

n8n workflows 学习指南

n8n workflows 自动化指南目录

1. 引言

- 1.1 背景信息
- 1.2 目标读者
- 1.3 目的与价值
- 1.4 为什么选择n8n

2. 安装与配置

- 2.1 云平台试用（无需部署，注册使用）
- 2.2 本地/私有化部署
- 2.3 界面介绍
- 2.4 凭证管理
- 2.5 n8n汉化

3. 学习与使用

- 3.1 常用节点
- 3.2 基本工作流创建
- 3.3 避坑指南
- 3.4 进阶应用 - 模板套用
- 3.5 进阶应用 - mcp
- 3.6 进阶应用 - 自然语言编程
- 3.7 进阶应用 - 社区节点

4. 参考资料与术语表

- 4.1 参考资料
- 4.2 术语表

5. AI卓越中心介绍

- 5.1 职能与支持
- 5.2 加入我们的方式

1. 引言

1.1 背景信息：拥抱人工智能浪潮

人工智能正以惊人的速度重塑我们的世界，从根本上改变着商业运营、社会互动和个人生活。它不仅仅是一项技术，更是一场深刻的时代变革。在这场汹涌的浪潮中，自动化、智能化工具成为提升效率、释放潜力、优化决策的关键。本指南将带您深入了解n8n，助您在智能时代乘风破浪。

💡 机遇：赋能与进化

- **企业层面：**
 - **提升效率与创新：** AI驱动的自动化能够优化业务流程，将员工从重复性劳动中解放出来，从而激发新产品和服务的开发。
 - **数据洞察与决策：** 强大的海量数据分析能力，为精准的商业决策和市场洞察提供坚实支持。
- **科研领域：**
 - 为**基础科学研究**提供强大的分析和模拟工具，**加速科学发现**，开辟**新的研究方向**，并促进**跨学科合作**。
- **个人层面：**
 - **职业发展：** 催生了对新技能的需求和大量新兴岗位，促进个人能力升级和职业转型。
 - **生活智能化：** AI融入日常生活，带来前所未有的便捷与个性化智能体验。

⚠️ 挑战：适应与应对

- **企业层面：**
 - **高投入与转型成本：** 企业需要投入大量资源进行AI基础设施建设、人才培养和组织架构调整。
 - **数据安全与伦理：** AI应用中的数据隐私、算法偏见和责任归属问题是必须面对的挑战。
- **科研领域：**
 - 面临**计算资源紧缺**、**高水平AI人才竞争**、**高质量数据获取困难**，以及**研究成果转化落地和伦理规范**的挑战。
- **个人层面：**
 - **人才结构变革：** 部分传统岗位面临被取代的风险，劳动力市场的再培训与技能重塑迫在眉睫。
 - **技术快速迭代：** AI技术日新月异，要求个人保持持续学习的能力，以快速适应市场变化。

1.2 目标读者

- **技术零基础的业务人员：** 旨在帮助您**快速入门**，理解 n8n 如何**简化日常重复性工作**，通过**模板和简单工作流**轻松实现业务自动化，显著提升工作效率。
- **技术支持与开发人员：** 旨在帮助您**深入理解 n8n 的部署、配置、运行和维护**，掌握**系统管理、故障排查和团队赋能**的能力，确保 n8n 服务的稳定运行和高效支持。

1.3 目的与价值

- **核心目的：** 让技术零基础的业务人员和负责技术支持的同事都能**快速上手 n8n**，并充分利用其强大的自动化能力。
- **对业务部门的价值：**
 - **赋能业务用户：** 无需编程知识即可**自动化日常任务**，例如邮件处理、数据同步、报告生成等。
 - **提升核心效率：** 通过自动化**节省大量时间**，让团队能更专注于创造价值的核心业务。
 - **探索自动化潜力：** 即使是技术新手也能通过**AI 驱动的自动化模板和即时连接应用程序**来发现更多业务价值。
- **对技术部门的价值：**
 - **灵活与强大：** 理解 n8n **自托管和低代码**的特性，可以**自定义代码节点**，实现**无限制的复杂自动化**。

- **系统管理与维护**：掌握 **n8n 服务器的部署**（如 npm 或 Docker）、**高级配置**（如环境变量）和**日常维护**，确保服务的**稳定性、安全性与可扩展性**。
- **构建AI自动化能力 (MCP)**：深入理解 **Model Context Protocol (MCP)** 如何让 AI 模型以标准化方式与**外部工具和数据源交互**。学习如何**将现有 n8n 工作流封装为 AI 代理可调用的“工具”**，从而**构建强大的端到端智能工作流**。
- **高效问题解决与支持**：掌握**常见问题的排查方法**，以及如何有效地**协助非技术同事解决工作流问题**，成为团队内部的 n8n 专家。

1.4 为什么选择n8n

低代码平台是赋能全所非技术人员的最优解工具，n8n则是低代码工具中发展性最好的一个。

方案/工具	无代码 (Dify)	低代码 (n8n)	传统代码开发 (python)	智能化代码开发 (如Cursor)
目标用户	业务人员、市场人员、非技术背景的开发人员	IT专业人员、开发者、既懂业务又懂技术的复合型人才	专业软件工程师、开发团队	开发者、工程师、研究人员
开发速度	最快，数小时到数天即可构建应用	快速，熟练后数小时到数天即可构建应用	较慢，数周到数月甚至更长	快速，根据AI工具熟练度与剖析问题的能力数天到数星期不等
核心理念	“配置优于编码”，通过可视化界面和预制模块快速搭建。	“更少代码，更多创造”，通过可视化与少量代码结合，加速开发。	“代码拥有一切”，通过编写代码实现完全的控制和定制。	AI赋能编码，提升开发效率与质量
灵活性与定制化	较低，受限于平台提供的功能和模板	较高，允许通过编写代码（如JavaScript、Python）进行扩展和定制。	极高，可以实现任何复杂的功能和独特的界面	高，代码由AI生成，可高度定制与修改
技术门槛	极低，无需编程知识	较低，需要基本的编程逻辑和少量编码能力	极高，需要精通编程语言、框架和软件工程原理	中，需有编程基础以审查、优化AI生成代码
主要应用场景	构建简单的内部工具、自动化流程、AI应用原型等	企业级自动化、系统集成、连接不同的API和服务、构建复杂的业务流程	开发大型、复杂的软件系统、高性能应用、底层系统开发	代码生成、智能补全、错误检测、文档生成、测试用例编写
常用工具	Dify、Zapier、Bubble	n8n、Retool、Mendix	python、java、C	Cursor、GitHub Copilot、Codeium

2. 安装与配置

2.1 快速开始 (n8n Cloud)

对于希望立即体验而无需任何部署的用户，n8n官方云服务是最佳选择。

- **官方网址：**<https://n8n.io/>
- **优势：**无需部署，注册即用，官方负责所有维护工作，让您专注于 workflows 构建。

2.2 本地/私有化部署

部署方式对比

1. n8n 云服务 (Cloud Hosting) - 最快速、无需部署

- **描述：**最简单快捷的入门方式，无需任何本地安装和配置。访问 <https://n8n.io> 注册即可开始，官方提供免费试用。
- **适用人群：**初学者、业务用户、希望快速验证自动化想法的用户。

2. 使用 npm 安装 - 适合开发者快速本地启动

- **描述：**对于熟悉 Node.js 环境的开发者，通过简单的 `npm` 命令行指令即可在本地快速启动 n8n，适合在本地进行开发、测试和调试。

3. 使用 Docker 安装 - 推荐的本地或服务器部署方式

- **描述：**Docker 提供容器化部署，确保环境一致性，避免复杂的依赖问题。它封装了 n8n 及其所有依赖，是本地开发和中小型项目生产部署的理想选择。

4. 服务器自托管 (Self-hosting) - 灵活且注重隐私

- **描述：**n8n 是一款 **公平代码许可 (fair-code licensed)** 的软件，支持自托管，提供极高的灵活性、隐私和安全性控制。您可以在 Digital Ocean, AWS, Azure, Google Cloud 等任意云服务器上部署，并进行深度定制化配置。
- **适用人群：**对数据隐私、系统性能或集成有特定需求的企业和高级用户。

相关教程：

- 图文教程
https://blog.csdn.net/2501_90561511/article/details/147479401?spm=1001.2014.3001.5502
- 视频教程：
<https://www.bilibili.com/video/BV1kZNrzpEy6>

Docker 部署注意事项

如果选择 Docker 部署，请特别注意 `volumes` 映射配置：

- **第一个 Volume：**用于持久化存储 n8n 的工作流、凭证等核心数据。
 - **Host Path：**您服务器上的一个路径，例如 `/data/n8n/config`。
 - **Container Path：**固定为 `/home/node/.n8n`。
- **第二个 Volume：**用于 n8n 工作流读写本地文件。
 - **Host Path：**您服务器上用于存放处理文件的路径，例如 `/data/n8n/files`。
 - **Container Path：**在 n8n 节点中访问该文件夹的路径，例如 `/files`（请记住这个路径）。

2.3 界面介绍

n8n 的界面直观易用，主要由 workflow 画布、节点面板、参数配置区和执行日志等部分组成。详细介绍请参考：

<https://n8n.akashio.com/article/n8n-ui>

2.4 凭证管理 (Credentials)

凭证是 n8n 连接外部世界的“钥匙”。它用于安全地存储和管理连接外部服务（如数据库、API、邮箱）所需的授权信息（如 API 密钥、OAuth 令牌、用户名密码等）。

- **安全存储：**n8n 对凭证进行加密存储，保护您的账户安全。
 - **集中管理：**在一个地方统一创建、查看和管理所有凭证。
 - **一次配置，多次使用：**创建一次凭证后，可在多个 workflow 和节点中重复使用。
 - **避免硬编码：**将敏感信息与 workflow 逻辑分离，是更安全、更易于维护的最佳实践。
- 常用凭证：

- 大模型凭证
- 邮箱凭证
- 数据库凭证

图文示例请参考：

<https://jishuzhan.net/article/1951148541186977794>

2.5 n8n 汉化

在使用 Docker 部署时，可以通过添加环境变量和挂载中文语言包的方式实现界面汉化。

参考：

<https://zhuanlan.zhihu.com/p/1937586953865372387>

3. 学习与使用

3.1 常用节点

1. 触发 (Triggers)

触发器是 workflow 的起点，用于在特定事件发生时启动自动化流程。

• 手动触发 (Manual Trigger):

- **功能和用法：**手动触发节点用于**用户启动的 workflow**。它允许您在需要时手动运行 workflow，而不是等待特定事件发生。
- **配置：**作为核心节点，通常不需要复杂配置，只需在需要时点击执行即可。

• 定时触发 (Schedule Trigger):

- **功能和用法：**定时触发节点用于**基于时间的自动化任务**。您可以设置 workflow 在特定时间或按固定时间间隔自动运行。
- **配置：**您需要配置触发的时间、频率等参数。文档中也提到了该节点可能遇到的“常见问题”。

• Webhook (Webhook):

- **功能和用法:** Webhook 节点用于**API 触发的自动化**。它允许外部服务通过发送HTTP请求来启动n8n工作流，实现实时集成和响应。
- **配置:** 需要设置一个生产URL，并在您的MCP客户端中使用这个URL。文档中提到了“工作流开发”和“常见问题”。在导入工作流时，需要更新Webhook URL。

• 自动化 (Automation):

- **功能和用法:** 自动化是n8n的核心能力和目标。它涵盖了**连接任何带有API的应用程序并对其数据进行操作**，以及**构建复杂的、端到端的工作流**。n8n特别适用于**批量数据处理、搬运和引入AI到现有工作流**。
- **配置:** 自动化是n8n平台提供的整体能力，而非单一节点。各种触发器、处理节点和逻辑节点共同实现自动化。

• 邮件触发 (Email Trigger):

- **功能和用法:** 邮件触发 (IMAP) 节点可以根据新邮件或其他邮件事件来启动工作流。例如，可以构建工作流来**自动标记收到的Gmail邮件** 或**根据新邮件内容草拟回复**。
- **配置:** 需要配置IMAP相关参数，如服务器、端口、凭据等。

• 对话触发 (Chat Trigger):

- **功能和用法:** 对话触发节点用于响应聊天消息来启动工作流。例如，“AI代理聊天”和“Telegram AI聊天机器人”是趋势模板，它们可能通过对话触发来接收用户输入。
- **配置:** 配置聊天平台和监听的事件。

2. 输入 (Input)

输入节点用于从各种数据源获取信息，供工作流后续处理使用。

• 读本地文件 (Read Local File):

- **功能和用法:** 允许工作流读取本地文件系统上的文件。
- **配置:** 使用“Read/Write Files from Disk”核心节点来指定文件路径和读取方式。

• HTTP 请求 (HTTP Request):

- **功能和用法:** HTTP请求节点是“**半开放式节点**”，它允许n8n接入**几乎任何开放API的第三方服务**。可以用于从互联网获取数据。
- **配置:** 需要配置请求URL、方法 (GET、POST等)、头部信息和请求体等。可以与ChatGPT结合使用来编写代码片段，帮助连接不在官方节点库中的功能。

• MySQL 获取数据 (MySQL Get Data):

- **功能和用法:** n8n支持与MySQL数据库集成。可以用于从MySQL数据库中**查询和检索数据**。
- **配置:** 需要配置MySQL凭据和要执行的查询语句。例如，可以使用AI代理与PostgreSQL或MongoDB数据库进行交互，并支持自定义SQL查询和模式内省，MySQL也应有类似功能。

• RSS 网络搜索 (RSS Web Search):

- **功能和用法:** RSS Read和RSS Feed Trigger节点用于**读取RSS订阅源**，从而获取最新的网络信息或文章更新。可以构建“**基于RSS的阅读分流器**” 或**监控RSS源并总结文章**。
- **配置:** 输入RSS订阅源URL。

• Google 表格、云盘 (Google Sheets, Google Drive):

- **功能和用法：**n8n与Google Sheets和Google Drive有深度集成。您可以**读取、写入Google Sheets中的数据；访问和利用Google Drive中的文件**，例如构建AI助手来回答基于文档内容的问题。还可以监控Google Drive中的新文档，并使用AI进行总结。
- **配置：**需要配置Google凭据，并指定要操作的表格或文件路径。

3. 处理 (Process)

处理节点用于对输入数据进行转换、分析或执行特定操作。

• AI 代理 (AI Agent):

- **功能和用法：**AI代理节点是n8n中构建高级AI功能的核心。它将**LangChain框架可视化**，允许AI代理**发现、管理和运行工作流**。AI代理能够根据任务自动选择和执行工具（如搜索、存储）。
- **配置：**AI代理节点具有子节点，允许灵活配置**模型 (Model)**、**记忆 (Memory)**、**工具 (Tool)**和**输出解析器 (Output Parser)**。在使用社区节点作为AI代理工具时，必须设置环境变量 `N8N_COMMUNITY_PACKAGES_ALLOW_TOOL_USAGE=true`。

• 代码 (Code):

- **功能和用法：**代码节点是“**半开放式节点**”，支持在服务器端运行**JavaScript和Python代码**。这使得您可以**编写自定义逻辑**或处理n8n内置节点无法直接支持的复杂任务。
- **配置：**在节点内部编写JavaScript或Python代码，可以访问和处理工作流中的数据。

• 编辑字段 (Edit Fields):

- **功能和用法：**Edit Fields (Set) 节点用于**设置或修改工作流中数据项的字段值**。这类似于电子表格中查找和修改单元格的操作，AI可以通过此类节点修改数据。
- **配置：**指定要编辑的字段名和新的值。

4. 逻辑 (Logic)

逻辑节点用于控制工作流的执行路径和数据流。

• If (If):

- **功能和用法：**If 节点用于**条件判断**，根据设定的条件将工作流分支到不同的路径。
- **配置：**设置条件表达式，当条件为真时执行一个分支，为假时执行另一个分支。

• 循环项 (Loop Over Items):

- **功能和用法：**Loop Over Items (Split in Batches) 节点用于**遍历数据集合**，对每个数据项执行重复操作。
- **配置：**配置循环的批处理大小或遍历方式。

• 切换 (Switch):

- **功能和用法：**Switch 节点类似于多条件判断，根据输入数据的不同值将工作流路由到多个不同的分支。
- **配置：**为不同的输入值设置对应的输出分支。

• 合并 (Merge):

- **功能和用法：**Merge 节点用于**合并来自不同分支的数据流**，使它们能够在统一的流中继续处理。
- **配置：**选择合并数据的策略。

5. 输出 (Output)

输出节点用于将 workflow 处理后的结果发送到外部服务或存储。

• **发送邮件 (Send Email):**

- **功能和用法:** Send Email 节点用于**发送电子邮件**。您可以配置邮件内容、收件人、主题等。
- **配置:** 需要配置邮件服务提供商的凭据 (如Gmail、Outlook.com、Yahoo) 。

• **响应 Webhook (Respond to Webhook):**

- **功能和用法:** Respond to Webhook 节点用于**向发起Webhook请求的服务发送响应**。这使得n8n可以作为API的后端，对外部请求进行处理并返回数据。
- **配置:** 配置响应的状态码和响应体内容。

• **生成文件到本地 (Generate File to Local):**

- **功能和用法:** 虽然没有直接名为“生成文件到本地”的节点，但“Read/Write Files from Disk”核心节点支持**写入文件到本地磁盘**。这意味着 workflow 可以将处理后的数据保存为本地文件。例如，AI 可以“SaveFile”将修改后的文件保存回硬盘。
- **配置:** 使用“Read/Write Files from Disk”节点，选择写入操作，并指定文件内容和保存路径。

3.2基本工作流创建

图文教程:

<https://mp.weixin.qq.com/s/olcmTW0Gg2r63C0Tn91r2Q>

视频教程:

https://www.bilibili.com/video/BV1nejbz6Eip/?spm_id_from=333.337.search-card.all.click&vd_source=d9f631f2aa422b95711dea08e6660a60

3.3避坑指南

n8n容易踩坑的环节，同事也是开发稳定性高的n8n工作流的要求。

<https://x.com/bing gandata/status/1951798212995829996>

1 97% 的 n8n 工作流缺乏异常处理，在出错时默默失败。建议：为每个工作流添加一个错误触发器，并强制对关键节点（HTTP/API/DB）进行异常处理+邮件/Slack 通知。

2 320 个 webhooks 未进行身份验证，152 个进程仍在 使用明文 HTTP。有些直接在节点中硬编码键。建议：对所有 webhooks 强制执行身份验证，全程使用 HTTPS，不要硬编码 API 密钥，定期轮换它们，并加密存储。

3 很多流程有无用节点、API 在循环里频繁请求、无效数据转换反复出现。建议：能批量就批量，流程结构清晰，节点合并，并行能并行，复杂逻辑直接用代码节点搞定。

4 35% 的工作流接入了 AI，但大部分没做向量数据库和缓存，token 消耗高。建议：Prompt 要分块，简单用 GPT-3.5，复杂用 GPT-4，批量处理+结果校验，AI 节点也要异常处理和日志监控。

5 74.7% 工作流分类都是 General，文档说明严重不足，协作查错全靠猜。建议：流程统一命名+标签，关键节点加 Sticky Note，协作排查省大把时间。

6 高频套路：数据转换（Set→HTTP）、API 链式（HTTP→HTTP）、条件分流（If/Switch→Set）、数据聚合（Set→Merge）建议：直接套用成熟流程模板，少造轮子。

7 上线前安全清单：所有 webhook 有认证、外部请求全 HTTPS、敏感数据加密、权限分层、异常信息不泄露隐私。建议：每次上线前都照单自查一遍。

8 维护和优化：慢流程拆分为子流程，API 并发要批量/加缓存，老旧节点及时清理。建议：每月安全&性能巡检，流程有改动就同步文档。

🔗 很多流程没监控，出错没人知道，维护成本高。建议：每个关键流程都加集中日志和监控，记录流程名、节点、错误和时间戳，AI/长流程建议加进度日志。

1️⃣ 0 只要你把这些最佳实践都落地了：绝大部分工作流有异常处理、webhook 全认证、HTTPS 全覆盖、无明文密钥、流程分类注释完整，平均执行时间都能压到 10 秒以内！

3.4进阶应用 - 模板套用

n8n的模板套用方式非常直观
首先在任意位置复制整段文本：

在n8n任意位置粘贴：

2000+模板

<https://github.com/enescingoz/awesome-n8n-templates> ✓

2000+模板

<https://github.com/Zie619/n8n-workflows> ✓

3.5进阶应用 - mcp

模型上下文协议（MCP）定义了AI代理与外部工具的标准化通信框架，允许在不同LLM之间无缝切换工具集成逻辑。其架构包含三大组件：

1. 主机（Host）：承载MCP连接的LLM应用（如Claude Desktop、Cursor IDE）
2. 客户端（Client）：作为对接MCP服务器的通信端点
3. 服务器（Server）：为主机提供工具、数据或提示资源的独立服务

这种架构使得n8n能以两种角色接入MCP生态：既可以将工作流作为MCP服务器对外暴露功能，也能作为客户端调用其他MCP服务器的能力。

如何将整个mcp工作流部署为mcp服务

图文教程

<https://zhuanlan.zhihu.com/p/1913630305291567877>

官网介绍：

<https://n8n.io/workflows/3770-build-your-own-n8n-workflows-mcp-server/>

3.6进阶应用 - 自然语言编程

使用官方扩展：n8nchat

<https://n8nchat.com>

部署开源项目

https://www.bilibili.com/video/BV1HuKEzDEaj/?spm_id_from=333.337.search-card.all.click&vd_source=c1396b80a656000a13265d2670a6de6e

3.7进阶应用 - 社区节点

社区节点排行榜 TOP 100

<https://github.com/restyler/awesome-n8n?tab=readme-ov-file>

点击设置

点击社区节点和下载

输入想要安装的社区节点

4. 参考资料与术语表

4.1 参考资料

n8n文档

<https://docs.n8n.io>

中文文档:

<https://docs.n8ncn.io>

官方社区 (模板、社区成员)

<https://community.n8n.io/>

快速开始:

<https://n8n.io/>

界面介绍

<https://n8n.akashio.com/article/n8n-ui>

教学合集:

<https://www.darrelltw.com/n8n-tutorial-resources/#templates>

零基础教程

https://www.bilibili.com/video/BV1nejbz6Eip/?spm_id_from=333.337.search-card.all.click&vd_source=d9f631f2aa422b95711dea08e6660a60

2000+模板

<https://github.com/enescingoz/awesome-n8n-templates>

2000+模板

<https://github.com/Zie619/n8n-workflows>

n8n踩坑环节。

<https://x.com/binggandata/status/1951798212995829996>

使用n8n自动抓取天气预报, 并发送到微信

<https://zhuanlan.zhihu.com/p/629643758>

n8n+mcp+deepseek

https://blog.csdn.net/m0_59235245/article/details/147950398

视频教程

https://www.bilibili.com/video/BV11QX8YNEjU/?spm_id_from=333.337.search-card.all.click&vd_source=d9f631f2aa422b95711dea08e6660a60

mcp节点

<https://github.com/nerding-io/n8n-nodes-mcp>

如何将整个mcp工作流部署为mcp服务

图文教程

<https://zhuanlan.zhihu.com/p/1913630305291567877>

官网介绍:

<https://n8n.io/workflows/3770-build-your-own-n8n-workflows-mcp-server/>

n8nchat (自然语言编程)

<https://n8nchat.com>

飞书表格

<https://mp.weixin.qq.com/s/MIoNKpCSBYhhBjxCAaGSYg>

多智能体工作流

<https://mp.weixin.qq.com/s/efsB3GwZzTVH4rjW5i0iGQ>

论文自动解读知识库

[<https://mp.weixin.qq.com/s/YBoM7YkpHjsFl-hyaSsnOw>]

4.2 术语表

- **n8n**: 一个开源且公平代码许可的工作流自动化工具，允许用户通过可视化界面连接应用程序和 API，以实现数据操作和业务流程自动化。
- **MCP (Model Context Protocol)**: 模型上下文协议，一种标准化协议，用于使 AI 模型能够与外部工具和数据源进行交互，从而扩展其能力。
- **n8n MCP 服务器工作流模板**: 一个预构建的 n8n 工作流，用于将现有的 n8n 工作流转换为一个 MCP 服务器，使得 AI 客户端可以发现、管理和执行这些工作流。
- **AI 代理 (AI Agent)**: 能够理解指令、规划行动、使用工具并与环境交互以实现目标的智能系统，通常由大型语言模型驱动。
- **工作流 (Workflow)**: 在 n8n 中，由一系列相互连接的节点组成的自动化流程，用于执行特定任务或自动化业务逻辑。
- **节点 (Node)**: n8n 工作流的基本组成单元，代表一个特定的操作或与一个应用程序的集成。
- **MCP 触发器 (MCP Trigger)**: n8n MCP 服务器工作流模板中的一个核心节点，用于接收 MCP 客户端的请求并触发相关操作（如发现、管理或执行工作流）。
- **“可用”工作流 (Available Workflows)**: 在 n8n MCP 服务器中，被明确允许 AI 代理访问和使用的 n8n 工作流集合，通常通过标签或其他过滤方式进行管理。
- **Redis**: 一个开源的、内存中的数据结构存储，可用作数据库、缓存和消息代理。在 n8n MCP 服务器模板中，它被用作内存来跟踪“可用”工作流。
- **Subworkflow 触发器 (Subworkflow Trigger)**: n8n 中的一种触发器节点，允许一个工作流作为另一个工作流的子流程被调用和执行，常用于模块化和参数传递。
- **输入模式 (Input Schema)**: 定义工作流或节点预期接收的数据结构和数据类型，确保数据以正确的格式传入。
- **直通方法 (Passthrough Method)**: 在 n8n 中，一种参数传递方式，表示当节点未明确设置参数时，将使用传入的数据作为默认值。
- **API 密钥 (API Key)**: 应用程序编程接口密钥，一种用于验证用户、应用程序或服务的身份并授予其访问 API 权限的凭证。
- **Claude Desktop**: 一种 MCP 客户端或 AI 代理，能够通过 MCP 协议与外部工具和数据源（如 n8n MCP 服务器）进行交互。
- **LangChain**: 一个流行的开源框架，旨在帮助开发者构建由大型语言模型 (LLM) 驱动的应用程序和代理，它将 LLM 相关代码封装为可复用的组件。
- **RAG (Retrieval-Augmented Generation)**: 检索增强生成，一种 AI 技术，通过从外部知识库中检索相关信息来增强大型语言模型的生成能力，使其能够提供更准确和实时的回答。
- **HTTP Request**: 一种 n8n 节点，用于向任意 HTTP/HTTPS 端点发送请求，常用于与未直接集成的第三方服务进行交互。
- **Code 节点 (Code Node)**: n8n 中的一种节点，允许用户在工作流中编写和执行自定义的 JavaScript 或 Python 代码，实现更复杂的逻辑或与特定库交互。
- **私有部署 (Self-hosting)**: 用户在自己的服务器或基础设施上安装和运行软件，而不是使用云服务提供商提供的托管版本，通常出于隐私、安全或定制需求。
- **低代码 (Low-code)**: 一种软件开发方法，通过最少量的手动编码实现应用程序的快速开发，通常依赖于图形用户界面和可视化工具。
- **无代码 (No-code)**: 一种软件开发方法，允许非技术用户通过拖拽界面和预构建组件来创建应用程序和自动化流程，无需编写任何代码。

- **Docker:** 一个用于开发、交付和运行应用程序的开源平台，通过容器化技术使应用程序及其依赖项能够打包在一起，并在任何环境中以相同方式运行。
- **GitHub:** 一个基于 Web 的版本控制和协作平台，用于软件开发，允许开发者存储、管理和共享代码仓库。n8n 的许多 workflow 模板和社区节点都托管在 GitHub 上。

5. n8n迷你Hackathon小组

5.1 职能与支持

- **技术支持:** 提供专业的技术支持，解决用户在使用n8n过程中遇到的问题。
- **项目咨询:** 为用户提供项目规划、设计和实施的咨询服务，确保项目的顺利进行。
- **培训服务:** 定期举办线上或线下的培训课程，帮助用户掌握n8n的使用技巧和最佳实践。
- **社区建设:** 维护和管理社区资源，促进用户之间的交流与合作，共同推动n8n的发展。

5.2 加入我们的方式

扫码入群聊：