Creating a Flow Variable
1. Right-click the workflow in KNIME Explorer & select Workbook Variables.
2. Use a Create or Copy button to create a Flow Variable at any point in your workflow.
3. Use any of the nodes converting data into Flow Variables.
4. Use the node configuration window in the Flow Variables tab, fill in a blank box with the name of the Flow Variable.

String Manipulation (Variable)
This node is the Flow Variables version of the String Manipulation node. Similarly, other nodes have their own version for Flow Variables like the Rule Engine Variable node & the Math Formula Variable node.

Merge Variables: Combines Flow Variables from two or more separate branches.
To add a branch click the two dots in the bottom left corner. If Flow Variables with the same name are collected, the Flow Variable in the top most connection is used.

Inject Variables (Data)
Adds (Injects) the Flow Variables at its Flow Variable input port into the data table at the top most connection if the Flow Variable data table is unrolled (un-enabled) in the node output port.

Extract Variables (Data)
Extracts the Flow Variables containing data into a data table at the top most connection if the Flow Variable data table is unrolled (un-enabled) in the node output port.

A Metanode or Component is a node that contains other nodes.
Creating a Metanode or Component
Select all relevant nodes, right-click and select Collapse into Metanode for a Metanode or Discapsulate into Component for a Component. Right-clicking a Metanode or Component opens the context menu with a number of options such as open, expand, resize, or collapse, and save as template.

ORCHESTRATION

Container Input (Table)
Receives a data table from the caller workflow. If no input is provided, the template default data table is used. Similar nodes are available to exchange Flow Variables & credentials. The corresponding "Container Output (Table)" node returns the results as a data table.

Container Input (JSON)
Receives a JSON-data structure from the caller workflow. If no input is provided, the template default structure is used. The corresponding "Container Output (JSON)" node returns the results as a data structure.

GET Request
Calls a REST service in GET mode. The node can send one single service request set in the configuration window, or multiple service requests stored in a column of the input table. Responses are saved in the output data table. Options to set authentication, request header, & request header content are available.

POST Request
Calls a REST service in POST mode. The node can send one single service request set in the configuration window, or multiple service requests stored in a column of the input table. Responses are saved in the output data table. Options to set authentication, request header, & request header content are available.

CASE Switch (Data)
Selectively activates only one of its three output ports, enabling three alternative paths for the calling workflow. The top output port can be configured manually or automatically via the value of a Flow Variable. The "If" node provides alternative ways to execute different workflow branches. All conditions with a Switch node & an If node can be paired up freely - loops can start with data and end with Flow Variables.

Loop End: Ends a loop by concatenating the resulting rows from each iteration.

Variable Loop End: Loop End nodes do not only work on data. At the end of a loop you might want to pass the results as a Flow Variable - like in the Variable Loop End node.

Line End: Finishes the corresponding script file.

CASE Switch Data (End)
Collects the resulting data rows from the active one among the branches connected to its input ports. The End node works similarly & is paired with the If node switch.

Try (Data Port)
Starts a try-catch construct to enable an alternative path for the data flow in case of failure in the main branch. If the conflict in the main branch is activated, the secondary branch is executed. It must be closed by a Catch node.

Empty Table Switch
Provides an alternative path for the data flow. If the main branch has no data rows, the top output port is deactivated while the bottom output port is activated. This setup is especially useful if the top output port deactivated the bottom output port or vice versa. The "If" node is also used to deactivate a branch even if it's active.

Active Branch Inverter
Changes the activity status of the branches. If the input port is active, the output port becomes inactive & vice versa. It's often used to force a branch to produce an output even if it’s inactive & vice versa. Some nodes to start & end a switch construct work on data, others on Flow variables, others on models, & others on database queries. All Start and End nodes can be paired up freely - loops can start with data and end with Flow Variables.

Resources
- E-Books: KNIME Advanced Luck covers these advanced workflows & more. Practicing Data Science is a collection of data science case studies from past projects. Both available at knime.org/bibliography
- KNIME Blog: Engaging topics, challenges, industry news, & knowledge nuggets at knime.org/blog
- E-Learning Course: Take our free online self-paced courses to learn about the different steps in a data science project (with exercises & solutions to test your knowledge). Visit knime.org/lms
- KNIME Hub: Browse and share workflows, nodes, and components. Add ratings, or compare to other workflows at hub.knime.com
- KNIME Forum: Join our global community & engage in conversations at forum.knime.com
- KNIME Server: For learn-by-collaboration, installation, management, & deployment check out KNIME Server at www.knime.com/knime-server

Knime Analytics Platform
A workflow is a sequence of operations that is repeated until a condition is met. If has a start, an end, & a loop body of operations. A loop is implemented using a Loop Start & End combination, or a different combination of KNIME Workflows & the Loop Loop port, depending on the kind of loops. Some nodes to start & end a loop include the Loop Variables. These nodes can be paired up freely - loops can start with data and end with Flow Variables & vice versa.

Hidden Flow Variable Ports
Each node has hidden Flow Variable ports to accept incoming Flow Variables & propagate them to the upstream nodes. To make the node visible, right-click the node & select Show Flow Variable Ports. Only ports of the same type can be connected.