**Title: Comprehensive Guide to Installing n8n via Different Methods**

## 1. Installing n8n Using npm (Node.js)

### Prerequisites:

* Node.js >= 18.x
* npm (comes with Node.js)

### Steps:

1. Install n8n globally:

* npm install n8n -g

1. Start n8n:

* n8n

1. Access n8n via browser at:

* http://localhost:5678

## 2. Installing n8n Using Docker

### Prerequisites:

* Docker installed
* Optional: Docker Compose

### Steps:

**Using Docker CLI**:

docker run -it --rm \  
 -p 5678:5678 \  
 -v ~/.n8n:/home/node/.n8n \  
 n8nio/n8n

**Using Docker Compose**: Create a docker-compose.yml file:

version: '3'  
services:  
 n8n:  
 image: n8nio/n8n  
 ports:  
 - 5678:5678  
 volumes:  
 - ~/.n8n:/home/node/.n8n

Then run:

docker-compose up -d

Access n8n at:

http://localhost:5678

## 3. Installing n8n Using npx (Temporary Run)

### Prerequisites:

* Node.js
* npm

### Steps:

npx n8n

This runs n8n temporarily without global installation.

## 4. Installing n8n on a Cloud VM (e.g., AWS, GCP, Azure)

### Prerequisites:

* VM instance with Linux OS
* SSH access

### Steps:

1. SSH into the VM:

* ssh user@your-server-ip

1. Install Node.js and npm (using NodeSource):

* curl -fsSL https://deb.nodesource.com/setup\_18.x | sudo -E bash -  
  sudo apt-get install -y nodejs

1. Install n8n globally:

* npm install n8n -g

1. Run n8n:

* n8n

Use tmux, screen, or set up pm2 to run it in the background.

## 5. Optional: Setup for Production

* Use reverse proxy (NGINX)
* Set up HTTPS with SSL (Let’s Encrypt)
* Set environment variables for authentication and credentials encryption
* Use database like Postgres for persistence

## Feasibility: Local vs Cloud Setup

### Local Setup

**Pros:**

* Fast to prototype and test workflows
* No hosting cost
* Easier access to local resources (files, databases)

**Cons:**

* Not accessible from the internet without tunneling (e.g., ngrok)
* Resource limitations based on your machine
* Manual handling of background execution, uptime

Note: If n8n is deployed as a Docker container inside a Kubernetes cluster, it can be accessed from other pods within the same cluster using internal networking (e.g., ClusterIP service or DNS-based service discovery). Internet tunneling is only required for standalone local setups.

### Cloud Setup

**Pros:**

* Publicly accessible with static IP or domain
* Easily integrated with cloud services and APIs
* Better scalability and reliability

**Cons:**

* Requires SSH and server management skills
* Hosting cost involved
* More setup time for security, persistence, and HTTPS

## GPU Access (Advanced)

* n8n does **not require GPU** by default
* Use --gpus all with Docker if running custom GPU workloads (like ML inference in Python nodes)

## References:

* <https://docs.n8n.io>
* <https://hub.docker.com/r/n8nio/n8n>

End of Document