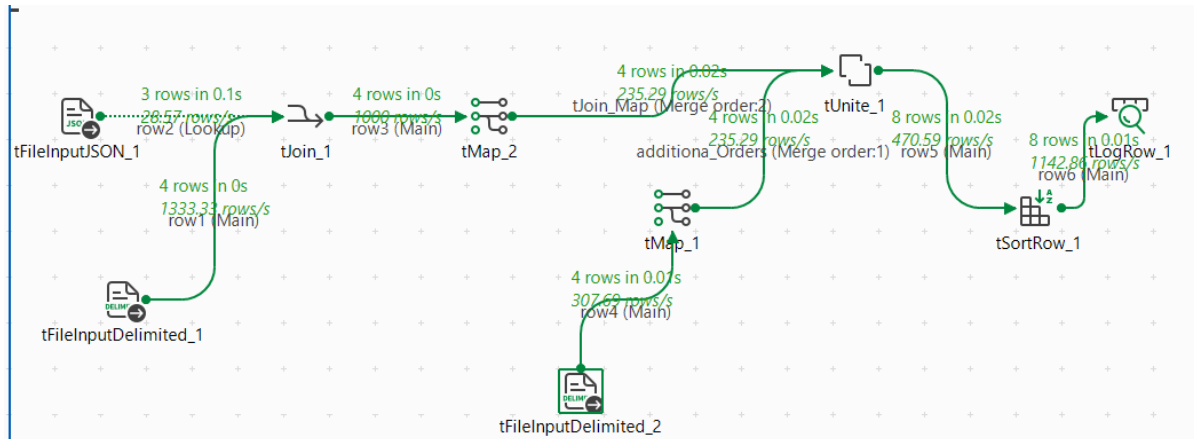


Sample Input Files

Summary of Data Flow



1. tFileInputJSON – Read `customers.json`

→ **Purpose:** Read customer details.

⚙️ Configuration:

- **File name:** `customers.json` (use full path or relative path)
- **Read by:** JSON Path
- **Loop XPath query:** `/root[*]`
- **Schema:**
 - `customer_id` (Integer)

✓ Steps:

1. Drag tFileInputJSON.
2. Set file path.
3. Click “Edit schema” → Add 3 fields.
4. Map:
 - `/customer_id` → `customer_id`

◆ 2. tFileInputDelimited – Read `orders.csv`

→ **Purpose:** Load standard orders.

⚙ Configuration:

- **File name:** orders.csv
- **Field separator:** ,
- **Header:** 1
- **Schema:**
 - order_id (Integer)
 - customer_id (Integer)
 - amount (Float or Double)

✓ Steps:

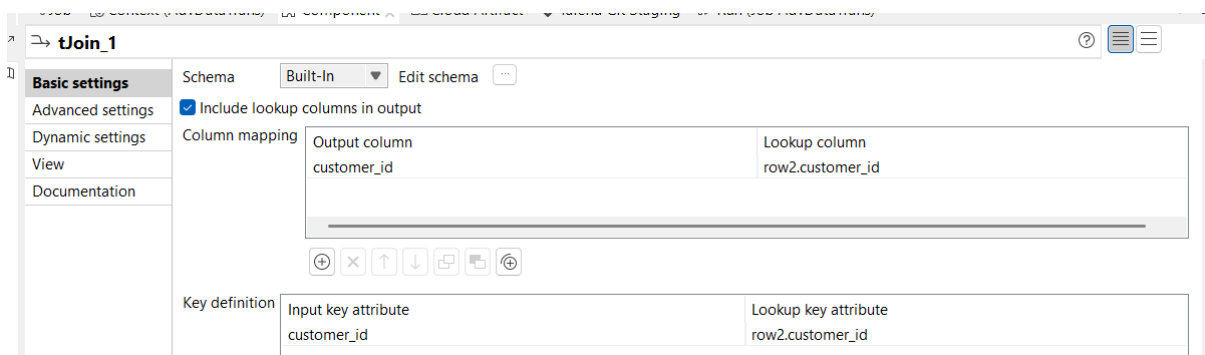
1. Drag tFileInputDelimited.
2. Set file path and schema.

◆ 3. tJoin – Join Orders with Customers

→ **Purpose:** Combine customer info into each order (inner join).

⚙ Configuration:

- **Main input:** orders.csv
- **Lookup input:** customers.json



✓ Steps:

1. Drag tJoin.
2. Connect:
 - Main: orders.csv → tJoin (Main Row)
 - Lookup: customers.json → tJoin (Lookup Row)
3. Set join keys in "Lookup" tab.

◆ 4. tMap (after tJoin) – Map Joined Output

→ **Purpose:** Format and add fields.

⚙ Configuration:

- Add output fields:
 - `order_id` (from main)
 - `customer_id` (from main)

✓ Steps:

1. Drag `tMap`.
 2. Connect `tJoin` → `tMap`.
 3. In `tMap`:
 - Create output.
 - Map fields from input to output.
 - Use `row2` for lookup fields.
-

◆ 5. `tFileInputDelimited` – Read `additional_orders.csv`

→ **Purpose:** Read extra orders (not joined).

⚙ Configuration:

- Same as `orders.csv`

✓ Steps:

1. Drag new `tFileInputDelimited`.
 2. Set same schema and file path.
-

◆ 6. `tMap` (for `additional_orders.csv`) – Add NULLs

→ **Purpose:** Add missing customer fields (`null` or empty).

⚙ Configuration:

- Add output fields:
 - `order_id`
 - `customer_id`

✓ Steps:

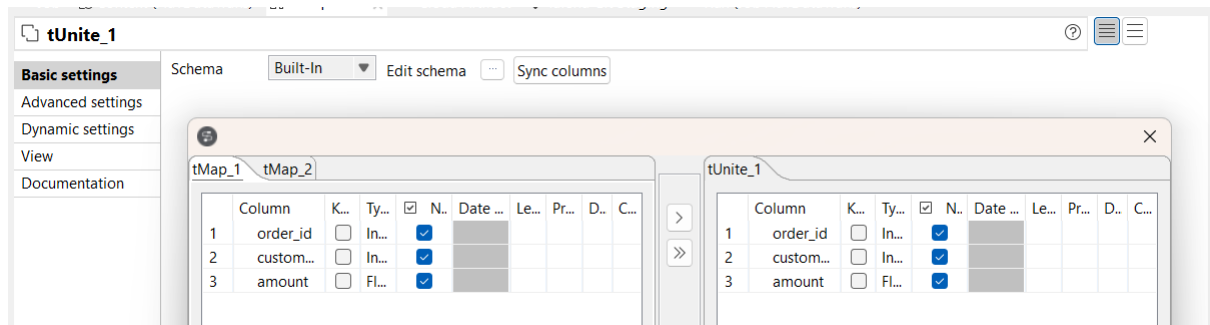
1. Drag `tMap`.
2. Connect `additional_orders.csv` → `tMap`.
3. In output, map:
 - Input columns directly.

- Add "" or null for customer fields.

◆ 7. tUnite – Combine Both Outputs

→ **Purpose:** Combine both tMap outputs.

Schema:



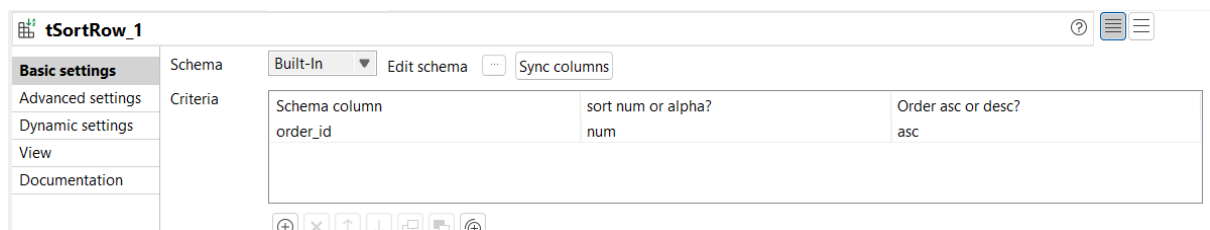
⚙ Configuration:

- Inputs must have identical schema.
- Connect both outputs to tUnite.

✓ Steps:

1. Drag tUnite.
2. Connect:
 - tMap (joined) → tUnite input 1
 - tMap (additional) → tUnite input 2

◆ 7. tSort – for sorting the orderwise



◆ 8. tLogRow– Output

→ **Purpose:** Display or store the result.

tLogRow_1

Basic settings | Advanced settings | Dynamic settings | View | Documentation

Schema: Built-In | Edit schema | Sync columns

Mode

- ☐ Basic
- ☒ Table (print values in cells of a table)
- ☐ Vertical (each row is a key/value list)

☒ Print content with log4j

⚙ Configuration (tFileOutputDelimited):

- Set output file path.
- Field separator: ,
- Include header: Yes

✓ Steps:

1. Drag tLogRow or tFileOutputDelimited.
2. Connect tUnite → Output.
3. Run the job.

✓ Final Output (CSV or Console)

```
order_id,customer_id,amount
101,1,500
103,1,150
104,3,300
105,2,180
106,3,220
```
