

Error codes

This guide includes an overview on error codes you might see from both the [API](#) and our [official Python library](#). Each error code mentioned in the overview has a dedicated section with further guidance.

API errors

CODE	OVERVIEW
401 - Invalid Authentication	Cause: Invalid Authentication Solution: Ensure the correct API key and requesting organization are being used.
401 - Incorrect API key provided	Cause: The requesting API key is not correct. Solution: Ensure the API key used is correct, clear your browser cache, or generate a new one .
401 - You must be a member of an organization to use the API	Cause: Your account is not part of an organization. Solution: Contact us to get added to a new organization or ask your organization manager to invite you to an organization .
429 - Rate limit reached for requests	Cause: You are sending requests too quickly. Solution: Pace your requests. Read the Rate limit guide .
429 - You exceeded your current quota, please check your plan and billing details	Cause: You have hit your maximum monthly spend (hard limit) which you can view in the account billing section . Solution: Apply for a quota increase .
500 - The server had an error while processing your request	Cause: Issue on our servers. Solution: Retry your request after a brief wait and contact us if the issue persists. Check the status page .
503 - The engine is currently overloaded, please try again later	Cause: Our servers are experiencing high traffic. Solution: Please retry your requests after a brief wait.

> 401 - Invalid Authentication

> 401 - Incorrect API key provided

> 401 - You must be a member of an organization to use the API

> 429 - Rate limit reached for requests

> 429 - You exceeded your current quota, please check your plan and billing details

> 503 - The engine is currently overloaded, please try again later

Python library error types

TYPE	OVERVIEW
APIError	Cause: Issue on our side. Solution: Retry your request after a brief wait and contact us if the issue persists.
Timeout	Cause: Request timed out. Solution: Retry your request after a brief wait and contact us if the issue persists.
RateLimitError	Cause: You have hit your assigned rate limit. Solution: Pace your requests. Read more in our Rate limit guide .
APIConnectionError	Cause: Issue connecting to our services. Solution: Check your network settings, proxy configuration, SSL certificates, or firewall rules.

TYPE	OVERVIEW
InvalidRequestError	Cause: Your request was malformed or missing some required parameters, such as a token or an input. Solution: The error message should advise you on the specific error made. Check the documentation for the specific API method you are calling and make sure you are sending valid and complete parameters. You may also need to check the encoding, format, or size of your request data.
AuthenticationError	Cause: Your API key or token was invalid, expired, or revoked. Solution: Check your API key or token and make sure it is correct and active. You may need to generate a new one from your account dashboard.
ServiceUnavailableError	Cause: Issue on our servers. Solution: Retry your request after a brief wait and contact us if the issue persists. Check the status page .

> APIError

> Timeout

> RateLimitError

> APIConnectionError

> InvalidRequestError

> AuthenticationError

> ServiceUnavailableError

Persistent errors

If the issue persists, [contact our support team via chat](#) and provide them with the following information:

- The model you were using
- The error message and code you received
- The request data and headers you sent
- The timestamp and timezone of your request
- Any other relevant details that may help us diagnose the issue

Our support team will investigate the issue and get back to you as soon as possible. Note that our support queue times may be long due to high demand. You can also [post in our Community Forum](#) but be sure to omit any sensitive information.

Handling errors

We advise you to programmatically handle errors returned by the API. To do so, you may want to use a code snippet like below:

```
1  try:
2      #Make your OpenAI API request here
3      response = openai.Completion.create(prompt="Hello world",
4                                          model="gpt-3.5-turbo-instruct")
5  except openai.error.APIError as e:
6      #Handle API error here, e.g. retry or log
7      print(f"OpenAI API returned an API Error: {e}")
8      pass
9  except openai.error.APIConnectionError as e:
10     #Handle connection error here
11     print(f"Failed to connect to OpenAI API: {e}")
12     pass
13 except openai.error.RateLimitError as e:
14     #Handle rate limit error (we recommend using exponential backoff)
```

```
15     print(f"OpenAI API request exceeded rate limit: {e}")
16     pass
```