Pentaho Data Integration (PDI) - Kettle – short guide

I. Creating new transformation

II. Opening existing template transformations for manual design approach

a. **Template for requirement 1**
b. **Template for requirement 2**

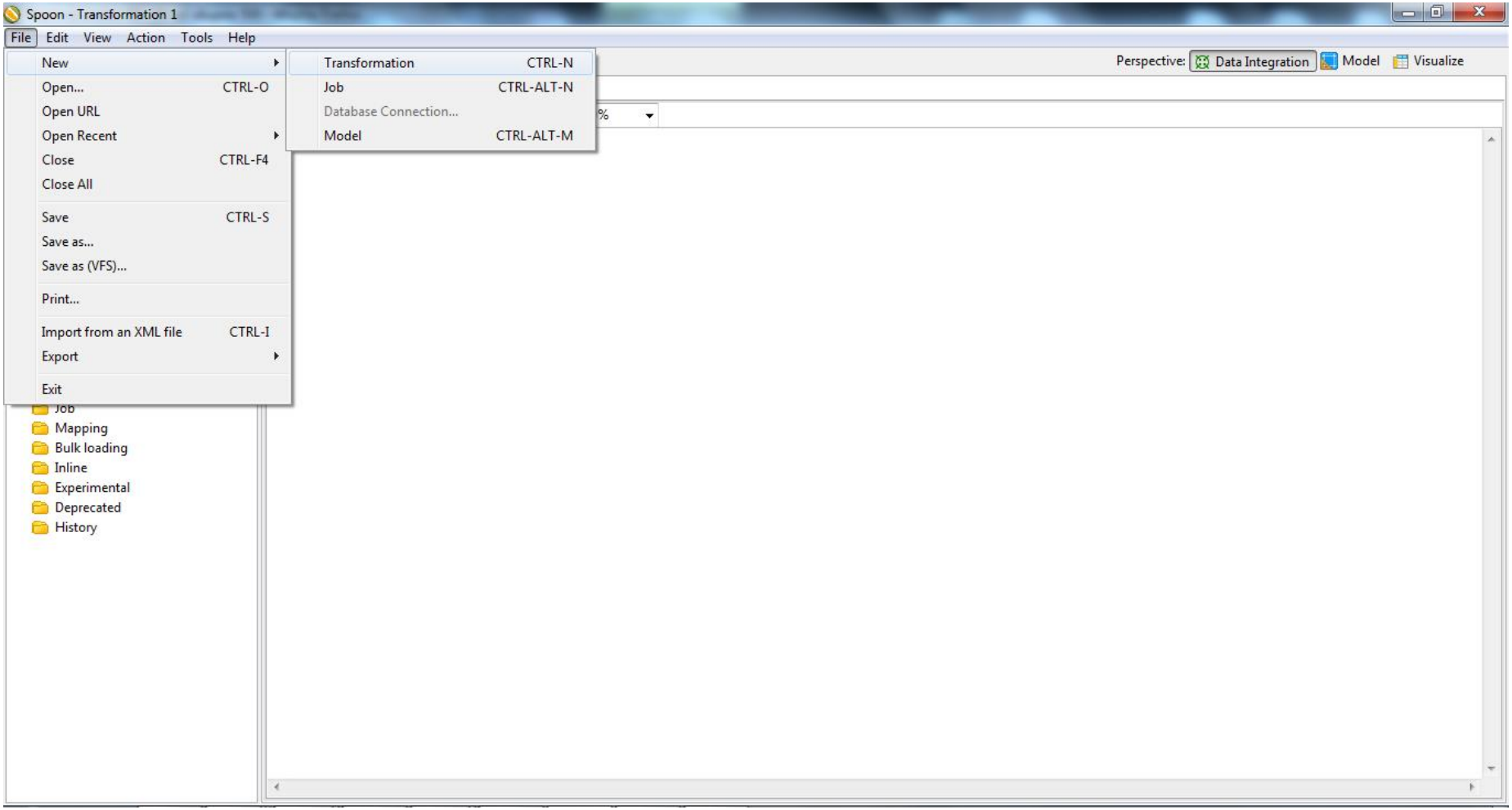
III. Adding operations (steps) into transformation.

a. **Transformation**

- *Sort Rows*
- *Select values*
- *Filter rows*
- *Modified Java Script Value*
- *Join Rows*
- Group By

IV. Creating edges between operations (hops)

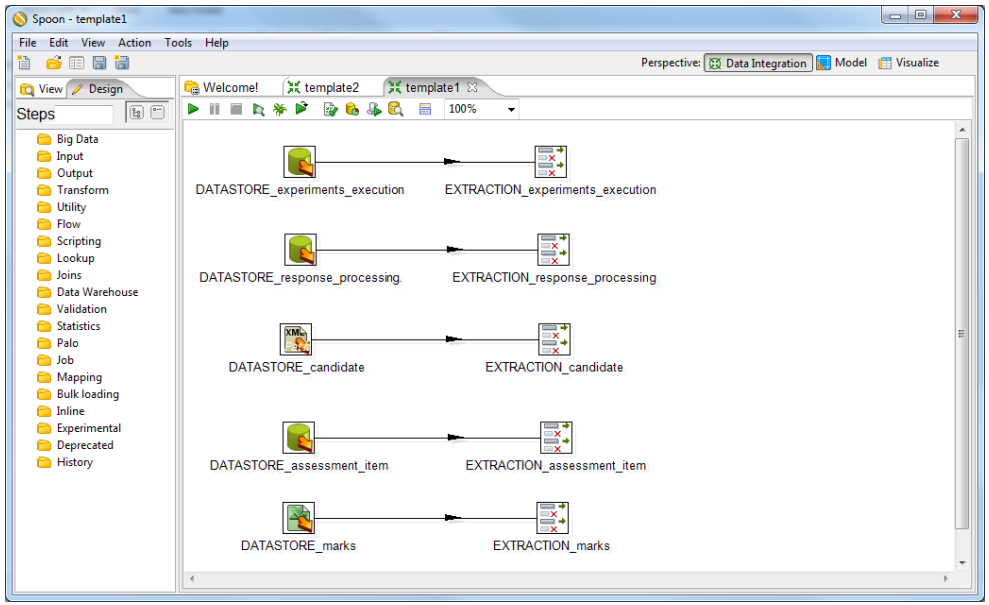
I. Creating new transformation



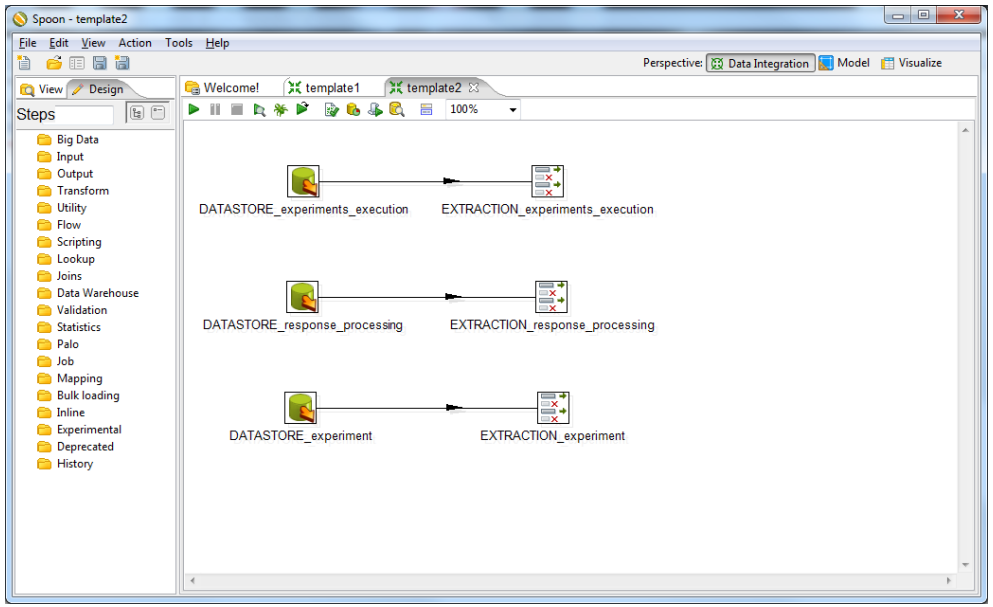
II. Opening existing template transformations for manual design approach

To open existing template transformation (template1.ktr or template2.ktr) choose File -> Open…

a) Template for requirement 1

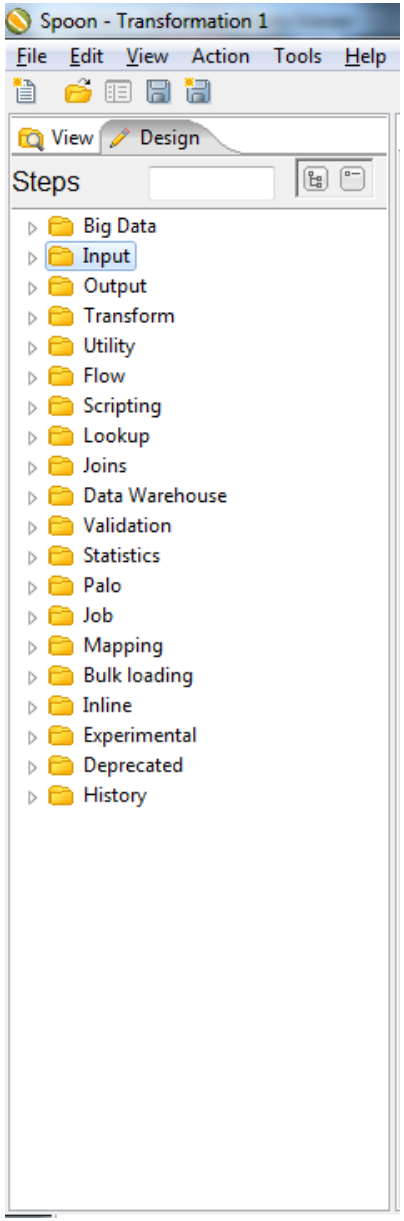


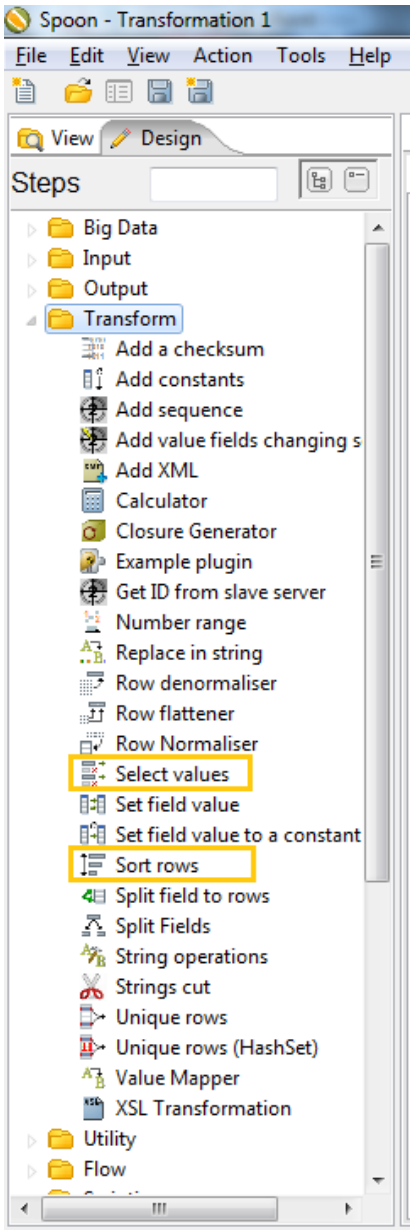
b) Template for requirement 2

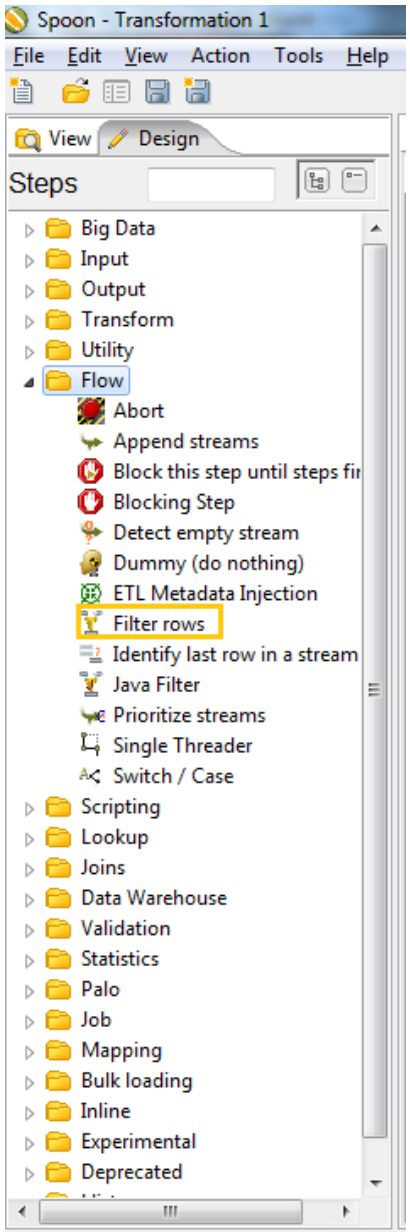


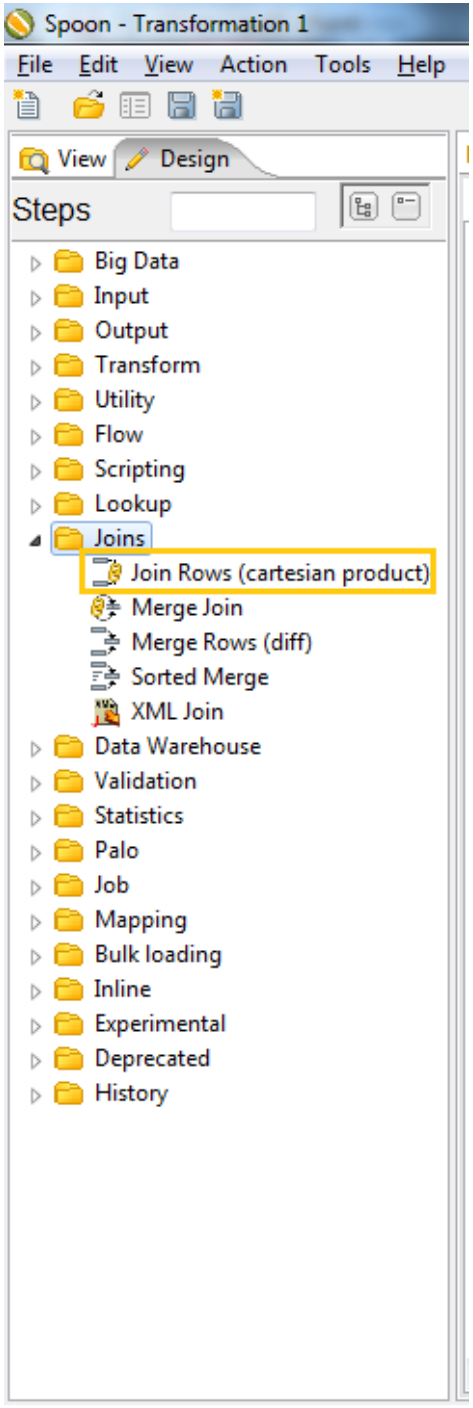
III. Adding operations (steps) into transformation.

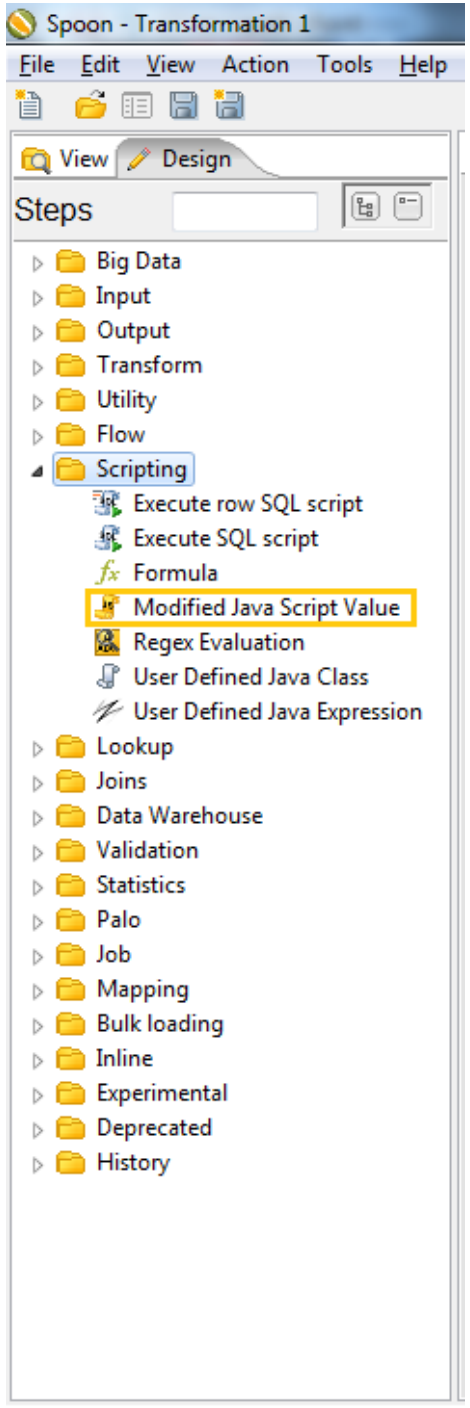
After you open the provided template transformations, in the tab “Design” you may find the complete palette of the operations supported by Kettle. The operations are divided into different groups and they are easily added to an ETL design by dragging them from the palette into the transformation canvas. In the following figures, you may see different operations in the Spoon design palette. The ones that should be used for this round of the case study are marked with orange rectangle and further explained.

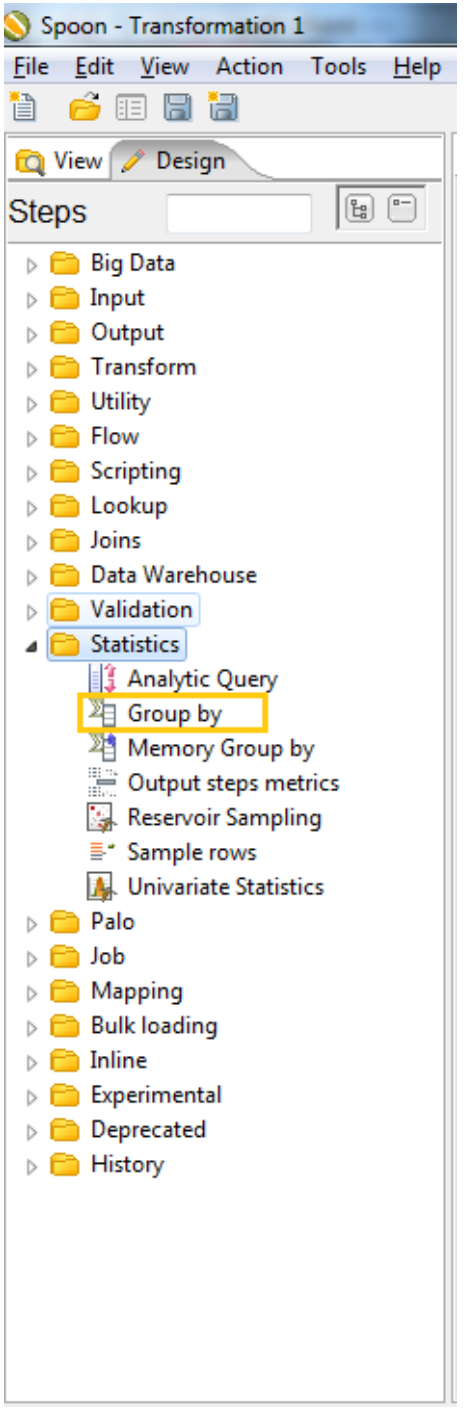







Spoon - Transformation 1
File Edit View Action Tools Help
View Design
Steps
Big Data
Input
Output
Transform
Utility
Flow
Scripting
Lookup
Joins
Join Rows (cartesian product)
Merge Join
Merge Rows (diff)
Sorted Merge
XML Join
Data Warehouse
Validation
Statistics
Palo
Job
Mapping
Bulk loading
Inline
Experimental
Deprecated
History


Spoon - Transformation 1
File Edit View Action Tools Help
View Design
Steps
Big Data
Input
Output
Transform
Utility
Flow
Scripting
Execute row SQL script
Execute SQL script
Formula
Modified Java Script Value
Regex Evaluation
User Defined Java Class
User Defined Java Expression
Lookup
Joins
Data Warehouse
Validation
Statistics
Palo
Job
Mapping
Bulk loading
Inline
Experimental
Deprecated
History


Spoon - Transformation 1
File Edit View Action Tools Help
View Design
Steps
Big Data
Input
Output
Transform
Utility
Flow
Scripting
Lookup
Joins
Data Warehouse
Validation
Statistics
Analytic Query
Group by
Memory Group by
Output steps metrics
Reservoir Sampling
Sample rows
Univariate Statistics
Palo
Job
Mapping
Bulk loading
Inline
Experimental
Deprecated
History

a) Transformation operations (steps)

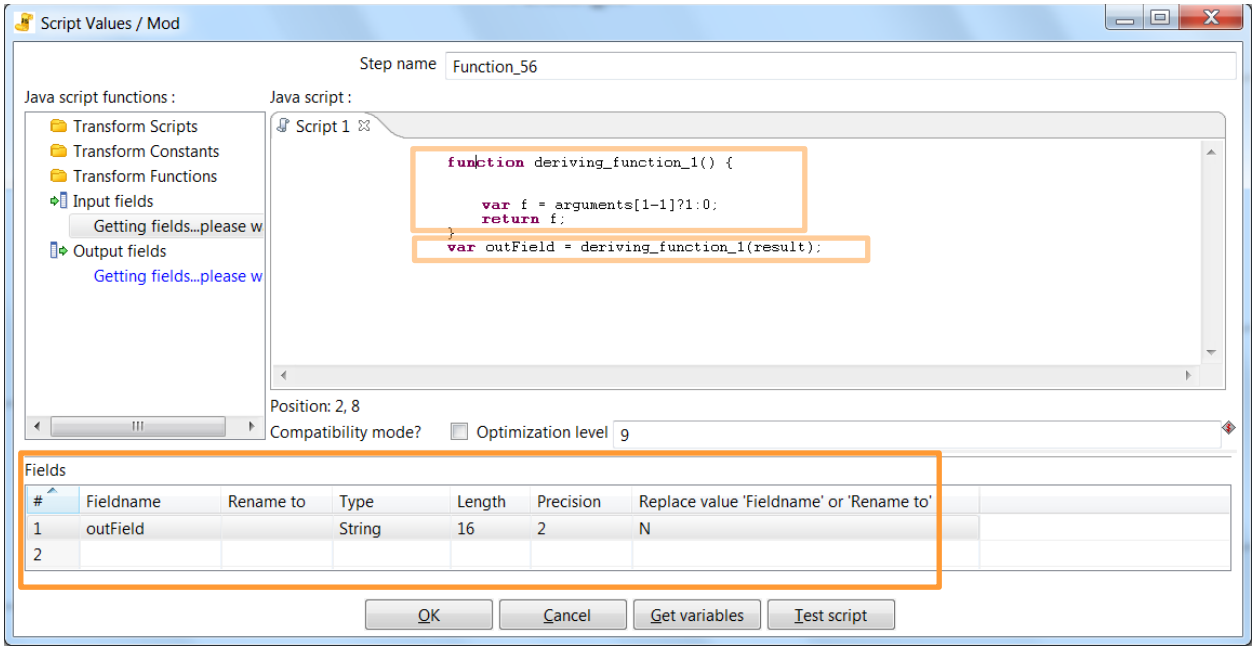
Sort Rows 	*The Sort rows step sorts rows based on the fields you specify and on whether they should be sorted in ascending or descending order. This step is necessary to be used before Group By operation to ensure that the inputs are sorted by the*
Option	**Description**
Step name	Name of the step; This name has to be unique in a single transformation.
Fields table	Specify the fields and direction (ascending/descending) to sort. You can specify whether to perform a case sensitive sort (optional)
Get Fields	Click to retrieve a list of all fields coming in on the stream(s).

Select values 	*This Select values step is useful for selecting (projection), renaming, changing data types and configuring the length and precision of the fields on the stream. These operations are organized into different categories:* • *Select & Alter - Specify the exact order and name in which the fields have to be placed in the output rows* • *Remove - Specify the fields that have to be removed from the output rows* • *Meta-data - Change the name, type, length and precision (the meta-data) of one or more fields*
SELECT & ALTER	
Option	**Description**
Step name	Name of the step; this name has to be unique in a single transformation
Fields	Allows you to rename a field and specify the length and precision
Get fields to select	Click to insert fields from all input steams to the step

Filter rows  Filter rows	*The Filter rows step allows you to filter rows based on conditions and comparisons. Once this step is connected to a previous step (one or more and receiving input), you can click on the "<field>", "=" and "<value>" areas to construct a condition. Click the Add condition icon to add conditions.* *Add condition converts the original condition to a sub-condition and adds a new condition. Click a sub-condition to edit it (go down one level into the condition tree).*
Option	**Description**
Step name	Name of the step; this name has to be unique in a single transformation.
Send 'true' data to step	The rows for which the condition specified is true are sent to this step
Send 'false' data to step	The rows for which the condition specified are false are sent to this step
The Condition	Click the 'NOT' button in the upper left to negate the condition. Click on the <Field> buttons to select from a list of fields from the input stream(s) to build your condition(s). Click on the <value> button to enter a specific value into your condition(s). To delete a condition, right-click and select Delete Condition

Modified Java Script Value  Modified Java Script Value	*Modified Java Script Value type allows you to perform complex calculations using JavaScript.*
Option	**Description**
Java Scripts	This section is where you edit the script for this step. You can insert functions, constants, input fields, etc. from the tree control on the left by double-clicking on the node you wish to insert or by dragging the object onto the Java Script panel.
Fields	The Fields table contains a list of variables from your script including the ability to add metadata like a descriptive name.

For example, you can define a java script function to derive a new value (e.g., derived measure). Function can be parameterized with the input fields of this step. The derived value (measure) need to be defined as the output field of the step in section "Fields". Provided java script needs to have the definition of the value for the output fields. However, you can define an arbitrary java script to calculate desired value from the input fields and assign it to the output field defined in section "Fields".

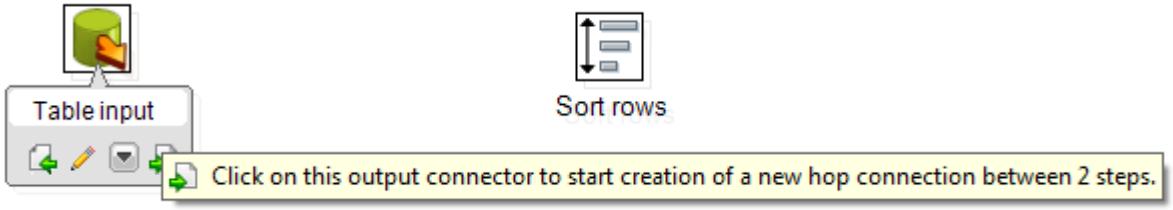


Join Rows  Join Rows (cartesian product)	*The Join rows step allows you to produce combinations (Cartesian product) of all rows in the input streams with additional condition to perform different kinds of joins.*
Option	**Description**
Step name	Name of the step; this name has to be unique in a single transformation
Main step to read from	Specifies the step from which to read most of the data; while the data from other steps are cached or spooled to disk, the data from this step is not.
The Condition(s)	You can enter a complex condition to limit the number of output row. **Note**: The fields in the condition must have unique names in each of the streams.

Group By  Group by	*This step allows you to calculate values over a defined group of fields.* ***Note:*** *This step requires inputs to be ordered by the group by fields (attributes). Sort Rows operation needs to precede it.*
Option	**Description**
Step name	Name of the step; this name has to be unique in a single transformation
Include all rows?	Enable if you want all rows in the output, not just the aggregation; to differentiate between the two types of rows in the output, a flag is required in the output. You must specify the name of the flag field in that case (the type is boolean).
Group fields table	Specify the fields over which you want to group. Click Get Fields to add all fields from the input stream(s).
Aggregates table	Specify the fields that must be aggregated, the method and the name of the resulting new field. Here are the available aggregation methods : • Sum • Average (Mean) • Minimum • Maximum • Number of values (N) • Concatenate strings separated by , (comma) • First non-null value • Last non-null value • First value (including null) • Last value (including null) • Cumulative sum (all rows option only!) • Cumulative average (all rows option only!) • Standard deviation • Concatenate strings separated by <Value>: specify the separator in the Value column • Number of distinct values The metadata injection values for the aggregation type are (in respective order): SUM, AVERAGE, MIN, MAX, COUNT_ALL, CONCAT_COMMA, FIRST, LAST, FIRST_INCL_NULL, LAST_INCL_NULL, CUM_SUM, CUM_AVG, STD_DEV, CONCAT_STRING, COUNT_DISTINCT

IV. Creating edges between operations (hops)

1) Hover the mouse pointer over the source operation node. You should be able to see the options like in figure (1) below.
2) Press&hold the option pointing to the right (see Figure below (1)) to create the output edge from the given operation node.
3) Drag the mouse pointer to the operation node you want to be the destination of the edge.



(1)

(2)

(3)