

GETTING STARTED

- Overview
- Quickstart
- Models
- OpenAI Compatibility
- Rate Limits
- Templates
- API Reference
- CORE FEATURES
- Text Generation
- Speech to Text
- Text to Speech
- OCR and Image Recognition
- Reasoning
- Content Moderation
- Structured Outputs
- Prompt Caching

TOOLS & INTEGRATIONS

- Tool Use
- Integrations Catalog

COMPOUND (AGENTIC AI)

- Overview
- Systems
- Use Cases

GUIDES

- Prompting Guide
- Assistant Message Prefilling

SERVICE TIERS

- Service Tiers
- Performance Tier
- Flex Processing
- Batch Processing

ADVANCED

- LoRA Inference

PRODUCTION READINESS

- Production Checklist
- Optimizing Latency
- Security Onboarding
- Prometheus Metrics

ACCOUNT AND CONSOLE

- Spend Limits
- Projects
- Model Permissions

Quickstart

Get up and running with the Groq API in a few minutes, with the steps below.

For additional support, catch our [onboarding video](#).

Create an API Key

Please visit [here](#) to create an API Key.

Set up your API Key (recommended)

Configure your API key as an environment variable. This approach streamlines your API usage by eliminating the need to include your API key in each request. Moreover, it enhances security by minimizing the risk of inadvertently including your API key in your codebase.

In your terminal of choice:

shell

export GROQ\_API\_KEY=<your-api-key-here>

Requesting your first chat completion

curl

JavaScript

Python

JSON

Install the Groq Python library:

shell

pip install groq

Performing a Chat Completion:

Python

1 import os  
2  
3 from groq import Groq  
4  
5 client = Groq(  
6 api\_key=os.environ.get("GROQ\_API\_KEY"),  
7 )  
8  
9 chat\_completion = client.chat.completions.create(  
10 messages=[  
11 {  
12 "role": "user",  
13 "content": "Explain the importance of fast language models",  
14 }  
15 ],  
16 model="llama-3.3-70b-versatile",  
17 )  
18  
19 print(chat\_completion.choices[0].message.content)

Using third-party libraries and SDKs

Vercel AI SDK

LiteLLM

LangChain

Using AI SDK:

On this page

- Create an API Key
- Set up your API Key (recommended)
- Requesting your first chat completion
- Using third-party libraries and SDKs

AI SDK is a Javascript-based open-source library that simplifies building large language model (LLM) applications. Documentation for how to use Groq on the AI SDK can be found here.

First, install the `ai` package and the Groq provider `@ai-sdk/groq` :

shell

```
pnpm add ai @ai-sdk/groq
```

Then, you can use the Groq provider to generate text. By default, the provider will look for `GROQ_API_KEY` as the API key.

JavaScript

```
1 import { groq } from '@ai-sdk/groq';
2 import { generateText } from 'ai';
3
4 const { text } = await generateText({
5   model: groq('llama-3.3-70b-versatile'),
6   prompt: 'Write a vegetarian lasagna recipe for 4 people.',
7 });
```

Now that you have successfully received a chat completion, you can try out the other endpoints in the API.

Next Steps

- Check out the [Playground](#) to try out the Groq API in your browser
- Join our GroqCloud [developer community](#)
- Add a how-to on your project to the [Groq API Cookbook](#)

Was this page helpful?



PDF-файл был скачан. Оцените нас: ☆☆☆☆☆

