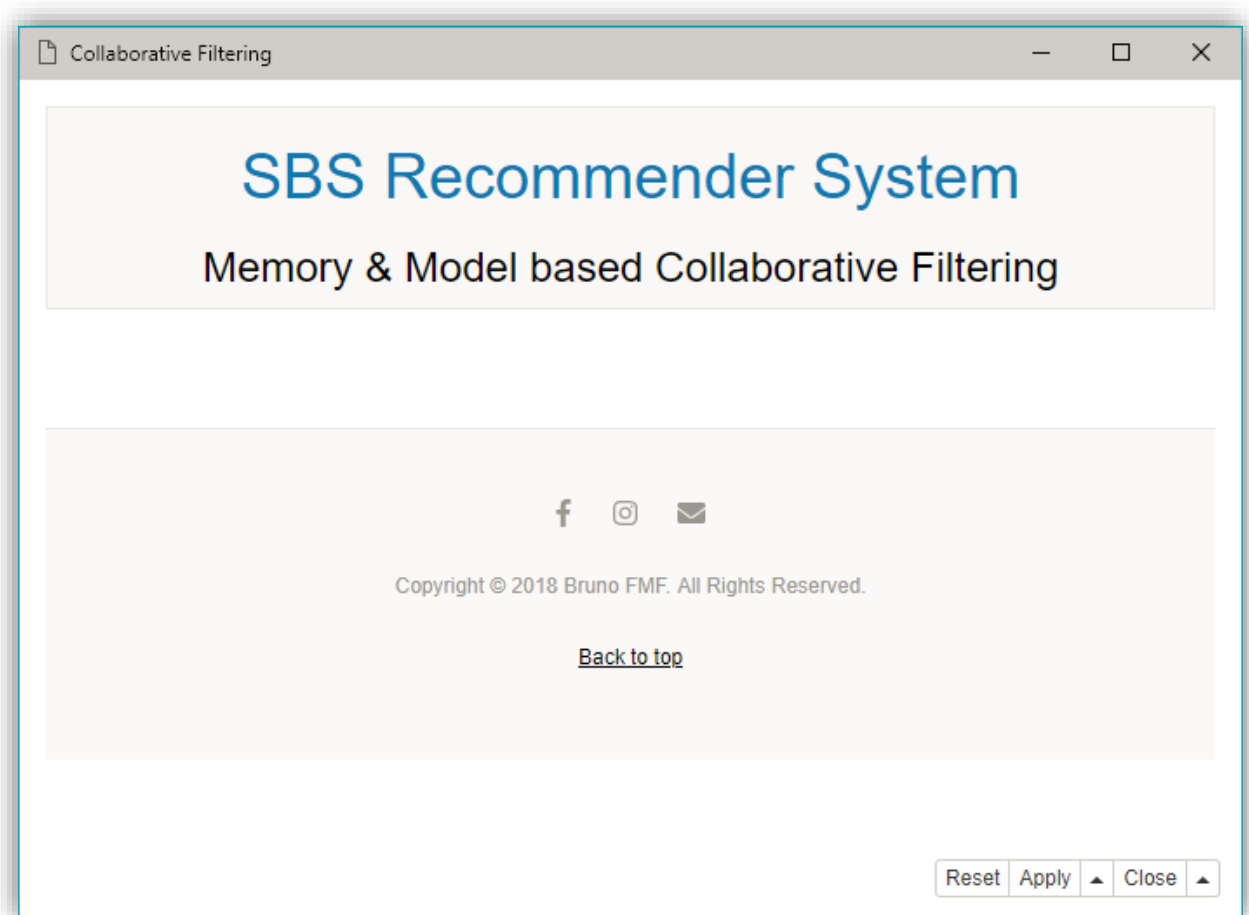
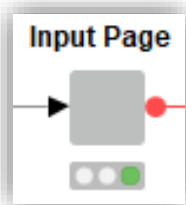
DATA VIZ

PMML

Hands On



Encapsulated inside a
Component



Composite Views



31

Flow Variables

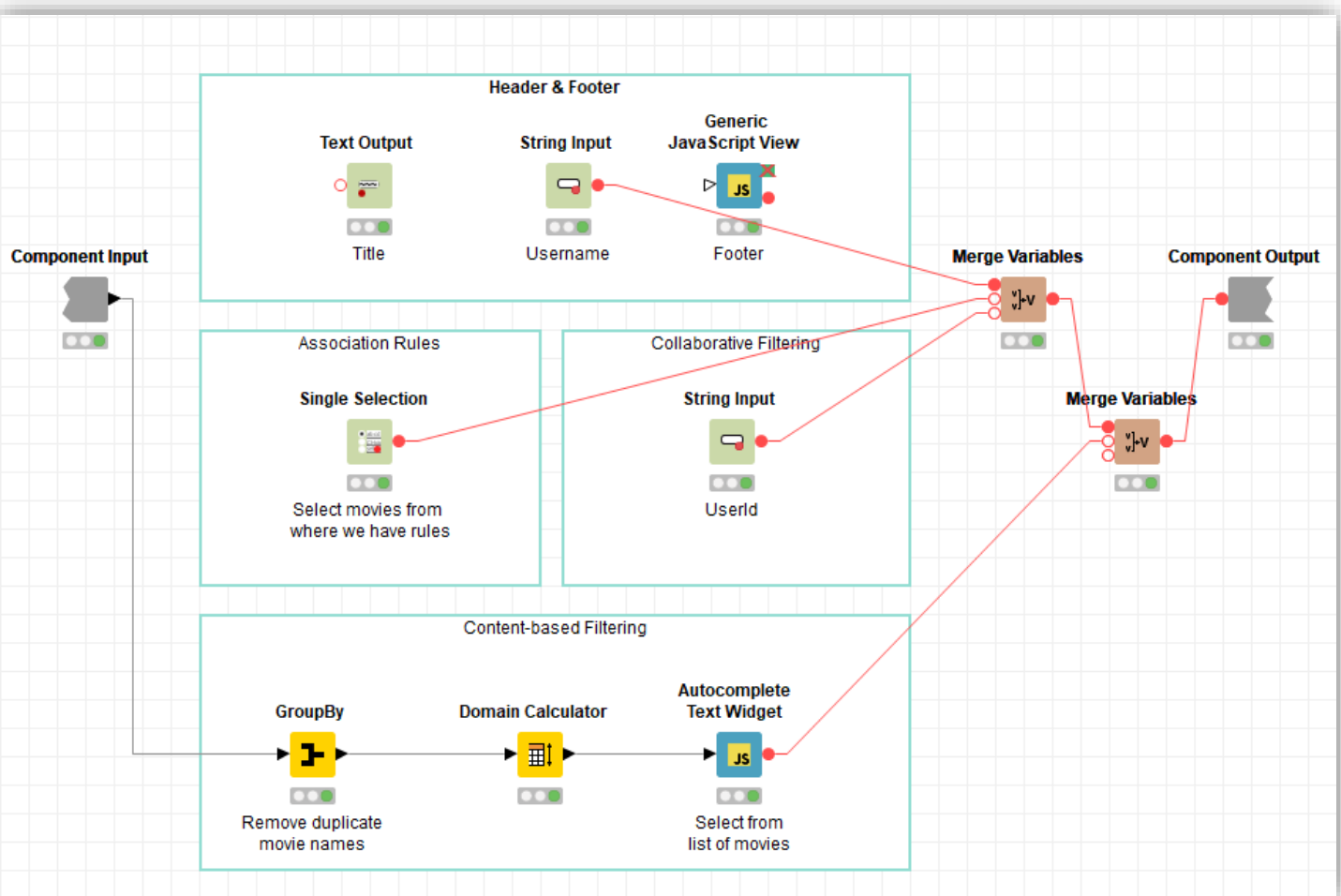
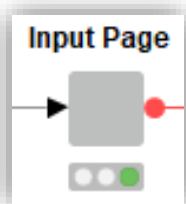
Metanodes

DATA VIZ

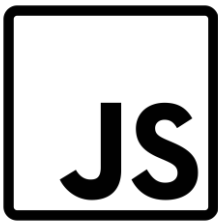
PMML

Hands On

Use JS nodes inside
Components!



Composite Views



32

Flow Variables

Metanodes

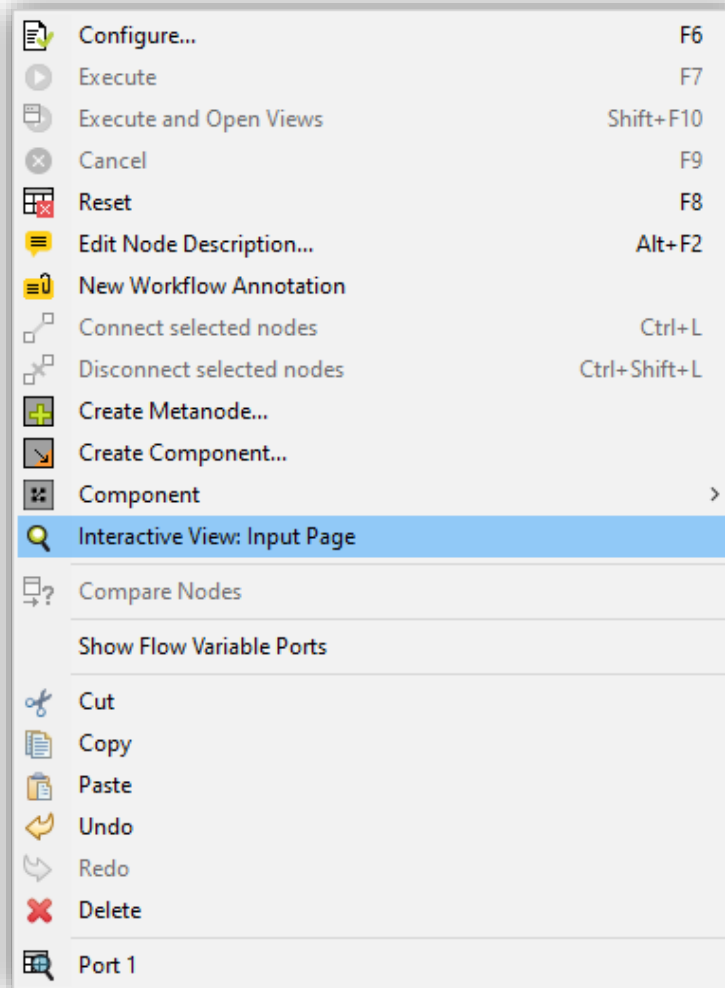
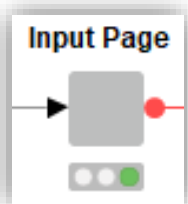
DATA VIZ

PMML

Hands On

Use JS nodes inside

Components!



Composite Views



33

Flow Variables

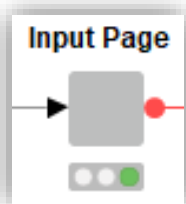
Metanodes

DATA VIZ

PMML

Hands On

Use JS nodes inside
Components!



Input Page

SBS Recommender System

Welcome! Define your data for an accurate recommendation!

Input your User Id

Or username

Select a seen movie for a recommendation!

- ☐ Forrest Gump (1994)
- ☐ Braveheart (1995)
- ☐ Star Wars: Episode V - The Empire Strikes Back (1980)
- ☐ Star Wars: Episode IV - A New Hope (1977)
- ☐ Matrix, The (1999)
- ☐ Silence of the Lambs, The (1991)
- ☒ Pulp Fiction (1994)
- ☐ Shawshank Redemption, The (1994)
- ☐ Jurassic Park (1993)

Search for a movie:

[Back to top](#)

Copyright © 2018 Bruno FMF. All Rights Reserved.

Reset Apply Close

Composite Views

Define the order of the elements inside the page

34

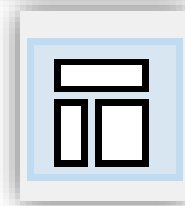
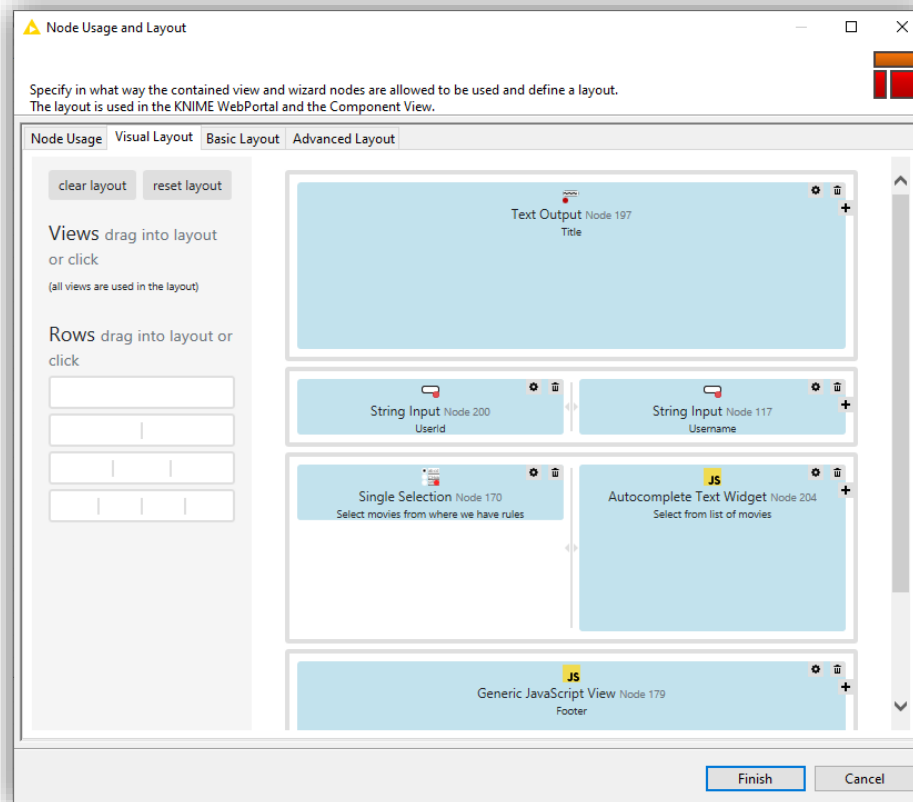
Flow Variables

Metanodes

DATA VIZ

PMML

Hands On



Composite Views

Define the order of the elements inside the page

35

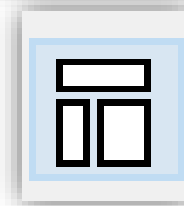
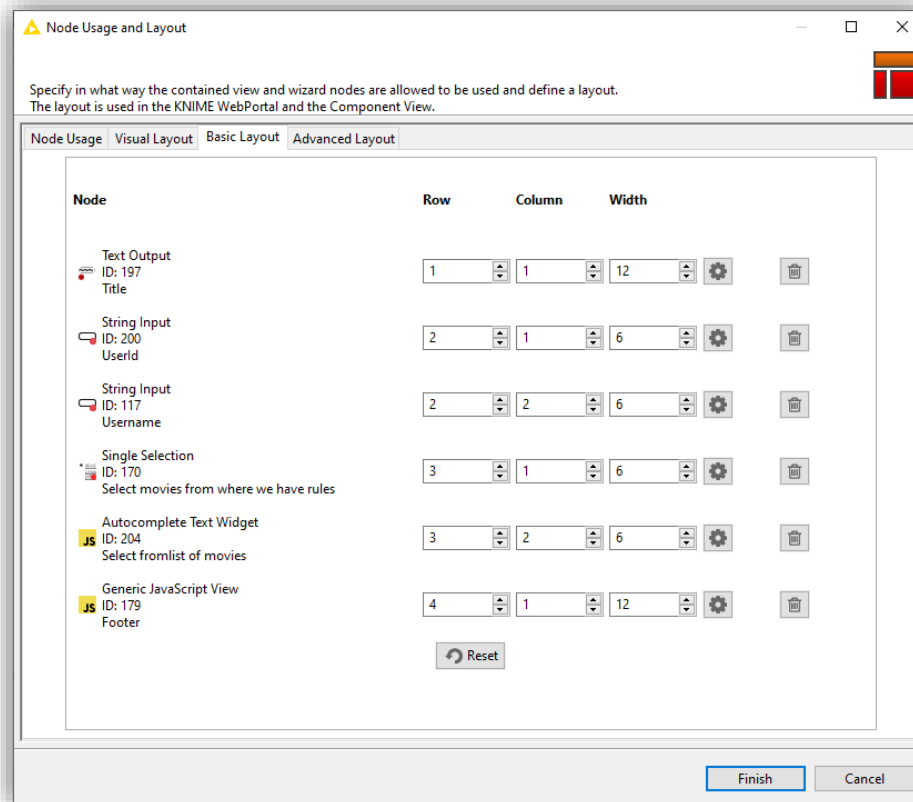
Flow Variables

Metanodes

DATA VIZ

PMML

Hands On



Java Integration

36

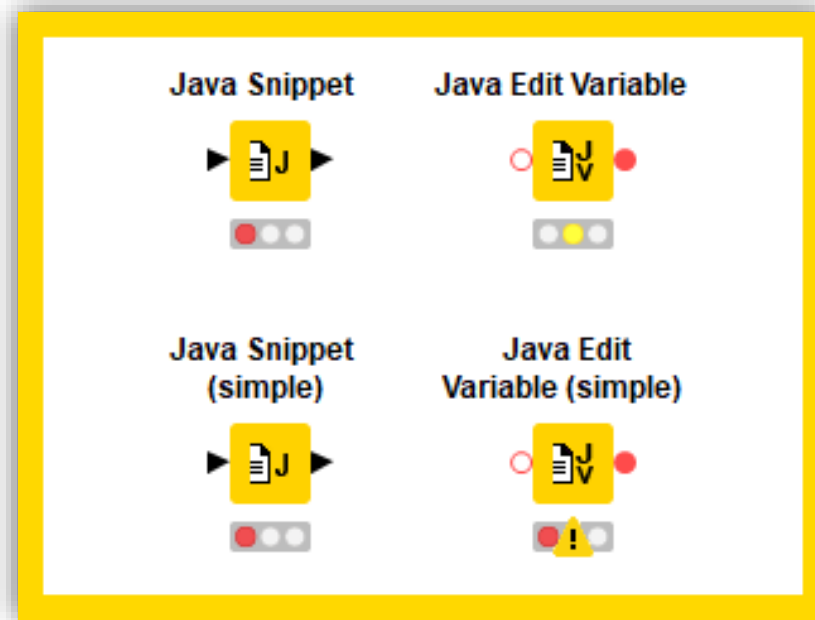
Flow Variables

Metanodes

DATA VIZ

PMML

Hands On



Java Integration

37

Flow Variables

Metanodes

DATA VIZ

PMML

Hands On

Java Snippet (simple)



Dialog - 0:145:102 - Java Snippet (simple)

File

Java Snippet Additional Libraries Flow Variables Memory Policy

Column List

- ROWID
- ROWINDEX
- ROWCOUNT
- I movieId
- S title
- S genres
- I Adventure
- I Action

Flow Variable List

- s knime.workspace

Global Variable Declaration

Method Body

Replace or append result

☒ Append Column: new column

☐ Replace Column: I Other

☐ Insert Missing As Null ☒ Compile on close

Return type

☐ Integer ☐ Long

☒ Double ☐ String

☐ Array Return

OK Apply Cancel ?

Java Integration



38

Flow Variables

Metanodes

DATA VIZ

PMML

Hands On



Java Snippet



Dialog - 3:145:99 - Java Snippet (Parse genre)

File

Java Snippet Additional Libraries Additional Bundles Templates Flow Variables Memory Policy

Column List

- ROWID
- ROWINDEX
- ROWCOUNT
- I movieId
- S title
- S genres

Flow Variable List

- S knime.workspace

```
1 // system imports
13 // Your custom imports:
14
15 // system variables
37 // Your custom variables:
38 int adventure, action, animation, comedy, drama, crime, other;
39
40 // expression start
42 // Enter your code here:
43 adventure = action = animation = comedy = drama = crime = other = 0;
44
45 String[] parts = c_genres.split("\\|");
46 for(int i=0; i<parts.length; i++){
47     switch(parts[i]){
48         case "Adventure":
49             adventure = 1;
50             break;
51         case "Action":
52             action = 1;
53             break;
54         case "Animation":
55             animation = 1;
56             break;
57         case "Comedy":
58             comedy = 1;
```

Input

Name	Java Type	Java Field
S genres	String	c_genres

Output

Field Type	Replace	Name	KNIME Type	Array	Java Type	Java Field
Column	<input type="checkbox"/>	Adventure	I Number (integer)	<input type="checkbox"/>	Integer	out_Adventure
Column	<input type="checkbox"/>	Action	I Number (integer)	<input type="checkbox"/>	Integer	out_Action

OK Apply Cancel ?

Java Integration



39

Flow Variables

Metanodes

DATA VIZ

PMML

Hands On



Java Snippet



Dialog - 3:145:99 - Java Snippet (Parse genre)

File

Java Snippet Additional Libraries Additional Bundles Templates Flow Variables Memory Policy

Column List

- ROWID
- ROWINDEX
- ROWCOUNT
- I movieId
- S title
- S genres

Flow Variable List

- S knime.workspace

```
55         animation = 1;
56         break;
57         case "Comedy":
58             comedy = 1;
59             break;
60         case "Drama":
61             drama = 1;
62             break;
63         case "Crime":
64             crime = 1;
65             break;
66         default:
67             other = 1;
68     }
69 }
70
71 out_Adventure = adventure;
72 out_Action = action;
73 out_Animation = animation;
74 out_Comedy = comedy;
75 out_Drama = drama;
76 out_Crime = crime;
77 out_Other = other;
78 // expression end
81
```

Input

Name	Java Type	Java Field
S genres	String	c_genres

Output

Field Type	Replace	Name	KNIME Type	Array	Java Type	Java Field
Column	<input type="checkbox"/>	Adventure	I Number (integer)	<input type="checkbox"/>	Integer	out_Adventure
Column	<input type="checkbox"/>	Action	I Number (integer)	<input type="checkbox"/>	Integer	out_Action

OK Apply Cancel ?

Java Integration

40

Flow Variables

Metanodes

DATA VIZ

PMML

Hands On



Java Snippet



Appended table - 3:145:99 - Java Snippet (Parse genre)

File Hilite Navigation View

Table "default" - Rows: 9742 Spec - Columns: 10 Properties Flow Variables

Row ID	movieId	title	genres	Advent...	Action	Animation	Comedy	Drama
Row0	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	1	0	1	1	0
Row1	2	Jumanji (1995)	Adventure Children Fantasy	1	0	0	0	0
Row2	3	Grumpier Old M...	Comedy Romance	0	0	0	1	0
Row3	4	Waiting to Exh...	Comedy Drama Romance	0	0	0	1	1
Row4	5	Father of the B...	Comedy	0	0	0	1	0
Row5	6	Heat (1995)	Action Crime Thriller	0	1	0	0	0
Row6	7	Sabrina (1995)	Comedy Romance	0	0	0	1	0
Row7	8	Tom and Huck (...)	Adventure Children	1	0	0	0	0
Row8	9	Sudden Death ...	Action	0	1	0	0	0
Row9	10	GoldenEye (1995)	Action Adventure Thriller	1	1	0	0	0
Row10	11	American Presi...	Comedy Drama Romance	0	0	0	1	1
Row11	12	Dracula: Dead ...	Comedy Horror	0	0	0	1	0
Row12	13	Balto (1995)	Adventure Animation Children	1	0	1	0	0
Row13	14	Nixon (1995)	Drama	0	0	0	0	1
Row14	15	Cutthroat Islan...	Action Adventure Romance	1	1	0	0	0
Row15	16	Casino (1995)	Crime Drama	0	0	0	0	1
Row16	17	Sense and Sen...	Drama Romance	0	0	0	0	1
Row17	18	Four Rooms (1...	Comedy	0	0	0	1	0

PMML

41

Flow Variables

Metanodes

Data Viz

PMML

Hands On

PMML stands for **Predictive Model Markup Language**

- It is the **leading standard** on Data Mining and **Machine Learning models representation**
- Being developed by the Data Mining Group (DMG)
- It is a XML-based format for **interchanging predictive models** among different applications
- Enables the instant deployment of predictive solutions
- The most popular software tools for ML already contain features to import/export PMML models

PMML

42

Flow Variables

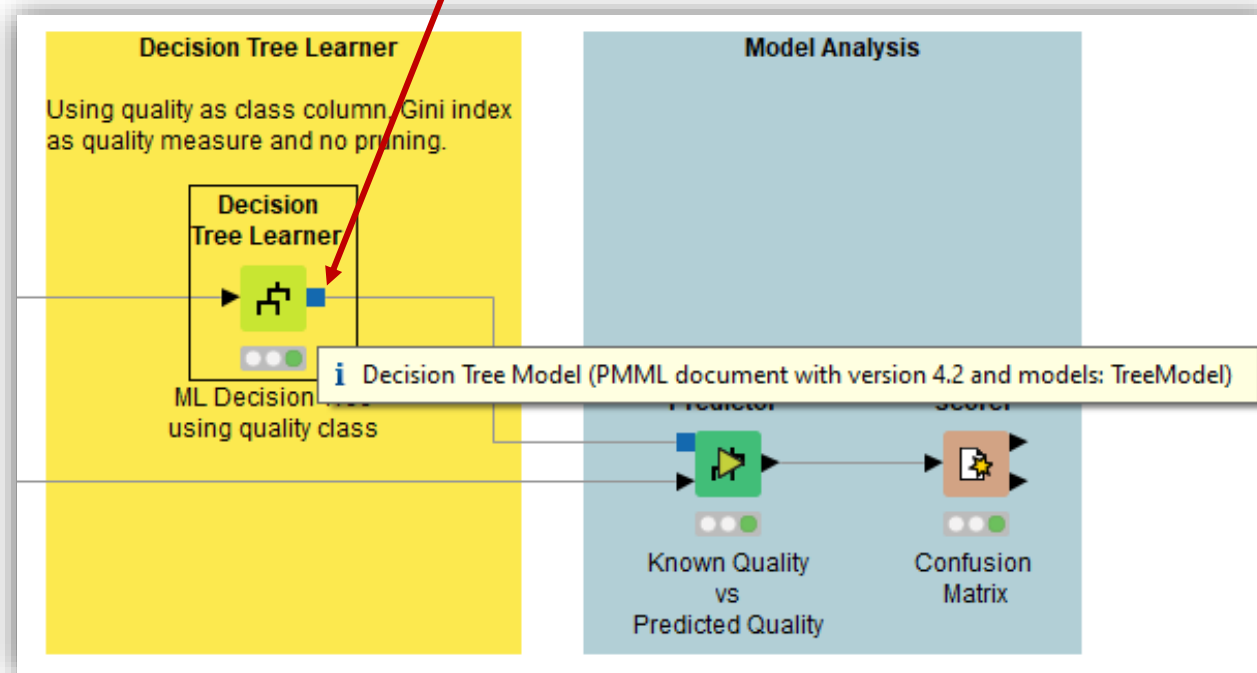
Metanodes

Data Viz

PMML

Hands On

Most Knime ML nodes export a **PMML model** (you can save it to the file system using the **PMML Writer node**)



PMML Structure

43

Flow Variables

Metanodes

Data Viz

PMML

Hands On

Header

Information about who created the PMML, the used software, copyright information and a timestamp

Data Dictionary

Information about the variables in the dataset: their name, type and range

Transformation Dictionary

Includes the pre and post-processing actions to be applied within the predictive model such as normalization, discretization, value mapping, text indexing, functions and/or aggregation

Predictive Model

Represents the predictive model itself. For example, a Multi-Layered Feedforward Neural Network is represented by a *NeuralNetwork* element containing attributes such as *Model Name*, *Function Name*, *Algorithm Name* and *Number of Layers*, followed by elements such as *Neural Inputs*, *Neural Layer* and *Neural Outputs*

PMML Structure

44

Flow Variables

Metanodes

Data Viz

PMML

Hands On

```
<?xml version="1.0" encoding="UTF-8"?>
<PMML version="4.2" xmlns="http://www.dmg.org/PMML-4_2">
  <Header copyright="user">
    <Application name="KNIME" version="3.6.1"/>
  </Header>
  <DataDictionary numberOfFields="11">
    <TreeModel modelName="DecisionTree" functionName="classification" splitCharacteristic="binarySplit"
      ="returnLastPrediction">
      <MiningSchema>
        <MiningField name="fixed acidity" invalidValueTreatment="asIs"/>
        <MiningField name="volatile acidity" invalidValueTreatment="asIs"/>
        <MiningField name="citric acid" invalidValueTreatment="asIs"/>
        <MiningField name="residual sugar" invalidValueTreatment="asIs"/>
        <MiningField name="chlorides" invalidValueTreatment="asIs"/>
        <MiningField name="total sulfur dioxide" invalidValueTreatment="asIs"/>
        <MiningField name="density" invalidValueTreatment="asIs"/>
        <MiningField name="pH" invalidValueTreatment="asIs"/>
        <MiningField name="sulphates" invalidValueTreatment="asIs"/>
        <MiningField name="alcohol" invalidValueTreatment="asIs"/>
        <MiningField name="quality" invalidValueTreatment="asIs" usageType="target"/>
      </MiningSchema>
      <Node id="0" score="=5" recordCount="1279.0">
        <True/>
        <ScoreDistribution value="=5" recordCount="556.0"/>
        <ScoreDistribution value="=6" recordCount="499.0"/>
        <ScoreDistribution value="=7" recordCount="164.0"/>
        <ScoreDistribution value="=4" recordCount="40.0"/>
        <ScoreDistribution value="=8" recordCount="15.0"/>
        <ScoreDistribution value="=3" recordCount="5.0"/>
        <Node id="1" score="=5" recordCount="786.0">
          <Node id="116" score="=6" recordCount="493.0">
          </Node>
        </Node>
      </TreeModel>
    </PMML>
```

Hands On

45

Flow Variables

Metanodes

Data Viz

PMML

HANDS ON

The screenshot displays the KNIME Analytics Platform interface. The top menu bar includes File, Edit, View, Node, and Help. Below the menu is a toolbar with various icons for file operations, workflow management, and execution. The main workspace is a large grid where a workflow is built. On the left, the 'KNIME Explorer' pane shows a hierarchical view of the project structure, including 'My-KNIME-Hub', 'EXAMPLES', 'LOCAL (Local Workspace)', and 'KNIME_Aula_Exercicio'. Below this is the 'Workflow Coach' pane, which lists 'Recommended Nodes' such as File Reader, CSV Reader, Excel Reader (XLS), Table Creator, Database Reader (legacy), Table Reader, List Files, and Database Connection T... At the bottom left is the 'Node Repository' pane, which categorizes nodes into IO, Manipulation, Views, Analytics, DB, Other Data Types, Structured Data, Scripting, Tools & Services, Workflow Control, Workflow Abstraction, and Reporting. On the right side, there is a 'Quick Access' pane with a search bar labeled 'Search workflows, nodes, and more...'. The bottom of the interface features a 'Console' and 'Outline' pane. A large, bold, black 'HANDS ON' watermark is diagonally placed across the center of the workspace grid.