

Integrations > Built-in nodes > Core nodes

Extract From File



A common pattern in n8n workflows is to receive a file, either from and [HTTP Request node](#) (for files you are fetching from a website), a [Webhook Node](#) (for files which are sent to your workflow from elsewhere), or from a local source. Data obtained in this way is often in a binary format, for example a spreadsheet or PDF.

The Extract From File node extracts data from a binary format file and converts it to JSON, which can then be easily manipulated by the rest of your workflow. For converting JSON back into a binary file type, please see the [Convert to File](#) node.

Operations

Use the **Operations** drop-down to select the format of the source file to extract data from.

- **Extract From CSV:** The "Comma Separated Values" file type is commonly used for tabulated data.
- **Extract From HTML:** Extract fields from standard web page HTML format files.
- **Extract From JSON:** Extract JSON data from a binary file.
- **Extract From ICS:** Extract fields from iCalendar format files.
- **Extract From ODS:** Extract fields from ODS spreadsheet files.
- **Extract From PDF:** Extract fields from Portable Document Format files.
- **Extract From RTF:** Extract fields from Rich Text Format files.
- **Extract From Text File:** Extract fields from a standard text file format.
- **Extract From XLS:** Extract fields from a Microsoft Excel file (older format).
- **Extract From XLSX:** Extract fields from a Microsoft Excel file.
- **Move File to Base64 String:** Converts binary data to a text-friendly [base64](#) format.

Example workflow

In this example, a Webhook node is used to trigger the workflow. When a CSV file is sent to the webhook address, the file data is output and received by the Extract From File node.

[View workflow file](#)

Set to operate as 'Extract from CSV', the node then outputs the data as a series of JSON 'row' objects:

```
1 {
2   "row": {
3     "0": "apple",
4     "1": "1",
5     "2": "2",
6     "3": "3"
7   }
8   ...
```

Receiving files with a webhook

Select the Webhook Node's **Add Options** button and select **Raw body**, then enable that setting to get the node to output the binary file that the subsequent node is expecting.

Node parameters

Input Binary Field

Enter the name of the field from the node input data that contains the binary file. The default is 'data'.

Destination Output Field

Enter the name of the field in the node output that will contain the extracted data.

This parameter is only available for these operations:

- Extract From JSON
- Extract From ICS
- Extract From Text File
- Move File to Base64 String

Cookie consent

We use cookies and other tracking technologies to improve your browsing experience, to analyze our website traffic, assist our marketing efforts and to understand where our visitors are coming from.

[Privacy policy](#)

Accept

Manage settings

Extract text from a PDF file

by amudhan

View template details

Scrape and store data from multiple website pages

by Miquel Colomer

View template details

Browse Extract From File integration templates, or search all templates

Popular integrations

- Google Sheets
- Telegram
- MySQL
- Slack
- Discord
- Postgres
- Show more

Top integration categories

- Development
- Communication
- Langchain
- AI
- Data & Storage
- Marketing
- Show more

Top guides

- Telegram bots
- Open-source chatbot
- Open-source LLM
- Open-source low-code platforms
- Zapier alternatives
- Make vs Zapier
- Show more

Trending combinations

- HubSpot and Salesforce
- Twilio and WhatsApp
- GitHub and Jira
- Asana and Slack
- Asana and Salesforce
- Jira and Slack
- Show more

Trending templates

- Creating an API endpoint
- AI agent chat
- Scrape and summarize webpages with AI
- Very quick quickstart
- Pulling data from services that n8n doesn't have a pre-built integration for
- Joining different datasets
- Show more

Cookie consent

We use cookies and other tracking technologies to improve your browsing experience, to analyze our website traffic, assist our marketing efforts and to understand where our visitors are coming from.

[Privacy policy](#)