

# ChatGPT Enterprise

<https://openai.com/blog/introducing-chatgpt-enterprise>

ChatGPT Enterprise is available today, it offers:

- **Enterprise-grade security and privacy**

Customer prompts and company data are not used for training OpenAI models.

Data encryption at rest (AES 256) and in transit (TLS 1.2+)

Certified SOC 2 compliant

- **Features for large-scale deployments**

Admin console with bulk member management

SSO

Domain verification

Analytics dashboard for usage insights

- **The most powerful version of ChatGPT yet**

Unlimited access to GPT-4 (no usage caps)

Higher-speed performance for GPT-4 (up to 2x faster)

Unlimited access to advanced data analysis (formerly known as Code Interpreter)

32k token context windows for 4x longer inputs, files, or follow-ups

Shareable chat templates for your company to collaborate and build common workflows

Free credits to use our **APIs** if you need to extend OpenAI into a fully **custom solution** for your org

# OpenAI API

<https://openai.com/blog/openai-api>

Given any text prompt, the API will return a text completion, attempting to match the pattern you gave it.

- **(Few shot)** You can “program” it by showing it just a few examples of what you’d like it to do; its success generally varies depending on how complex the task is.

- **(Fine-tuning)** The API also allows you to hone performance on specific tasks by training on a dataset (small or large) of examples you provide, or by learning from human feedback provided by users or labelers.

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

## Few shot

The three settings we explore for in-context learning

### Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 cheese => ..... ← prompt
```

### One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 sea otter => loutre de mer ← example
3 cheese => ..... ← prompt
```

### Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 sea otter => loutre de mer ← examples
3 peppermint => menthe poivrée ←
4 plush girafe => girafe peluche ←
5 cheese => ..... ← prompt
```

Traditional fine-tuning (not used for GPT-3)

### Fine-tuning

The model is trained via repeated gradient updates using a large corpus of example tasks.



## Fine-tuning

<https://platform.openai.com/docs/guides/fine-tuning>

At a high level, fine-tuning involves the following steps:

1. Prepare and upload training data
2. Train a new fine-tuned model

```
openai.FineTuningJob.create(training_file="file-abc123",
model="gpt-3.5-turbo")
```

### 3. Use your fine-tuned model

```
completion = openai.ChatCompletion.create(
model="ft:gpt-3.5-turbo:my-org:custom_suffix:id",
messages=[
{"role": "system", "content": "You are a helpful assistant."},
{"role": "user", "content": "Hello!"}
]
)
print(completion.choices[0].message)
```

Fine-tuning models

Create your own custom models by fine-tuning our base models with your training data. Once you fine-tune a model, you'll be billed only for the tokens you use in requests to that model.

[Learn about fine-tuning ↗](#)

Model	Training	Input usage	Output usage
babbage-002	\$0.0004 / 1K tokens	\$0.0016 / 1K tokens	\$0.0016 / 1K tokens
davinci-002	\$0.0060 / 1K tokens	\$0.0120 / 1K tokens	\$0.0120 / 1K tokens
GPT-3.5 Turbo	\$0.0080 / 1K tokens	\$0.0120 / 1K tokens	\$0.0160 / 1K tokens

**messages** array **Required**

A list of messages comprising the conversation so far. [Example Python code.](#)

— Hide Properties

**role** string **Required**

The role of the messages author. One of `system`, `user`, `assistant`, or `function`.

**content** string or null **Required**

The contents of the message. `content` is required for all messages, and may be null for assistant messages with function calls.

**name** string **Optional**

The name of the author of this message. `name` is required if role is `function`, and it should be the name of the function whose response is in the `content`. May contain a-z, A-Z, 0-9, and underscores, with a maximum length of 64 characters.

**function\_call** object **Optional**

The name and arguments of a function that should be called, as generated by the model.

— Hide Properties

**name** string **Required**

The name of the function to call.

**arguments** string **Required**

The arguments to call the function with, as generated by the model in JSON format. Note that the model does not always generate valid JSON, and may hallucinate parameters not defined by your function schema. Validate the arguments in your code before calling your function.

```
{"messages": [{"role": "system", "content": "Marv is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "What's the capital of France?"}, {"role": "assistant", "content": "Paris, as if everyone doesn't know that already."}]}

{"messages": [{"role": "system", "content": "Marv is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "Who wrote 'Romeo and Juliet?'"}, {"role": "assistant", "content": "Oh, just some guy named William Shakespeare. Ever heard of him?"}]}

{"messages": [{"role": "system", "content": "Marv is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "How far is the Moon from Earth?"}, {"role": "assistant", "content": "Around 384,400 kilometers. Give or take a few, like that really matters."}]}
```

## Plan

1. Try few-shot (citi's policy?)
2. Prepare data for fine-tuning with GPT3.5
3. Fine-tuning based on BERT<sub>BASE</sub> (1 hour on a single TPU, a few hours on a GPU)
  - (1) Language modeling
  - (2) Multi Objectives

## Risk

1. Low calculation resource
2. Low quality data



hi



Hello! How can I assist you today?



Suppose you are a quality analyst in the financial information technology department of an investment bank, and your business area is Derivatives. Can you write a cucumber feature about trade booking process, and prepare some data?



Certainly! Below is an example of a Cucumber feature file for the trade booking process in the Derivatives department of an investment bank. I'll also provide some sample data to go along with it. Please note that this is a simplified example, and you should adapt it to your specific business processes and requirements.



Feature: Trade Booking Process for Derivatives

Background:

Given the following preconditions

Account	Account Type	Available Balance
ABC123	Investment	1000000
XYZ456	Investment	500000
DerivativeFund	Derivative	2000000

Scenario: Booking a Derivative Trade

When a trader submits a new trade request with the following details:

Trade ID	Account	Instrument	Quantity	Price
T12345	ABC123	Options	1000	50.00

Then the trade request should be in "Pending" status

And the trade details should be validated successfully

#### Scenario: Approving a Derivative Trade

Given there is a trade request with the following details:

Trade ID	Account	Instrument	Quantity	Price	Status
T12345	ABC123	Options	1000	50.00	Pending

When the trade is approved by the risk manager

Then the trade status should be updated to "Approved"

And the trade should be booked successfully

#### Scenario: Rejection of a Derivative Trade

Given there is a trade request with the following details:

Trade ID	Account	Instrument	Quantity	Price	Status
T67890	XYZ456	Futures	500	60.00	Pending

When the trade is rejected by the risk manager

Then the trade status should be updated to "Rejected"

And the trade should not be booked

#### Scenario: Booking a Trade Exceeding Available Balance

When a trader submits a new trade request with the following details:

Trade ID	Account	Instrument	Quantity	Price
T54321	DerivativeFund	Swaps	3000	40.00

Then the trade request should be in "Pending" status

And the trade should not be booked

And an error message should indicate insufficient available balance

#### Scenario: Invalid Trade Request

When a trader submits an invalid trade request with missing details

Then the trade request should be rejected

And an error message should indicate the missing information