Node Guide: PUT API

Overview

The **PUT API** node allows you to **update data** in an external system using a PUT request. You provide the endpoint, any required headers and authentication, and a customizable JSON body — which can include values from your workflow variables.

This node is typically used when you need to **modify or replace** existing records in a database, CRM, or any third-party system.

What It Does

Once executed, this node:

- Sends a **PUT request** to the specified API URL
- Includes a **JSON body** that can be static or dynamic (with variables)
- Accepts headers, path parameters, and authentication if needed
- Saves the **response code**, **response body**, and **headers** in variables

---

Configuration Details

1. **PUT Endpoint URL**
    - The URL of the API you want to call
    - You can include dynamic path variables like: `https://api.example.com/users/{{user_id}}`
2. **Headers and Request Parameters (Optional)**
    - Add key-value pairs like `Content-Type: application/json`

- Both key and value can be static or use workflow variables

3. **JSON Body**
    - Enter your data in JSON format

Example with static values:

json
CopyEdit

```
{
  "name": "John Doe",
  "status": "active"
}
```

Example with variables:

json
CopyEdit

```
{
  "name": "{{user_name}}",
  "status": "{{user_status}}"
}
```

4. **Authentication**
    - Choose the authentication method: **Basic**, **Bearer**, or **OAuth2**
    - You can input credentials directly or use variables
5. **Test the API Call**
    - Run a test before saving to verify your setup
    - Helps ensure the node is properly configured and the endpoint is reachable

---

Inputs

- **PUT Endpoint URL**

- **Headers and Request Parameters (Optional)**
- **JSON Body (with or without variables)**
- **Authentication (if required)**

Outputs

- `response_code` — HTTP status code from the response
- `response_body` — Full content returned by the API (usually JSON)
- `response_headers` — Metadata from the response

You can assign these outputs to variables and use them later in the workflow.

When to Use

Use the PUT API node when you need to:

- Update user profiles or records in another system
- Change the status or data of an existing entry
- Overwrite resources with new values

Example Flow: Update User Subscription Plan

Scenario

You want to upgrade a user's subscription in your billing system after payment is confirmed.

Flow Steps

1. **PUT API Node**

- URL: `https://billing.example.com/api/users/{{user_id}}/subscription`
- Authentication: Bearer token stored in a variable

JSON Body:

json
CopyEdit

```
{
  "plan": "{{new_plan}}",
  "renewal": true
}
```

-
- Store `response_body` in variable `update_status`

2. **Send Message Node (Teams or WhatsApp)**
   - Message: `"Hi {{user_name}}, your subscription has been updated to the {{new_plan}} plan."`

Summary of the Flow

- PUT request updates the user's subscription on an external system
- Response is saved and used to notify the user with confirmation