

Node Guide: Update Row

Overview

The **Update Row** node allows you to modify existing records in a selected table within your workflow. You can specify which rows to update using match conditions (like SQL **WHERE** clauses) and define new values for any of the columns.

What This Node Does

- Updates one or more rows in a table (entity)
 - New values can be static or pulled from variables
 - Records to update are filtered using match conditions
 - Returns:
 - **update_status**: Success or failure of the operation
 - **no_of_rows_updated**: Number of rows affected
 - **updated_rows**: List of updated records (if needed for further processing)
-

Configuration Details

1. Entity Name (Table)

- **Mandatory**
- Select the table in which data needs to be updated
- Choose from the dropdown list of available entities

2. Update Values

- Once a table is selected, its columns will appear

- For each column, you can:
 - Enter a **static value**, or
 - **Select a variable** from the dropdown to provide the new value dynamically

3. Match Conditions

- Specify conditions to decide which rows to update (similar to SQL **WHERE**)
 - For each condition:
 - Select a **column**
 - Choose an **operator** (e.g., **=**, **!=**, **LIKE**)
 - Enter a **value** or map a **variable** from the dropdown
 - You can define **multiple conditions**, and they will be combined logically (e.g., **AND** logic)
-

Inputs

- **Entity Name:** Select from table/entity dropdown
 - **Update Values:** Static input or mapped variable values for each column
 - **Match Conditions:** Column, operator, and value or variable
-

Outputs

- **update_status:**
 - Stores whether the update succeeded or failed
- **no_of_rows_updated:**
 - Stores the number of records that were updated
- **updated_rows:**

- Stores the actual updated row data (if needed for later use)
-

When to Use

Use this node when you want to:

- Modify specific records in a table based on user actions or data updates
 - Track changes or status updates within the flow
 - Dynamically update data using live values from agent variables
-

Example Flow: Update User Status

Scenario

You want to update a user's profile status to **verified** based on their ID.

Flow Steps

1. User Input or Lookup Node

- Capture or retrieve the **user_id**

2. Update Row Node

- Table Name: **user_profiles**
- Update Column: **status** → Static value: **verified**
- Match Condition: **user_id** = variable **user_id**
- Output:
 - **update_status** → **status_result**
 - **no_of_rows_updated** → **rows_updated**
 - **updated_rows** → **updated_user_data**

Summary

The Update Row node gives you fine-grained control to change records in your workflow database. With configurable value mapping and filtering conditions, it provides a powerful and flexible mechanism to keep your data up-to-date and reactive to changes in the flow.