



[Get ALL AI Automation + Agent Templates >>](#)

Prompts & Links - I Built an AI Agent That Consumes 100 Hours Content For You (n8n + No-Code)

(Follow the instructions in this video: <https://youtu.be/dF1e3nuiSjU>)

Templates:

PRO Version "AI Mentor" Template for Instant Setup:

<https://fabimarkl.com/automation-templates/#ai-mentor>

Links:

Free Trial for N8N Account:

<https://n8n.io/>

YouTube Video Scraper:

<https://rapidapi.com/rahilkhan224/api/youtube-video-summarizer-gpt-ai>

Create Supabase Account:

<https://supabase.com/>

Create Firecrawl Account:

<https://www.firecrawl.dev/>

Create OpenAI Account:

<https://platform.openai.com/>

Cost Breakdown:

<https://claude.ai/public/artifacts/83b119fc-a83a-4aa7-8dfc-7420a0dc780d>

**n8n links are affiliate links that will give me a small percentage of every sale, but have no effect on the price you pay or their free trial.*

[Click Here to Get AI Automation Template for Instant Setup >>](#)

Supabase Setup Guide for AI Mentor System

Part 1: Create Your Supabase Project

Step 1: Go to supabase.com and click "Start your project"

Sign up with GitHub or email.

Step 2: Click "New Project" and fill in:

- **Name:** `knowlEdge-base` (or whatever you want)
- **Password:** Create one and save it somewhere safe
- **Region:** Pick the closest to you

Click "Create new project" and wait about 2 minutes.

Part 2: Create the Database Table

Step 3: In the left sidebar, click "SQL Editor"

Step 4: You'll see a blank text box. Copy and paste this entire block:

```
create extension if not exists vector;

create table knowledge_chunks (

  id bigserial primary key,

  content text,

  metadata jsonb,

  embedding vector(1536)

);
```

```
create index on knowledge_chunks using ivfflat (embedding vector_cosine_ops)
WITH (lists = 100);
```

Step 5: Click the green "Run" button (bottom right)

You should see "Success" messages.

Part 3: Add the Search Function

Step 6: Stay in SQL Editor. Clear the text box and paste this block:

sql

```
create or replace function match_documents (
  query_embedding vector(1536),
  match_count int default 5,
  filter jsonb default '{}'
)
returns table (
  id bigint,
  content text,
  metadata jsonb,
  similarity float
)
language plpgsql
as $$
begin
  return query
```

```
select
  knowledge_chunks.id,
  knowledge_chunks.content,
  knowledge_chunks.metadata,
  1 - (knowledge_chunks.embedding <=> query_embedding) as similarity
from knowledge_chunks
where knowledge_chunks.metadata @> filter
order by knowledge_chunks.embedding <=> query_embedding
limit match_count;
end;
$$;
```

Step 7: Click the green "Run" button

You should see "Success" again.

Part 4: Get Your API Keys

Step 8: In the left sidebar, click the gear icon (Project Settings)

Step 9: Click "Data API"

Step 10: Copy and save these two things:

- **Project URL** – looks like <https://something.supabase.co>

Step 11: Click "API Keys" then on "Legacy anon, service_role API keys" and copy the service_role secret.

Part 5: Connect Supabase to n8n

Step 12: In n8n, go to **Settings** → **Credentials** → **Add Credential**

Step 13: Search for "Supabase" and click it

Step 14: Fill in:

- **Host:** Paste your Project URL
- **Service Role Key:** Paste your service_role key

Step 15: Click Save

Done!

Your Supabase database is ready. Now n8n can store and search your mentor content.

Split:

```
const input = $input.item.json['Enter Link to YouTube Videos and
Blog Articles'];
const urlList = input.split('\n').filter(url => url.trim() !==
'');

// Return array of URLs
return urlList.map(url => ({
  json: { url: url.trim() }
}));
```

Identifier:

```
// Detect URL type (processes ONE URL at a time)
const url = $input.item.json.url.trim();
let type = 'unknown';

// YouTube detection
if (url.includes('youtube.com') || url.includes('youtu.be')) {
  type = 'youtube';
}
// Apple Podcasts detection
else if (url.includes('podcasts.apple.com') ||
url.includes('itunes.apple.com')) {
  type = 'apple_podcast';
}
// Spotify Podcasts (blocked but detect anyway)
else if (url.includes('spotify.com/episode') ||
url.includes('spotify.com/show')) {
  type = 'spotify_podcast';
}
// Everything else = article/blog
else if (url.startsWith('http://') ||
url.startsWith('https://')) {
  type = 'article';
}

return {
  json: {
    url: url,
    type: type
  }
};
```

Get Video ID:

```
const url = item.json.url;

let videoId = null;

const m1 = url.match(/[?&]v=([^\&]+)/);
if (m1) videoId = m1[1];

if (!videoId) {
  const m2 = url.match(/youtu\.be\/([^\?]+)/);
  if (m2) videoId = m2[1];
}

return {
  json: {
    videoId
  }
};
```

Get Video Data:

URL:

<https://youtube-video-summarizer-gpt-ai.p.rapidapi.com/api/v1/get-transcript-v2>

Query Parameters:

video_id
{{ \$json.videoId }}

platform
youtube

Header Parameters

Name
X-Rapidapi-Key
YOUR_API_KEY

X-Rapidapi-Host

```
youtube-video-summarizer-gpt-ai.p.rapidapi.com
```

```
Accept  
application/json
```

Clean Text:

```
// Access the nested transcript array  
const transcript = item.json.data.transcripts.en_auto.custom;  
  
if (!Array.isArray(transcript)) {  
  throw new Error('Transcript array not found at  
data.transcripts.en_auto.custom');  
}  
  
// Join all text segments into one full string  
const fullText = transcript  
  .map(seg => seg.text || "")  
  .join(" ")  
  .replace(/&#39;/g, "")  
  .replace(/&/g, "&");  
  
// The new Code node returns a single object, not an array  
return {  
  json: {  
    fullText  
  }  
};
```

Scrape Website Content:

<https://api.firecrawl.dev/v1/scrape>

Header Parameters:

```
Authorization  
Bearer YOUR_API_KEY
```

```
---
```

```
{  
  "url": "{{ $json.url }}",  
  "formats": [  

```



```
"markdown",  
  "html"  
]  
}
```

AI Agent:

System Prompt:

You are a helpful assistant with access to a knowledge base.

IMPORTANT: For EVERY user question, you MUST use the knowledge_base tool to search for relevant information before answering. Never answer from your own knowledge - always search first.

After searching, answer based on what you find. If the search returns no relevant results, tell the user you couldn't find information about that topic in the knowledge base.

Supabase Tool Description:

Search through YouTube videos, and articles to find relevant information