

Memoria Práctica 1

Jaime Parra Jiménez

INTEGRACIÓN DE TECNOLOGÍAS Y SERVICIOS INFORMÁTICOS

Universidad de Almería

Correo: jj451@inlumine.ual.es

22 de octubre de 2025

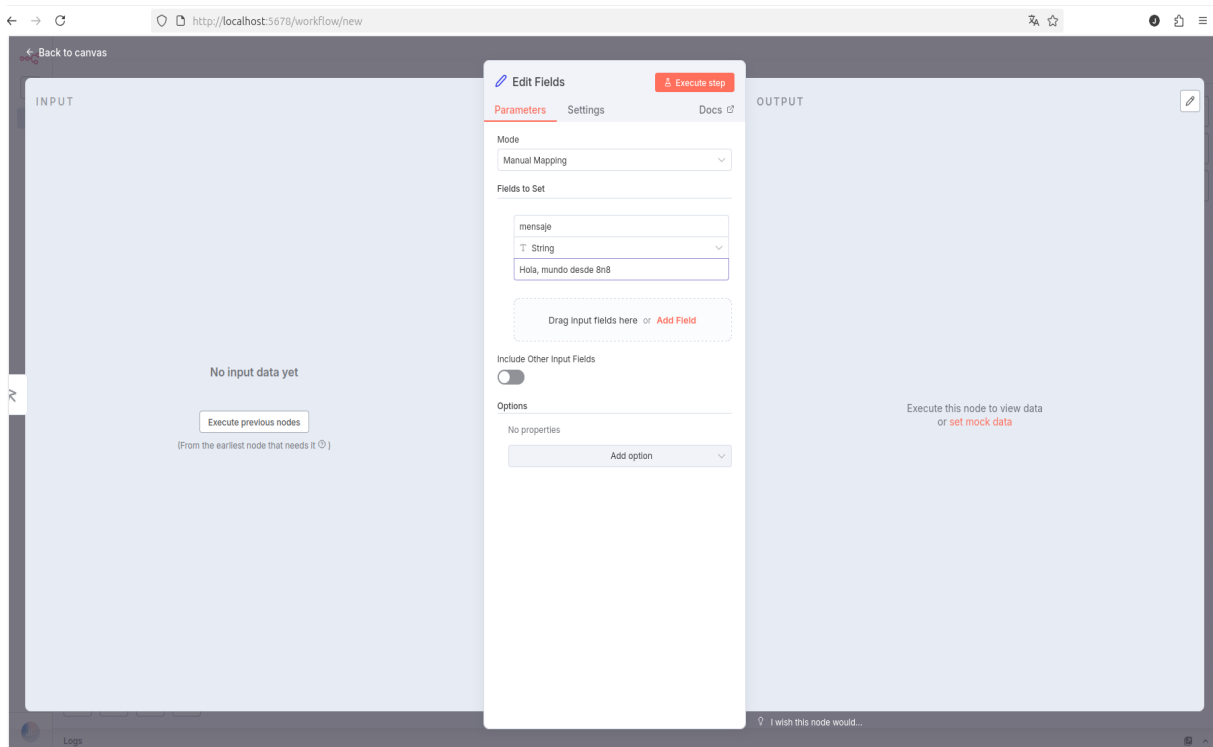
Índice general

1. Ejercicio Guiado 1	2
2. Ejercicio Guiado 2	4
3. Ejercicio Guiado 3	6

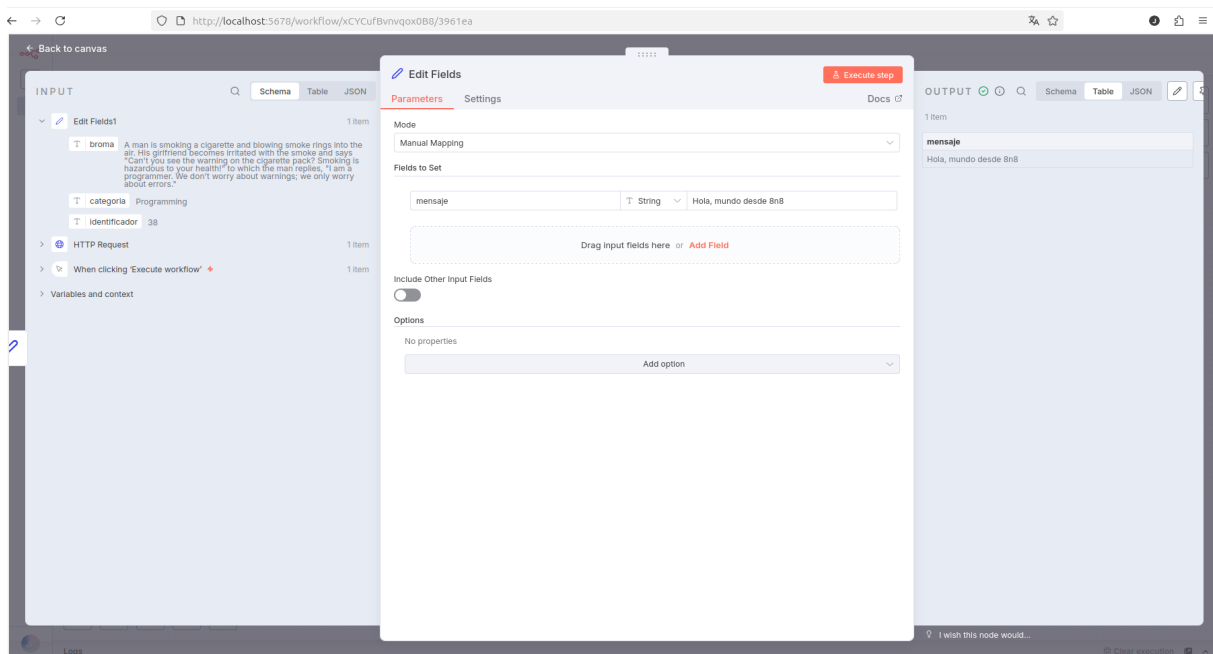
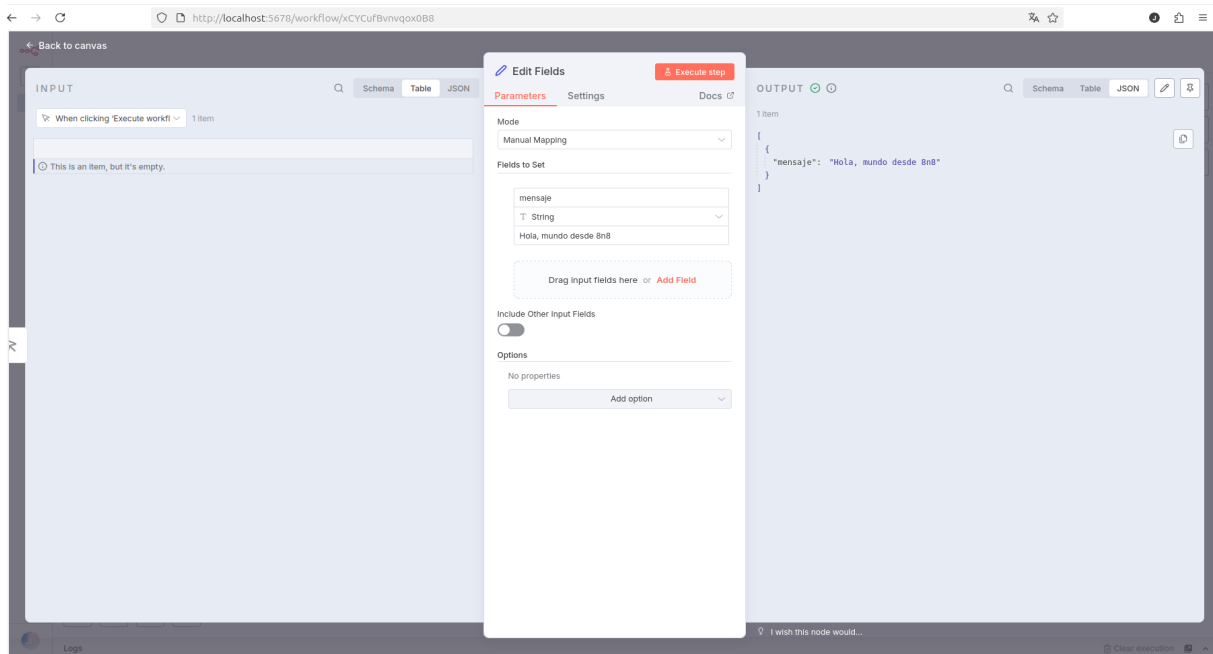
Capítulo 1

Ejercicio Guiado 1

En el Ejercicio Guiado 1 el propósito será crear un flujo en el canva para que muestre el mensaje “Hola, mundo desde n8n”. Para ello, lo primero que se debe hacer es añadir a la interfaz de n8n, la cual se le llama “Canvas”, un Manual Trigger que será el punto de partida ideal para el desarrollo y depuración del proyecto. A continuación se tiene que añadir el nodo Edits Fields (Set), útil para crear, modificar o eliminar campos de datos, donde se creará un valor de nombre “mensaje” y de valor “Hola, mundo desde n8n.”.



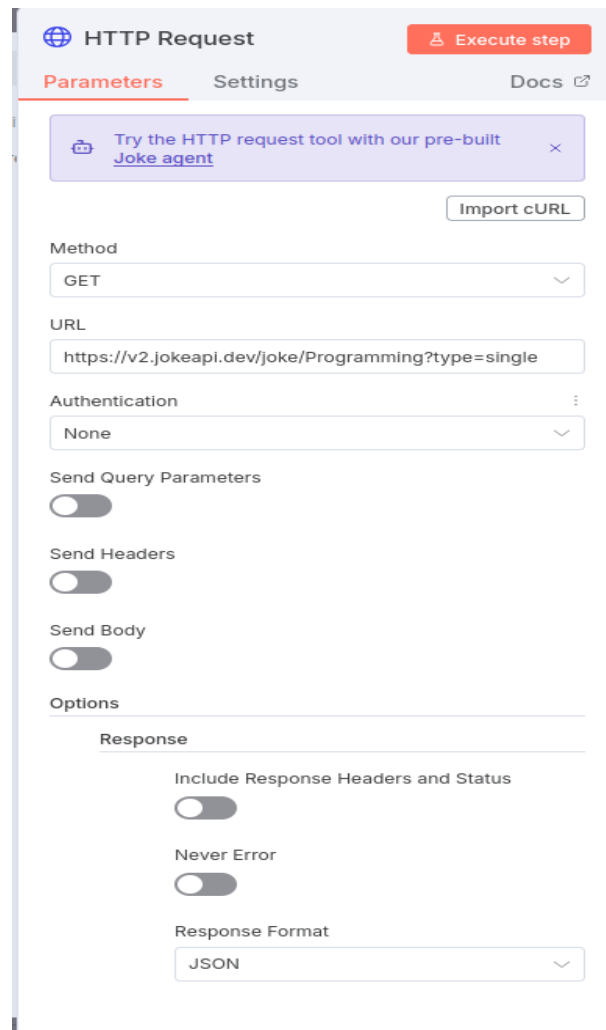
Una vez configurado el nodo correctamente se debe presionar el botón de “Test Step” el cual ejecutará el flujo hasta el nodo. Una vez ejecutado el nodo se podrá ver una vista “Table” donde se verá una columna “mensaje” y una fila con el valor “Hola,mundo desde 8n8” y otra vista “JSON” para ver la estructura de datos subyacente donde se verá el array de objetos con el mensaje. A continuación se podran ver dos fotos mostrando ambas vistas.



Capítulo 2

Ejercicio Guiado 2

El propósito del Ejercicio Guiado 2 será interactuar con una API REST Pública. Para ello se tendrá que introducir el nodo HTTP Request para consumir datos de un servicio externo. Una vez añadido habrá que configurarlo para poder consumir datos de una URL. En la configuración se debe definir el Método, la URL y el formate de respuesta.



The image shows a web-based configuration interface for an HTTP Request tool. At the top, there's a header with a globe icon, the text "HTTP Request", and a red button labeled "Execute step". Below the header are three tabs: "Parameters" (active), "Settings", and "Docs". A purple notification box at the top left says "Try the HTTP request tool with our pre-built Joke agent" with a close button. Below this is an "Import cURL" button. The main configuration area includes: a "Method" dropdown set to "GET"; a "URL" text input containing "https://v2.jokeapi.dev/joke/Programming?type=single"; an "Authentication" dropdown set to "None"; three toggle switches for "Send Query Parameters", "Send Headers", and "Send Body", all currently turned off; an "Options" section with a "Response" subsection containing two more toggle switches, "Include Response Headers and Status" and "Never Error", both turned off; and a "Response Format" dropdown set to "JSON".

Una vez se tenga el nodo configurado correctamente hay que ejecutarlo y mostrará un panel de resultados en la vista “JSON” con varios campos como error, category, type, joke, flogs, e id. De esta forma se podrá entender la estructura de los datos que se reciben.

The screenshot shows a workflow editor interface with a central 'HTTP Request' node configuration panel. The left pane shows the 'INPUT' section with a trigger 'When clicking "Execute workflow"' and a message 'This is an item, but it's empty.' The central panel is configured for a GET request to 'https://v2.jokeapi.dev/joke/Programming?type=single'. The right pane shows the 'OUTPUT' in JSON format, displaying a single item with the following structure:

```
{
  "error": false,
  "category": "Programming",
  "type": "single",
  "joke": "A man is smoking a cigarette and blowing smoke rings into the air. His girlfriend becomes irritated with the smoke and says \"Can't you see the warning on the cigarette pack? Smoking is hazardous to your health!\" to which the man replies, \"I am a programmer. We don't worry about warnings; we only worry about errors.\"",
  "flags": {
    "nsfw": false,
    "religious": false,
    "political": false,
    "racist": false,
    "sexist": false,
    "explicit": false
  },
  "id": 38,
  "safe": true,
  "lang": "en"
}
```

At the bottom of the interface, there is a 'Logs' section and a 'Node executed successfully' message.

Capítulo 3

Ejercicio Guiado 3

En el Ejercicio Guiado 1 el propósito será consumir la respuesta “cruda” de la API y transformarla en un objeto de salida limpio. Para ello se debe añadir un node Edit Fields(Set) para añadirle expresiones y así limpiar las salida. En la salida se quiere que solamente se muestre el texto, la categoría y la id. Para conseguir este propósito hay que añadir las expresiones “ json.joke ”, “ json.category ” y “ json.id ”.

The screenshot shows the n8n editor interface. On the left, the 'INPUT' section displays a table with one item. The table has columns: error, category, type, joke, flags, id, and safe. The data for the first item is: error: false, category: Programming, type: single, joke: A man is smoking a cigarette and blowing smoke rings into the air. His girlfriend becomes irritated with the smoke and says "Can't you see the warning on the cigarette pack? Smoking is hazardous to your health!" to which the man replies, "I am a programmer. We don't worry about warnings; we only worry about errors.", flags: nsfw: false, religious: false, political: false, racist: false, sexist: false, explicit: false, id: 38, safe: true.

In the center, the 'Edit Fields1' node is configured. The 'Mode' is set to 'Manual Mapping'. Under 'Fields to Set', three fields are defined: 'broma' (Type: String, Expression: {{ \$json.joke }}), 'categoria' (Type: String, Expression: {{ \$json.category }}), and 'identificador' (Type: String, Expression: {{ \$json.id }}). The 'Include Other Input Fields' toggle is turned off. The 'Options' section shows 'No properties'.

On the right, the 'OUTPUT' section shows the resulting JSON object: { "broma": "A man is smoking a cigarette and blowing smoke rings into the air. His girlfriend becomes irritated with the smoke and says 'Can't you see the warning on the cigarette pack? Smoking is hazardous to your health!' to which the man replies, 'I am a programmer. We don't worry about warnings; we only worry about errors.'", "categoria": "Programming", "identificador": "38" }.