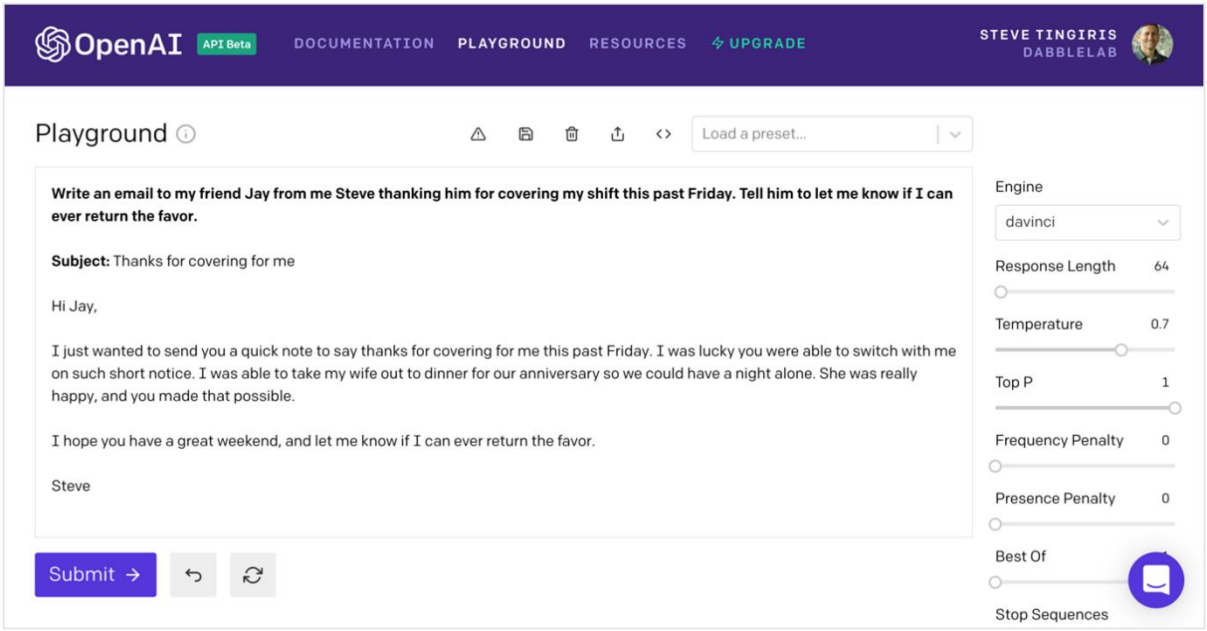
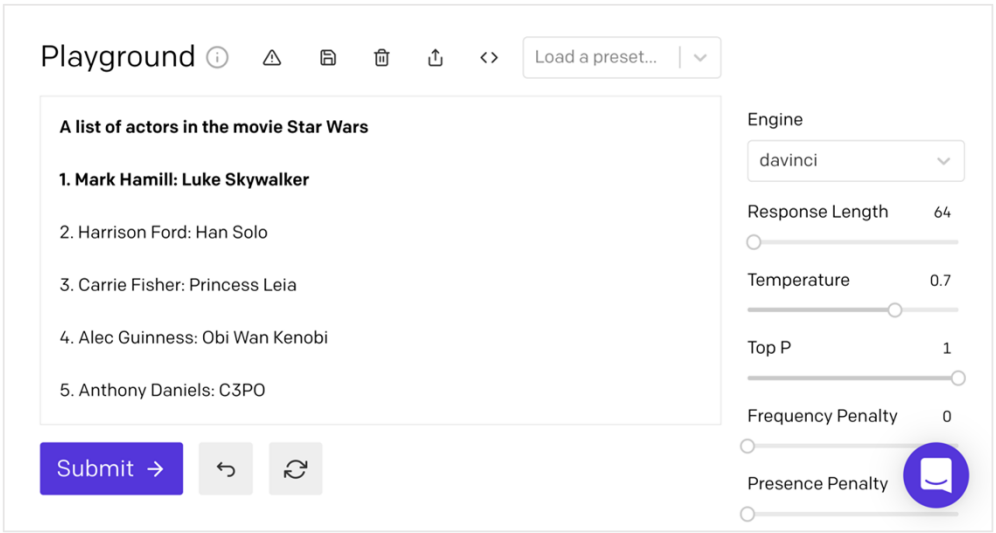
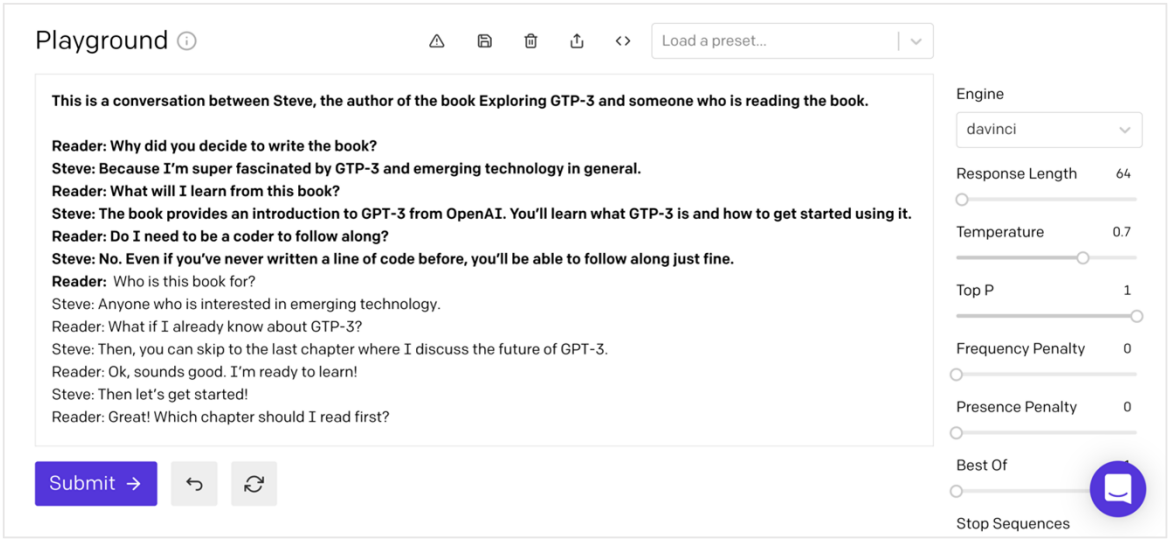
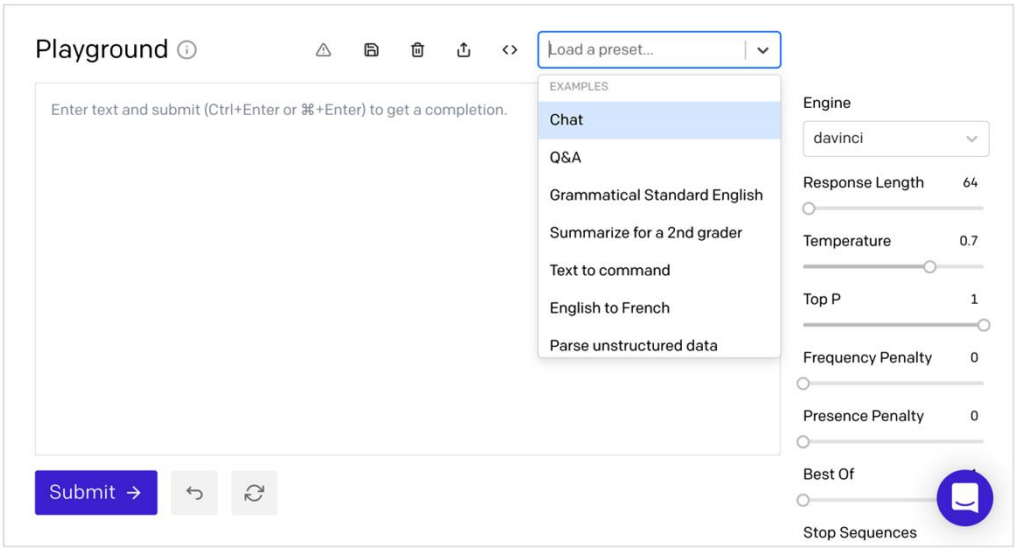
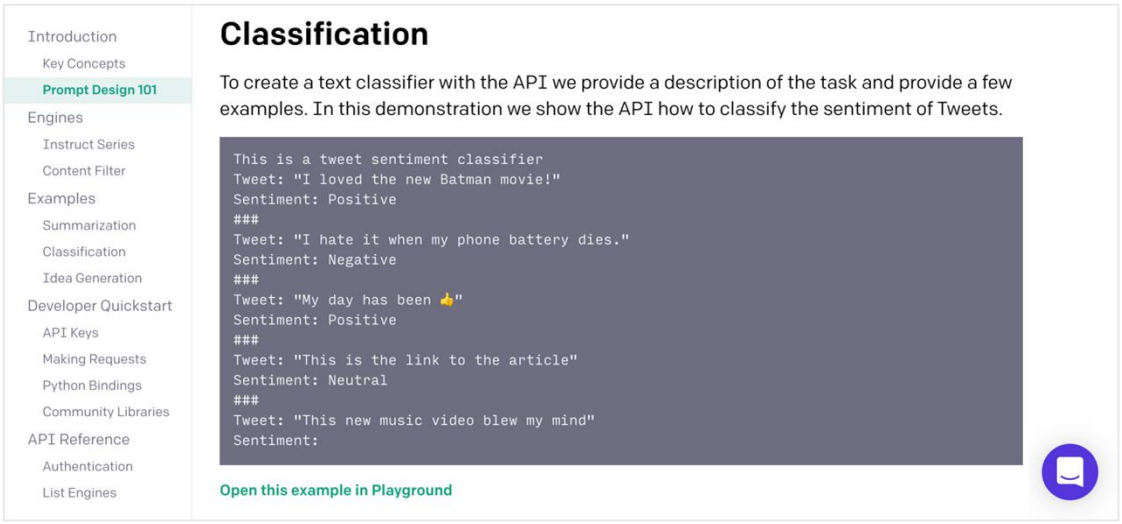
Chapter 1: Introducing GPT-3 and the OpenAI API






Playground
Load a preset...
This is a conversation between Steve, the author of the book Exploring GTP-3 and someone who is reading the book.
Reader: Why did you decide to write the book?
Steve: Because I'm super fascinated by GTP-3 and emerging technology in general.
Reader: What will I learn from this book?
Steve: The book provides an introduction to GPT-3 from OpenAI. You'll learn what GTP-3 is and how to get started using it.
Reader: Do I need to be a coder to follow along?
Steve: No. Even if you've never written a line of code before, you'll be able to follow along just fine.
Reader: Who is this book for?
Steve: Anyone who is interested in emerging technology.
Reader: What if I already know about GTP-3?
Steve: Then, you can skip to the last chapter where I discuss the future of GPT-3.
Reader: Ok, sounds good. I'm ready to learn!
Steve: Then let's get started!
Reader: Great! Which chapter should I read first?
Submit →
Engine
davinci
Response Length 64
Temperature 0.7
Top P 1
Frequency Penalty 0
Presence Penalty 0
Best Of
Stop Sequences


Playground
Load a preset...
EXAMPLES
Chat
Q&A
Grammatical Standard English
Summarize for a 2nd grader
Text to command
English to French
Parse unstructured data
Enter text and submit (Ctrl+Enter or ⌘+Enter) to get a completion.
Submit →
Engine
davinci
Response Length 64
Temperature 0.7
Top P 1
Frequency Penalty 0
Presence Penalty 0
Best Of
Stop Sequences


Introduction
Key Concepts
Prompt Design 101
Engines
Instruct Series
Content Filter
Examples
Summarization
Classification
Idea Generation
Developer Quickstart
API Keys
Making Requests
Python Bindings
Community Libraries
API Reference
Authentication
List Engines
Classification
To create a text classifier with the API we provide a description of the task and provide a few examples. In this demonstration we show the API how to classify the sentiment of Tweets.
This is a tweet sentiment classifier
Tweet: "I loved the new Batman movie!"
Sentiment: Positive
###
Tweet: "I hate it when my phone battery dies."
Sentiment: Negative
###
Tweet: "My day has been 👍"
Sentiment: Positive
###
Tweet: "This is the link to the article"
Sentiment: Neutral
###
Tweet: "This new music video blew my mind"
Sentiment:
Open this example in Playground

ORGANIZATION

dabblelab

Settings

Usage

Members

Billing

USER

API Keys

Usage

Below you'll find a summary of API usage for all users in your organization. All dates and times are UTC-based, and data may be delayed up to 5 minutes.

Please note that monetary values shown are based on our per-engine pricing rates, and may not reflect any credits or discounts associated with your account.

< December >

Tokens | $ | Daily | Cumulative

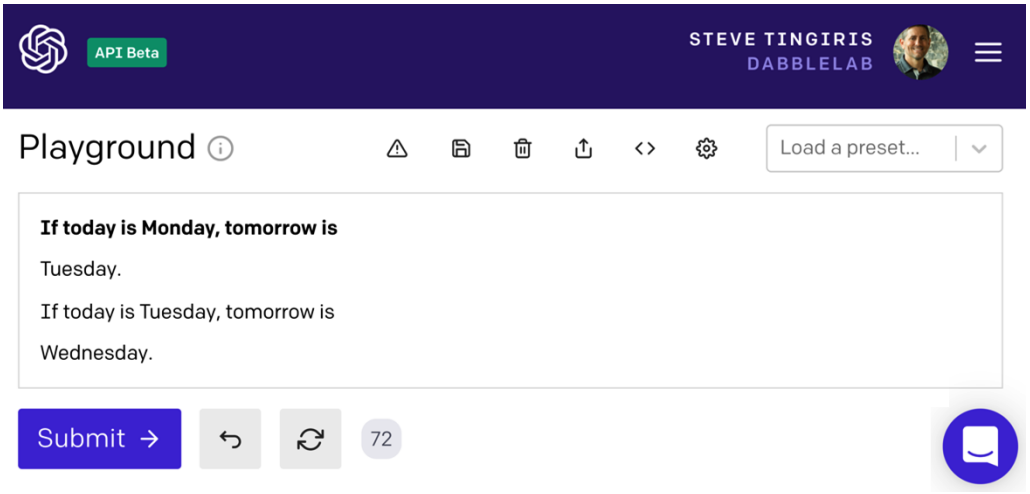
FREE TRIAL USAGE

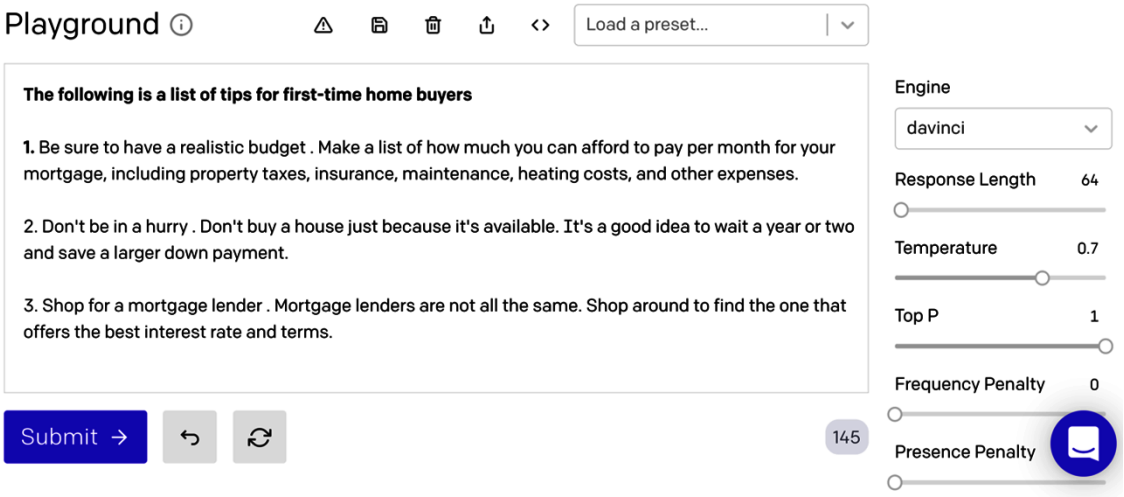
132,694 / 300,000

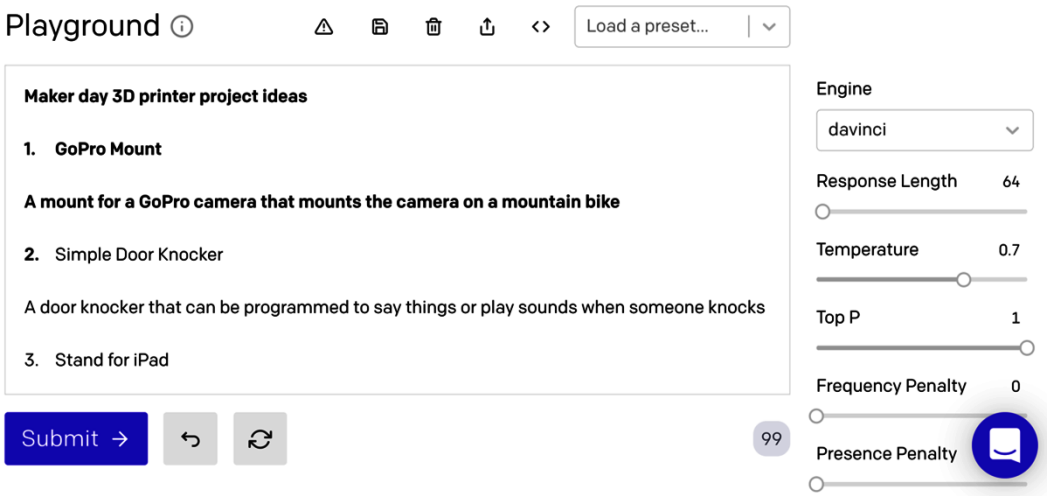
REQUEST BREAKDOWN (UTC)

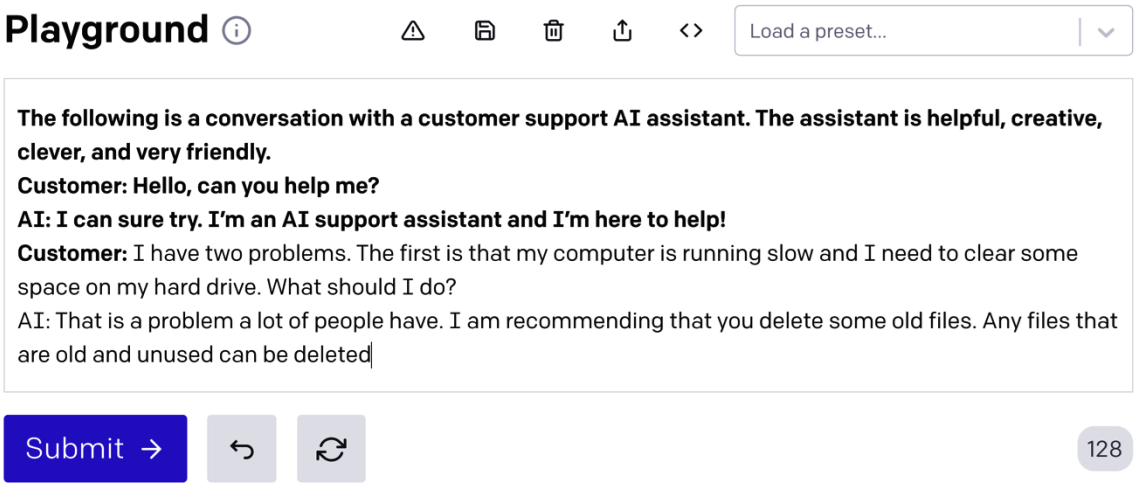
December 1

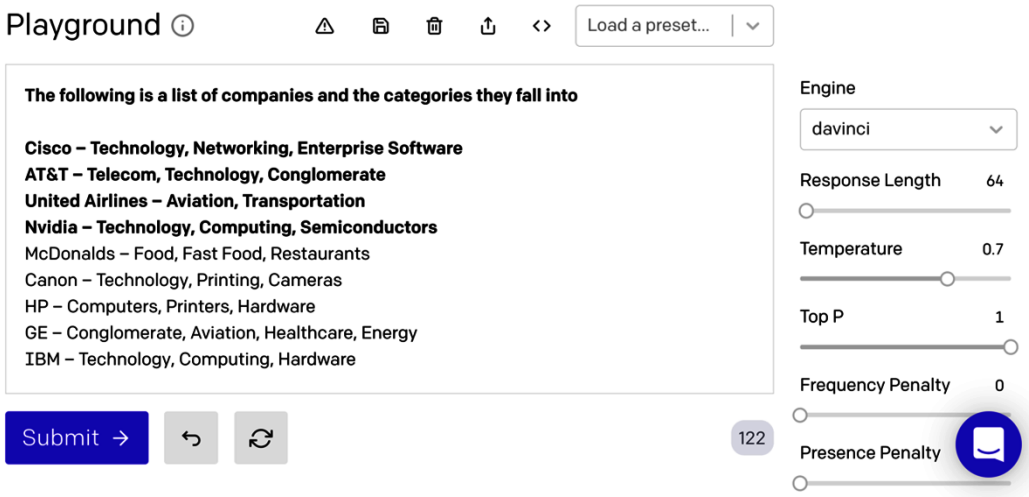
Chapter 2: GPT-3 Applications and Use Cases

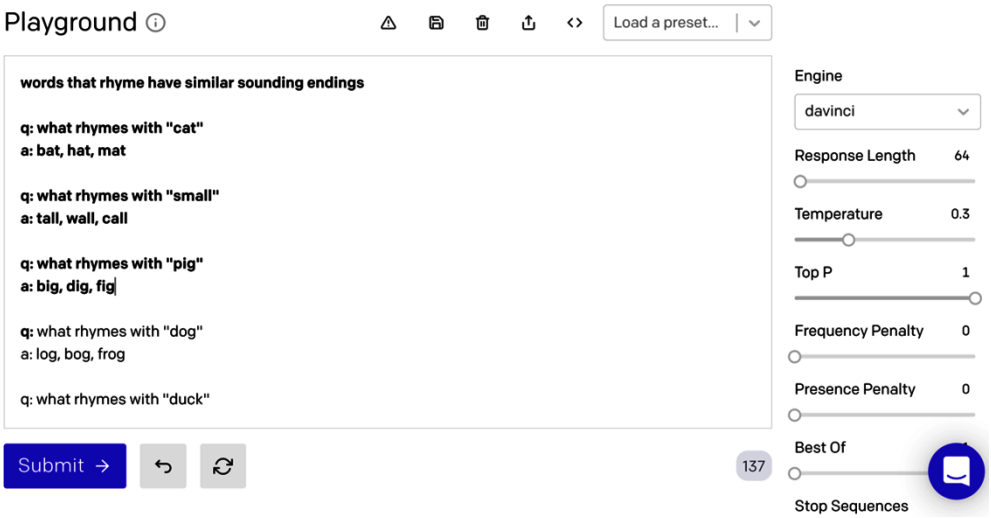


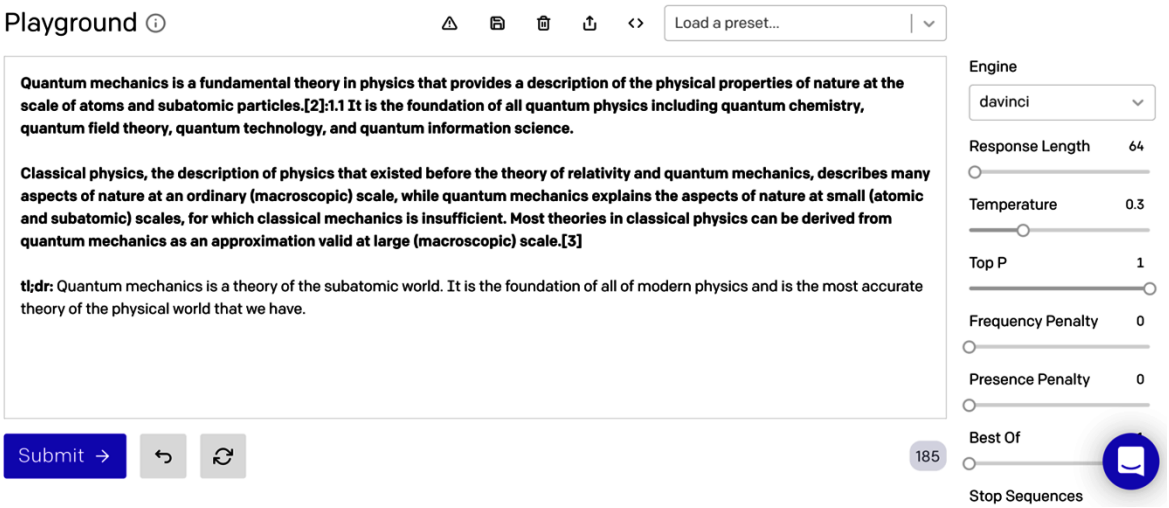


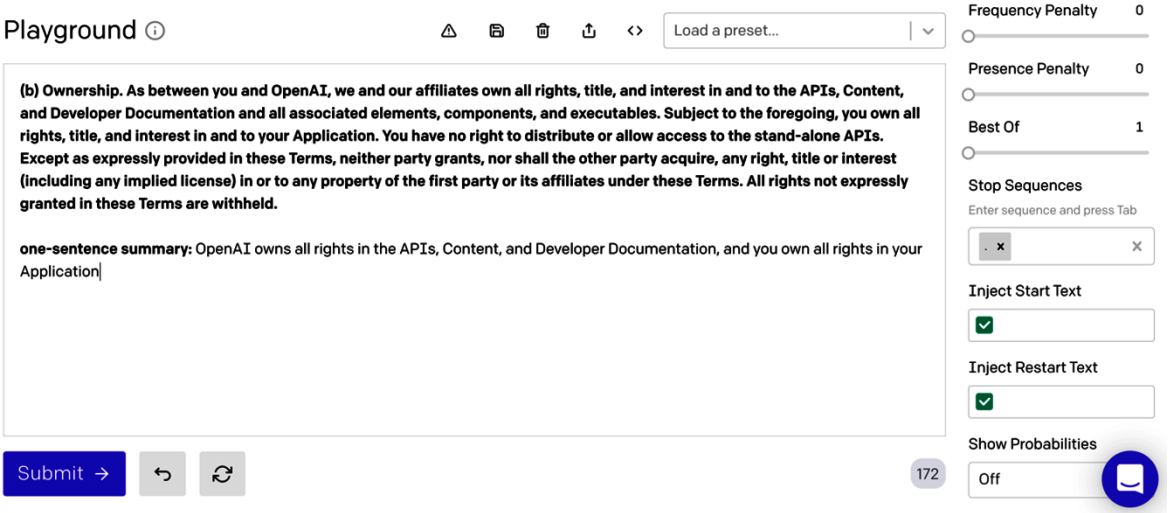


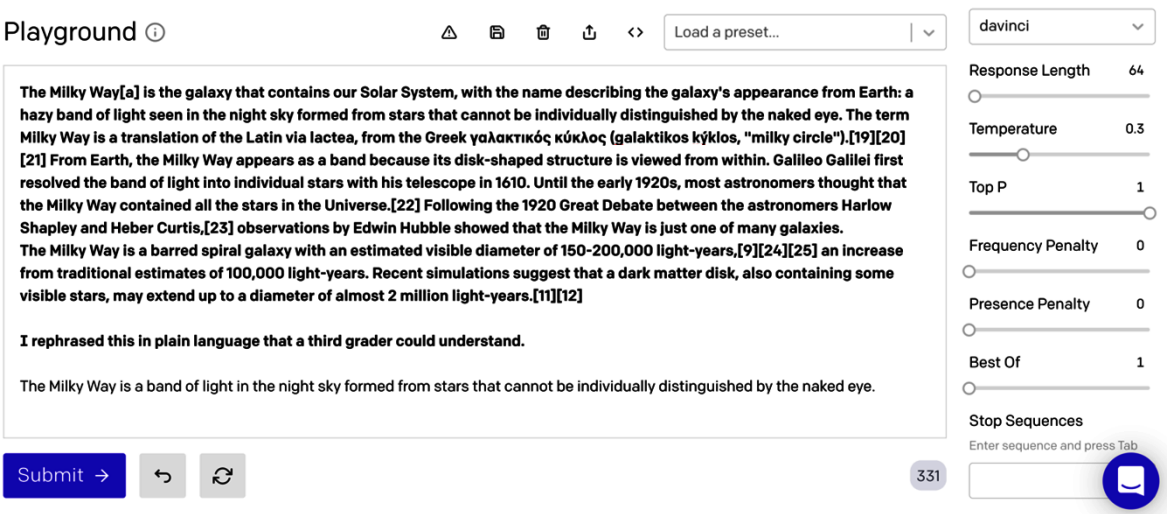

Playground
Load a preset...
The following is a conversation with a customer support AI assistant. The assistant is helpful, creative, clever, and very friendly.
Customer: Hello, can you help me?
AI: I can sure try. I'm an AI support assistant and I'm here to help!
Customer: I have two problems. The first is that my computer is running slow and I need to clear some space on my hard drive. What should I do?
AI: That is a problem a lot of people have. I am recommending that you delete some old files. Any files that are old and unused can be deleted
Submit
128

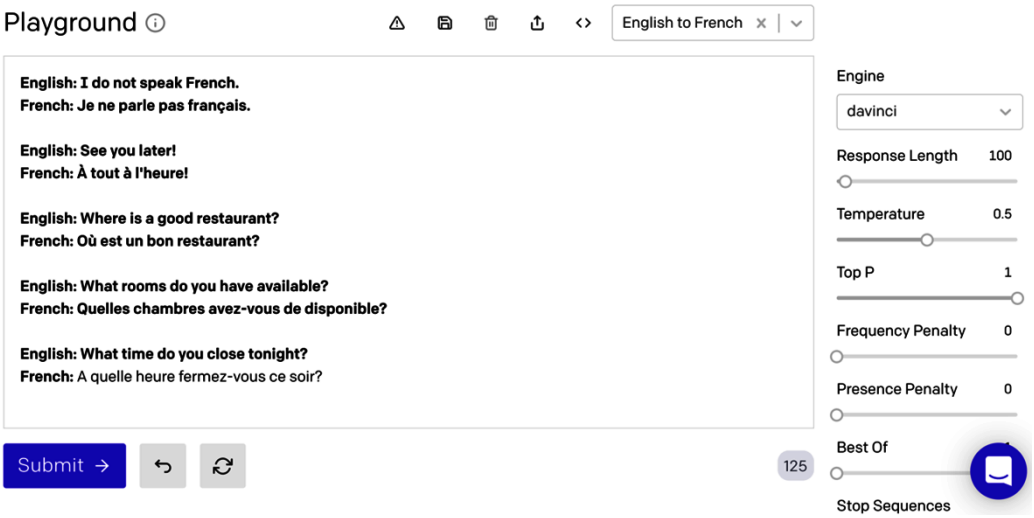

Playground
Load a preset...
The following is a list of companies and the categories they fall into
Cisco – Technology, Networking, Enterprise Software
AT&T – Telecom, Technology, Conglomerate
United Airlines – Aviation, Transportation
Nvidia – Technology, Computing, Semiconductors
McDonalds – Food, Fast Food, Restaurants
Canon – Technology, Printing, Cameras
HP – Computers, Printers, Hardware
GE – Conglomerate, Aviation, Healthcare, Energy
IBM – Technology, Computing, Hardware
Submit
122
Engine
davinci
Response Length 64
Temperature 0.7
Top P 1
Frequency Penalty 0
Presence Penalty

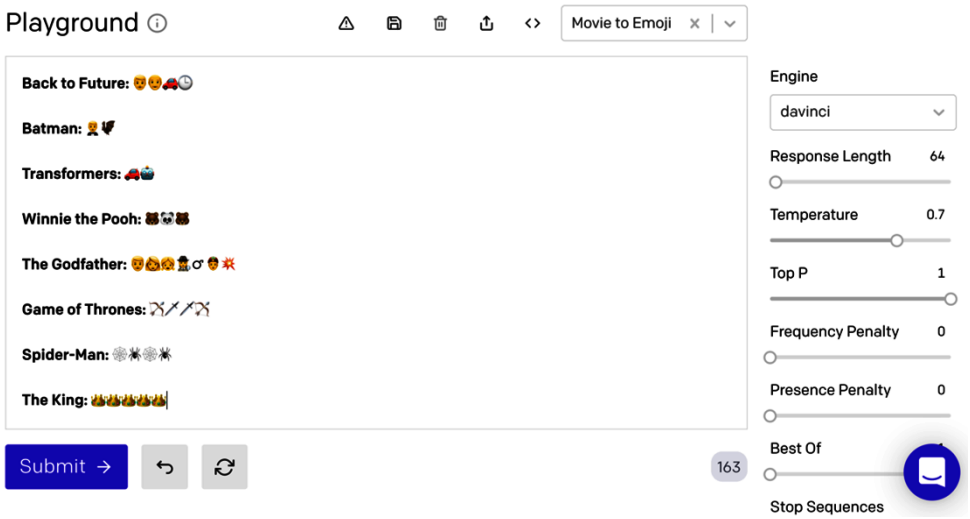

Playground
Load a preset...
words that rhyme have similar sounding endings
q: what rhymes with "cat"
a: bat, hat, mat
q: what rhymes with "small"
a: tall, wall, call
q: what rhymes with "pig"
a: big, dig, fig
q: what rhymes with "dog"
a: log, bog, frog
q: what rhymes with "duck"
Submit
137
Engine
davinci
Response Length 64
Temperature 0.3
Top P 1
Frequency Penalty 0
Presence Penalty 0
Best Of
Stop Sequences

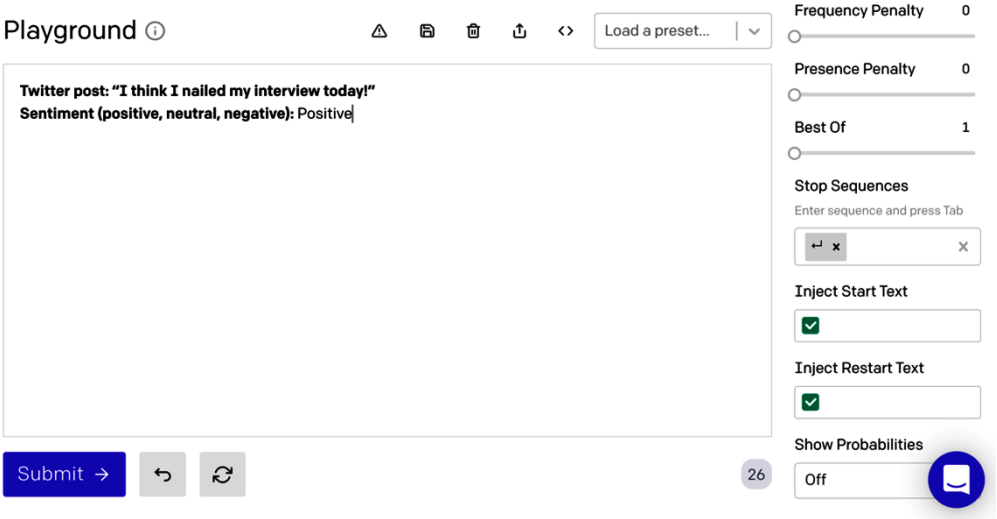

Playground
Load a preset...
Quantum mechanics is a fundamental theory in physics that provides a description of the physical properties of nature at the scale of atoms and subatomic particles.[2]:1.1 It is the foundation of all quantum physics including quantum chemistry, quantum field theory, quantum technology, and quantum information science.
Classical physics, the description of physics that existed before the theory of relativity and quantum mechanics, describes many aspects of nature at an ordinary (macroscopic) scale, while quantum mechanics explains the aspects of nature at small (atomic and subatomic) scales, for which classical mechanics is insufficient. Most theories in classical physics can be derived from quantum mechanics as an approximation valid at large (macroscopic) scale.[3]
tl;dr: Quantum mechanics is a theory of the subatomic world. It is the foundation of all of modern physics and is the most accurate theory of the physical world that we have.
Submit
185
Engine
davinci
Response Length 64
Temperature 0.3
Top P 1
Frequency Penalty 0
Presence Penalty 0
Best Of
Stop Sequences

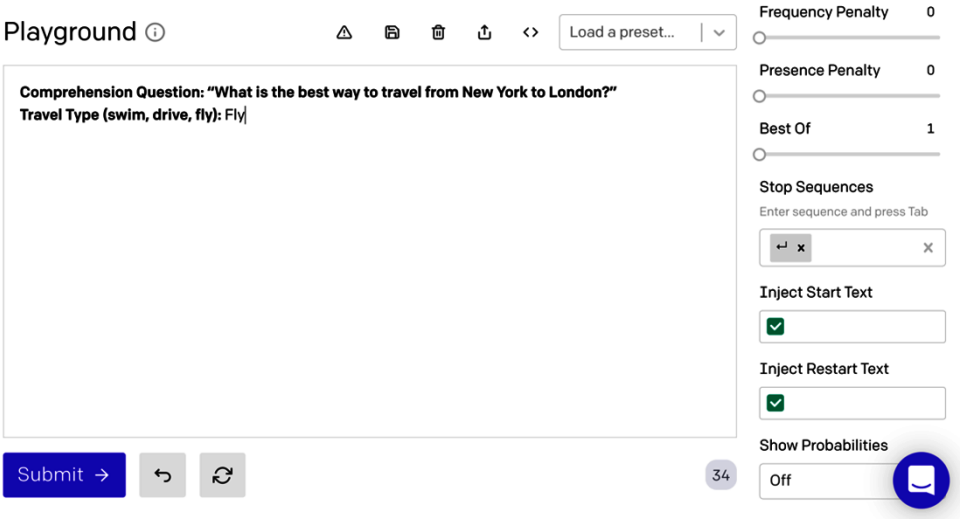

Playground
Load a preset...
(b) Ownership. As between you and OpenAI, we and our affiliates own all rights, title, and interest in and to the APIs, Content, and Developer Documentation and all associated elements, components, and executables. Subject to the foregoing, you own all rights, title, and interest in and to your Application. You have no right to distribute or allow access to the stand-alone APIs. Except as expressly provided in these Terms, neither party grants, nor shall the other party acquire, any right, title or interest (including any implied license) in or to any property of the first party or its affiliates under these Terms. All rights not expressly granted in these Terms are withheld.
one-sentence summary: OpenAI owns all rights in the APIs, Content, and Developer Documentation, and you own all rights in your Application
Submit
172
Frequency Penalty 0
Presence Penalty 0
Best Of 1
Stop Sequences
Enter sequence and press Tab
Inject Start Text
Inject Restart Text
Show Probabilities
Off

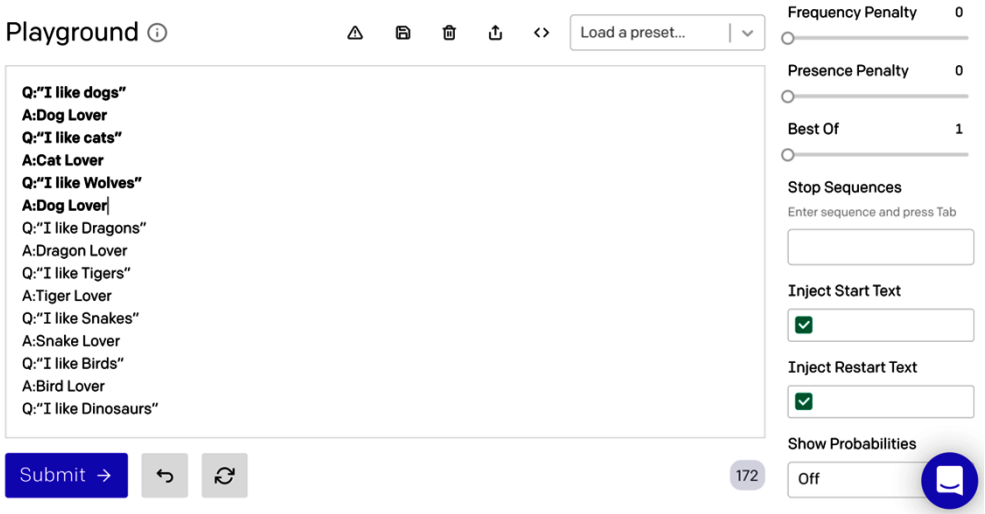

Playground
Load a preset...
The Milky Way[a] is the galaxy that contains our Solar System, with the name describing the galaxy's appearance from Earth: a hazy band of light seen in the night sky formed from stars that cannot be individually distinguished by the naked eye. The term Milky Way is a translation of the Latin via lactea, from the Greek γαλακτικός κύκλος (galaktikos kýklos, "milky circle").[19][20][21] From Earth, the Milky Way appears as a band because its disk-shaped structure is viewed from within. Galileo Galilei first resolved the band of light into individual stars with his telescope in 1610. Until the early 1920s, most astronomers thought that the Milky Way contained all the stars in the Universe.[22] Following the 1920 Great Debate between the astronomers Harlow Shapley and Heber Curtis,[23] observations by Edwin Hubble showed that the Milky Way is just one of many galaxies.
The Milky Way is a barred spiral galaxy with an estimated visible diameter of 150-200,000 light-years,[9][24][25] an increase from traditional estimates of 100,000 light-years. Recent simulations suggest that a dark matter disk, also containing some visible stars, may extend up to a diameter of almost 2 million light-years.[11][12]
I rephrased this in plain language that a third grader could understand.
The Milky Way is a band of light in the night sky formed from stars that cannot be individually distinguished by the naked eye.
Submit
331
davinci
Response Length 64
Temperature 0.3
Top P 1
Frequency Penalty 0
Presence Penalty 0
Best Of 1
Stop Sequences
Enter sequence and press Tab

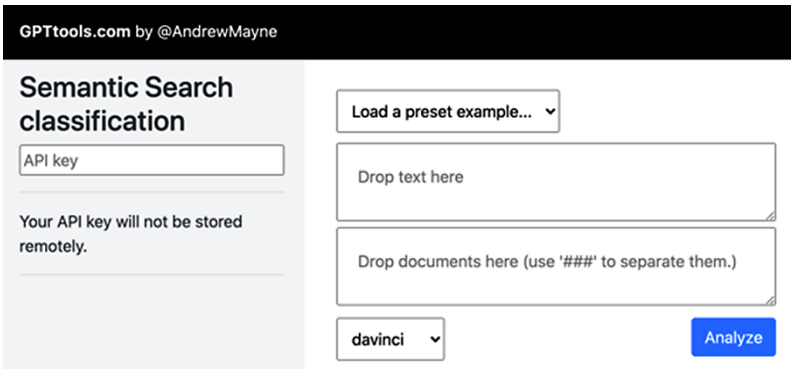

Playground
English to French
English: I do not speak French.
French: Je ne parle pas français.
English: See you later!
French: À tout à l'heure!
English: Where is a good restaurant?
French: Où est un bon restaurant?
English: What rooms do you have available?
French: Quelles chambres avez-vous de disponible?
English: What time do you close tonight?
French: A quelle heure fermez-vous ce soir?
Submit
125
Engine
davinci
Response Length 100
Temperature 0.5
Top P 1
Frequency Penalty 0
Presence Penalty 0
Best Of
Stop Sequences


Playground
Movie to Emoji
Back to Future:
Batman:
Transformers:
Winnie the Pooh:
The Godfather:
Game of Thrones:
Spider-Man:
The King:
Submit
163
Engine
davinci
Response Length 64
Temperature 0.7
Top P 1
Frequency Penalty 0
Presence Penalty 0
Best Of
Stop Sequences


Playground
Load a preset...
Twitter post: "I think I nailed my interview today!"
Sentiment (positive, neutral, negative): Positive
Submit
26
Frequency Penalty 0
Presence Penalty 0
Best Of 1
Stop Sequences
Enter sequence and press Tab
Inject Start Text
Inject Restart Text
Show Probabilities
Off


Playground
Load a preset...
Comprehension Question: "What is the best way to travel from New York to London?"
Travel Type (swim, drive, fly): Fly
Frequency Penalty 0
Presence Penalty 0
Best Of 1
Stop Sequences
Enter sequence and press Tab
Inject Start Text
Inject Restart Text
Show Probabilities
Off
Submit
34


Playground
Load a preset...
Q:"I like dogs"
A:Dog Lover
Q:"I like cats"
A:Cat Lover
Q:"I like Wolves"
A:Dog Lover
Q:"I like Dragons"
A:Dragon Lover
Q:"I like Tigers"
A:Tiger Lover
Q:"I like Snakes"
A:Snake Lover
Q:"I like Birds"
A:Bird Lover
Q:"I like Dinosaurs"
Frequency Penalty 0
Presence Penalty 0
Best Of 1
Stop Sequences
Enter sequence and press Tab
Inject Start Text
Inject Restart Text
Show Probabilities
Off
Submit
172


GPTtools.com by @AndrewMayne
Semantic Search classification
API key
Your API key will not be stored remotely.
Load a preset example...
Drop text here
Drop documents here (use '###' to separate them.)
davinci
Analyze

ORGANIZATION

dabblelab

Settings

Usage

Members

Billing

USER

API Keys

API Keys

This is your **Secret API Key**. Do not share this key with others, or expose it in the browser or other client-side code.

sk-vwccn [redacted]

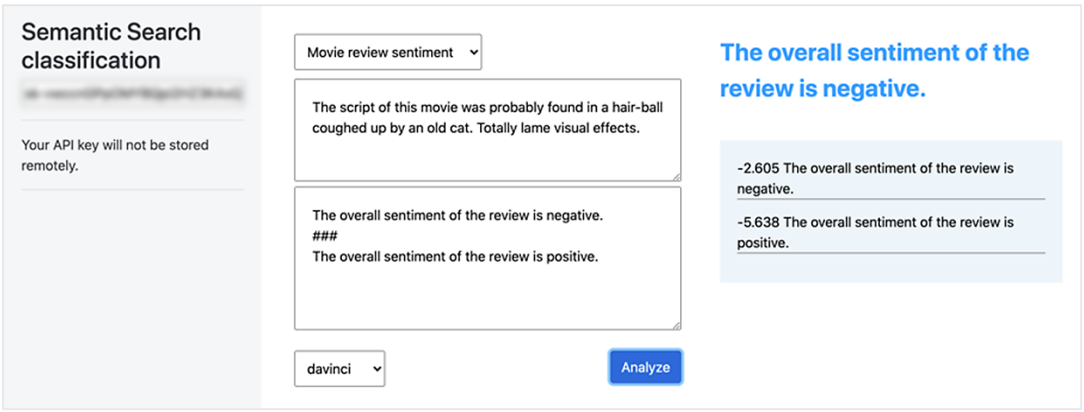
Rotate Key

Default Organization

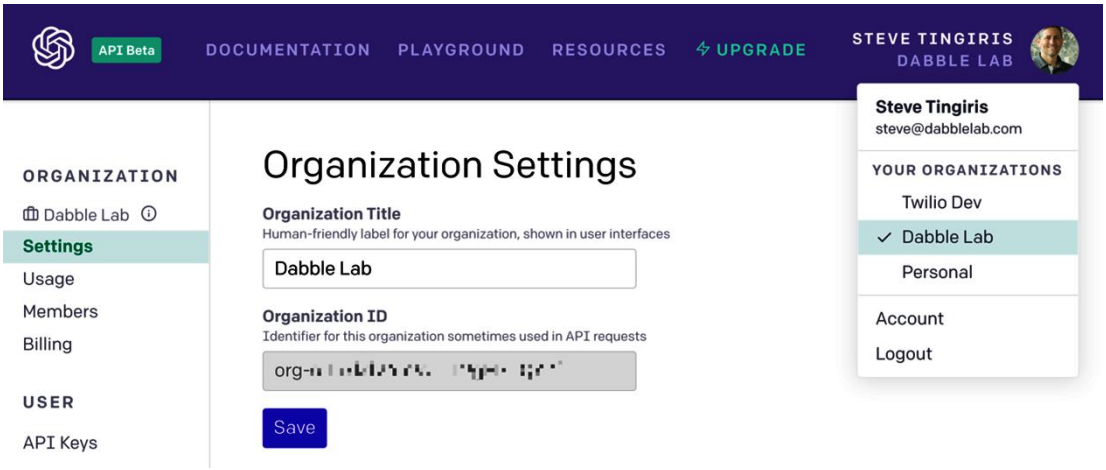
If you belong to multiple organizations, this setting controls which organization is used by default when making requests with the API key above.

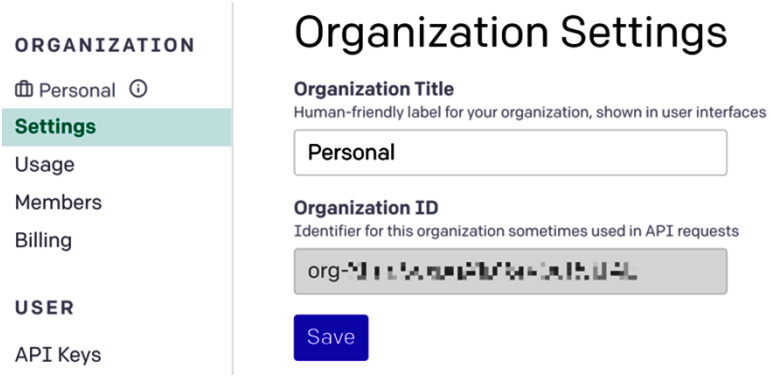
dabblelab

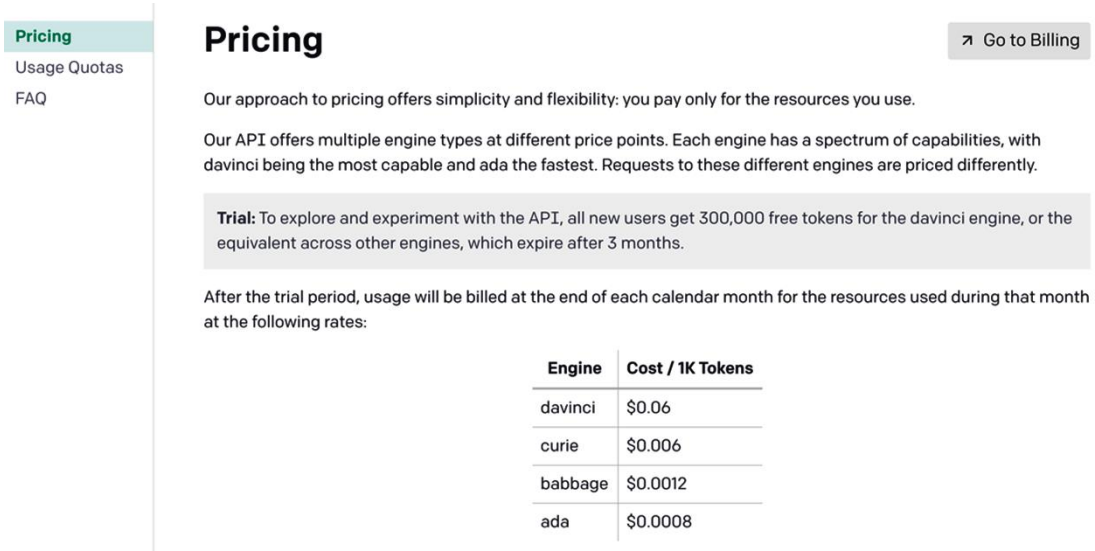
Note: You can also specify which organization to use for each API request. See Authentication to learn more.

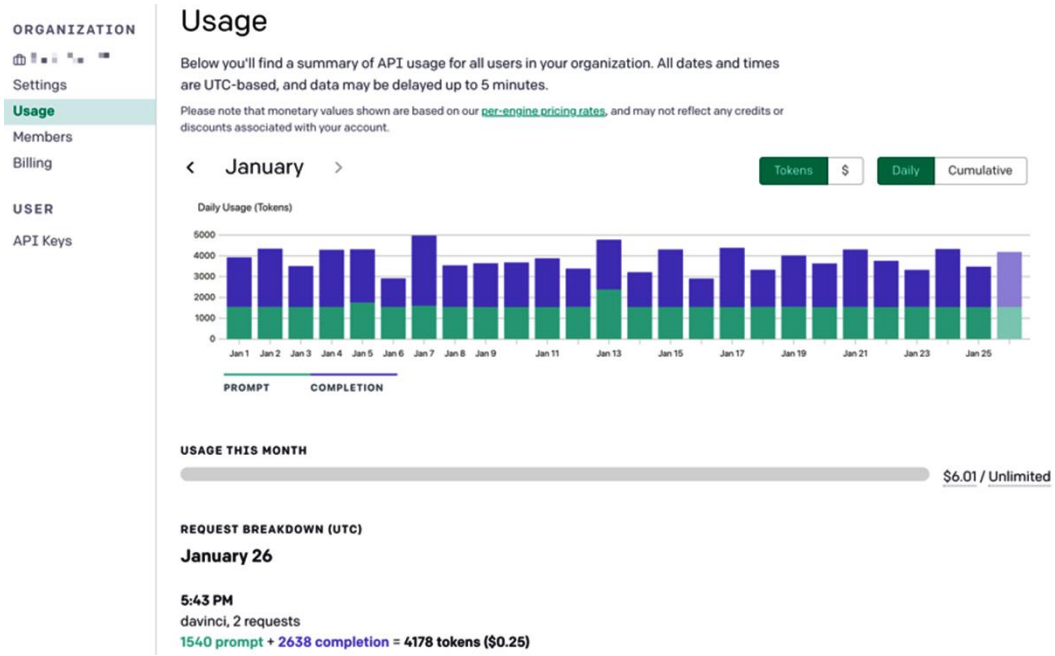


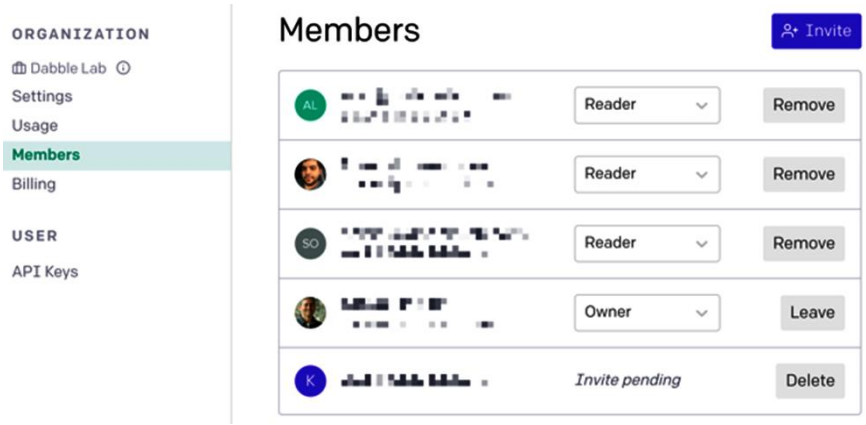
Chapter 3: Working with the OpenAI Playground

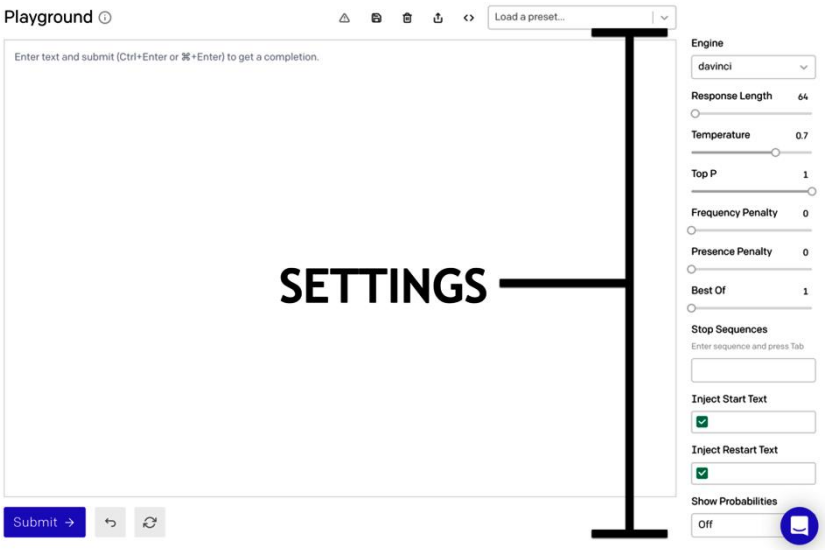


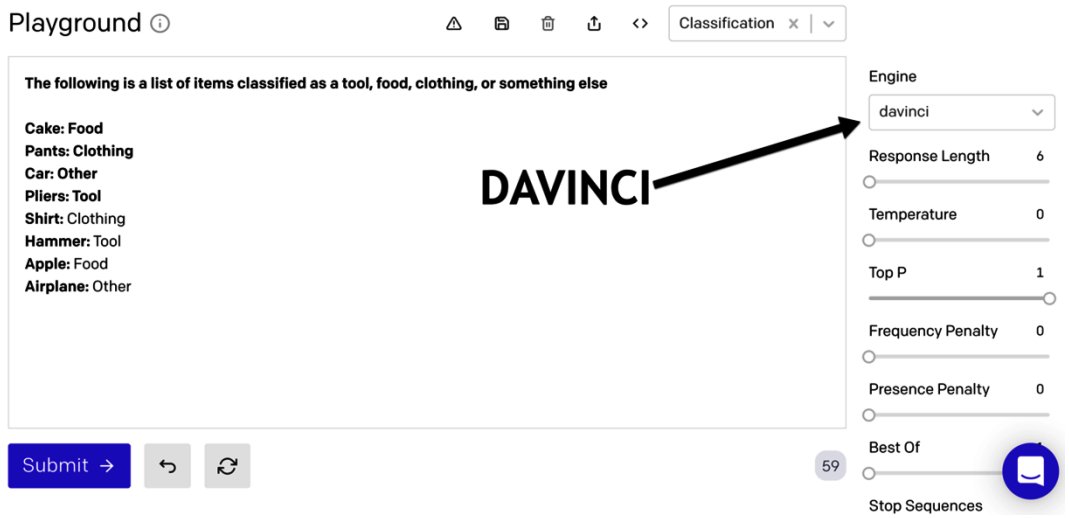


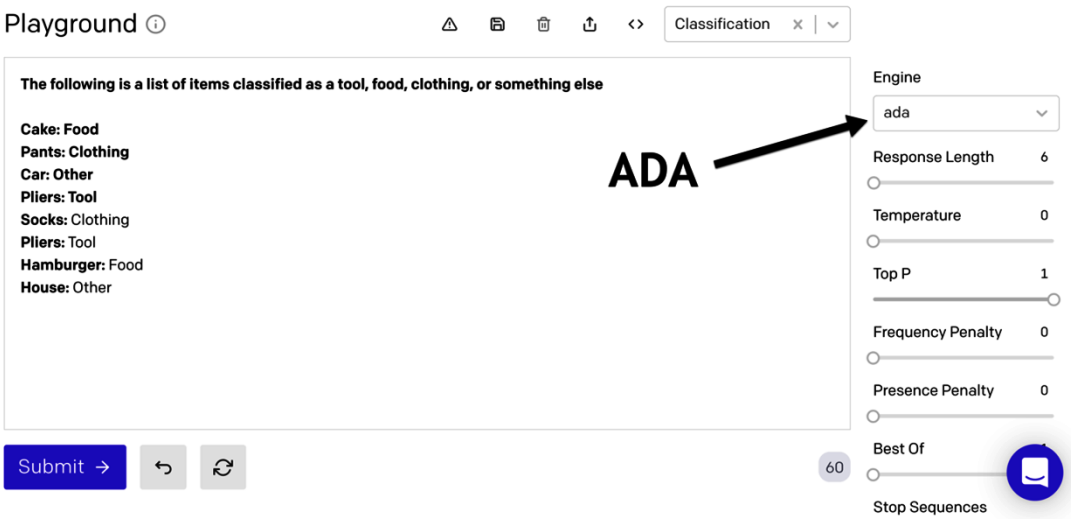


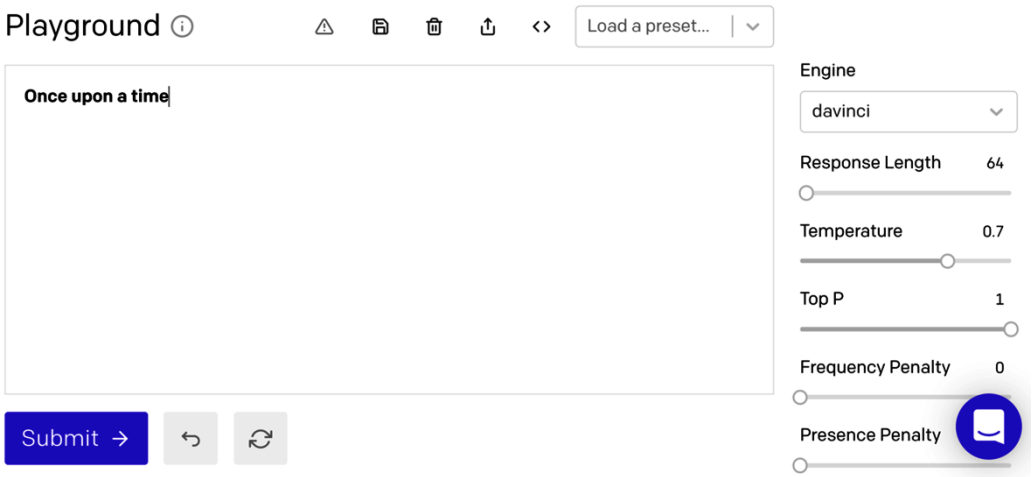

ORGANIZATION
Settings
Usage
Members
Billing
USER
API Keys
Usage
Below you'll find a summary of API usage for all users in your organization. All dates and times are UTC-based, and data may be delayed up to 5 minutes.
Please note that monetary values shown are based on our per-engine pricing rates, and may not reflect any credits or discounts associated with your account.
January
Tokens
$
Daily
Cumulative
Daily Usage (Tokens)
5000
4000
3000
2000
1000
0
Jan 1 Jan 2 Jan 3 Jan 4 Jan 5 Jan 6 Jan 7 Jan 8 Jan 9 Jan 11 Jan 13 Jan 15 Jan 17 Jan 19 Jan 21 Jan 23 Jan 25
PROMPT
COMPLETION
USAGE THIS MONTH
$6.01 / Unlimited
REQUEST BREAKDOWN (UTC)
January 26
5:43 PM
davinci, 2 requests
1540 prompt + 2638 completion = 4178 tokens ($0.25)

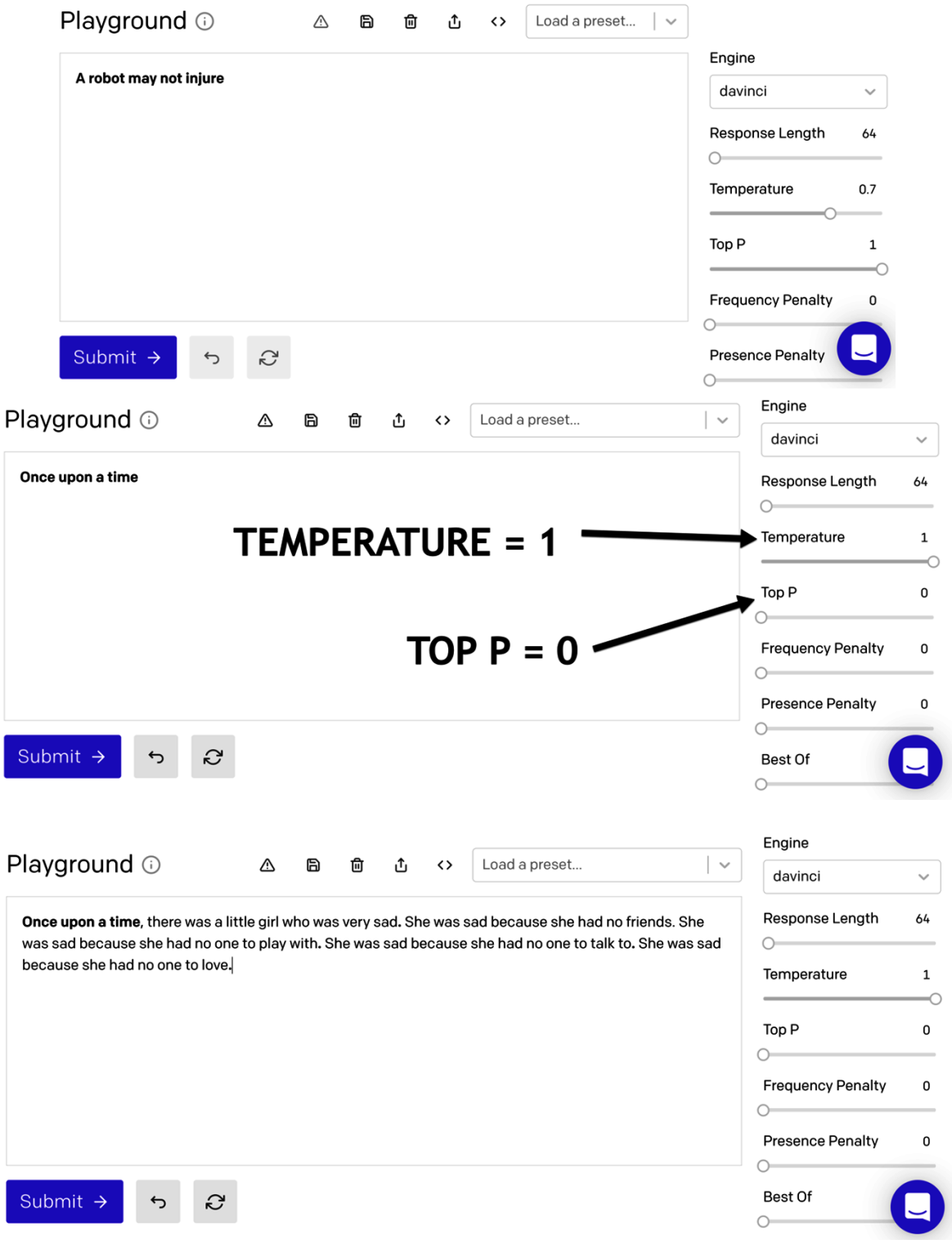

ORGANIZATION
Dabble Lab
Settings
Usage
Members
Billing
USER
API Keys
Members
Invite
Reader
Remove
Reader
Remove
Reader
Remove
Owner
Leave
Invite pending
Delete

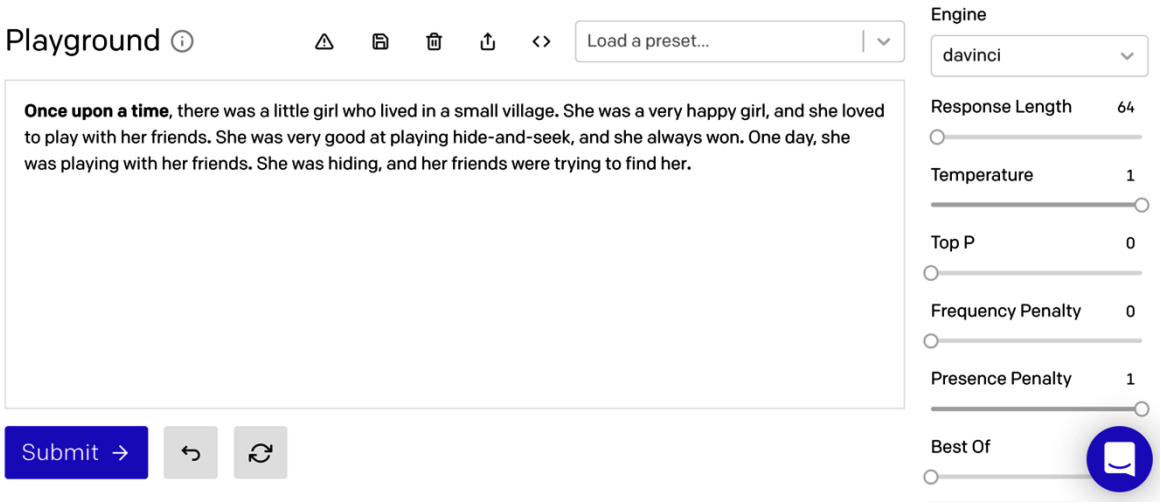

Playground
Load a preset...
Enter text and submit (Ctrl+Enter or ⌘+Enter) to get a completion.
SETTINGS
Engine
davinci
Response Length 64
Temperature 0.7
Top P 1
Frequency Penalty 0
Presence Penalty 0
Best Of 1
Stop Sequences
Enter sequence and press Tab
Inject Start Text
Inject Restart Text
Show Probabilities
Off
Submit →

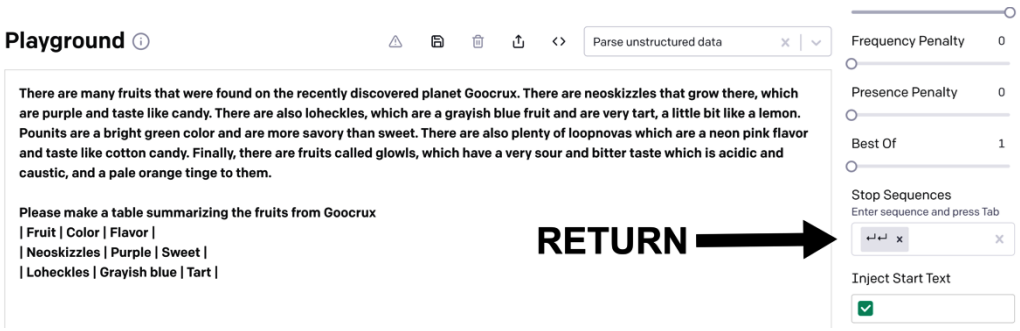

Playground
Classification
The following is a list of items classified as a tool, food, clothing, or something else
Cake: Food
Pants: Clothing
Car: Other
Pliers: Tool
Shirt: Clothing
Hammer: Tool
Apple: Food
Airplane: Other
DAVINCI
Engine
davinci
Response Length 6
Temperature 0
Top P 1
Frequency Penalty 0
Presence Penalty 0
Best Of
Stop Sequences
Submit
59

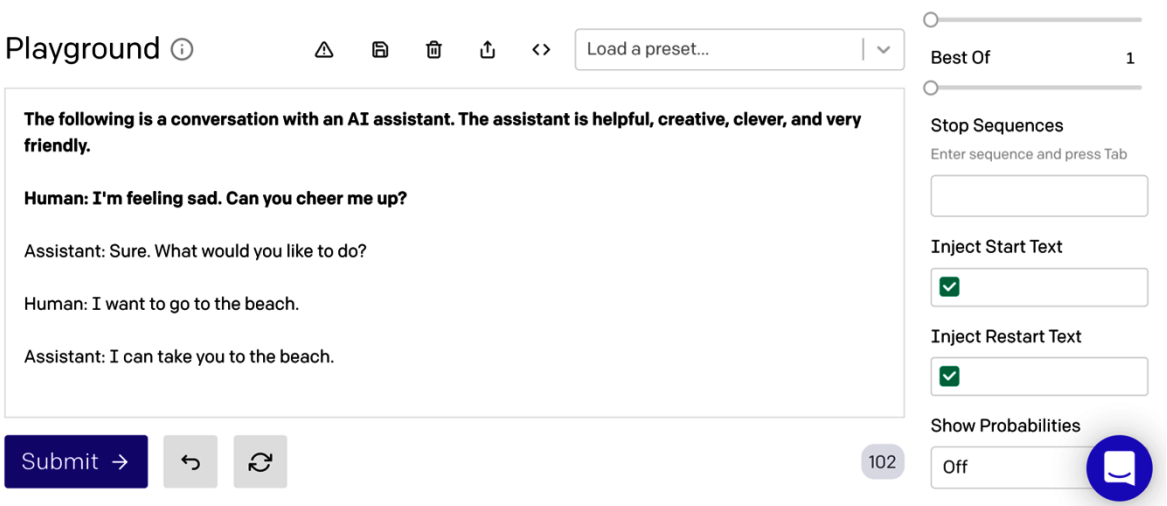

Playground
Classification
The following is a list of items classified as a tool, food, clothing, or something else
Cake: Food
Pants: Clothing
Car: Other
Pliers: Tool
Socks: Clothing
Pliers: Tool
Hamburger: Food
House: Other
ADA
Engine
ada
Response Length 6
Temperature 0
Top P 1
Frequency Penalty 0
Presence Penalty 0
Best Of
Stop Sequences
Submit
60

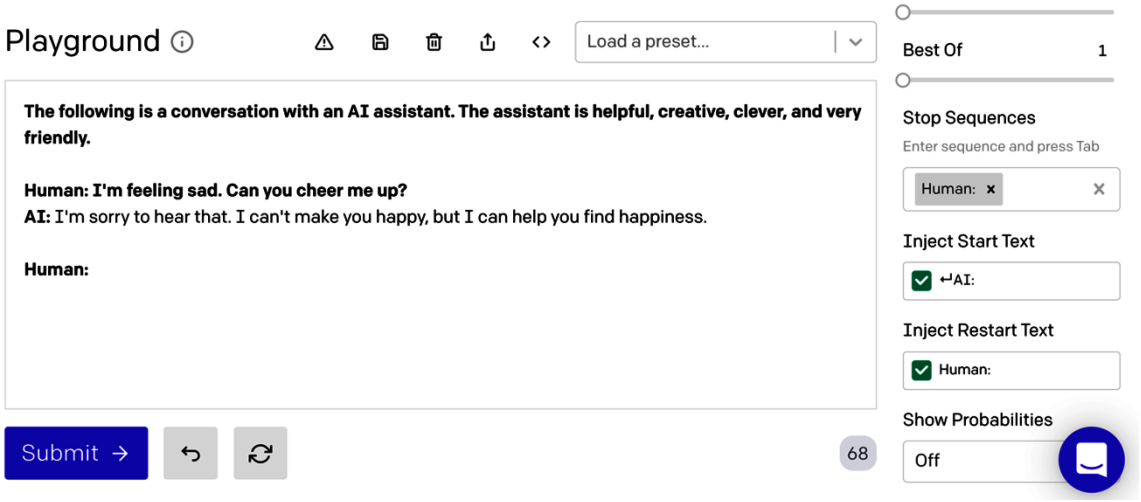

Playground
Load a preset...
Once upon a time
Engine
davinci
Response Length 64
Temperature 0.7
Top P 1
Frequency Penalty 0
Presence Penalty
Submit

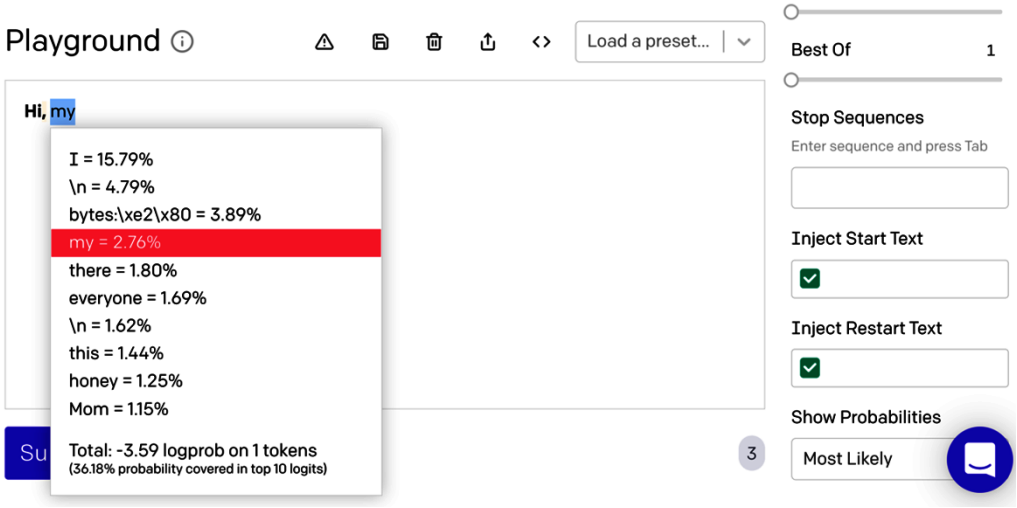

Playground
Load a preset...
A robot may not injure
Submit
Engine
davinci
Response Length 64
Temperature 0.7
Top P 1
Frequency Penalty 0
Presence Penalty 0
Playground
Load a preset...
Once upon a time
TEMPERATURE = 1
TOP P = 0
Submit
Engine
davinci
Response Length 64
Temperature 1
Top P 0
Frequency Penalty 0
Presence Penalty 0
Best Of
Playground
Load a preset...
Once upon a time, there was a little girl who was very sad. She was sad because she had no friends. She was sad because she had no one to play with. She was sad because she had no one to talk to. She was sad because she had no one to love.
Submit
Engine
davinci
Response Length 64
Temperature 1
Top P 0
Frequency Penalty 0
Presence Penalty 0
Best Of

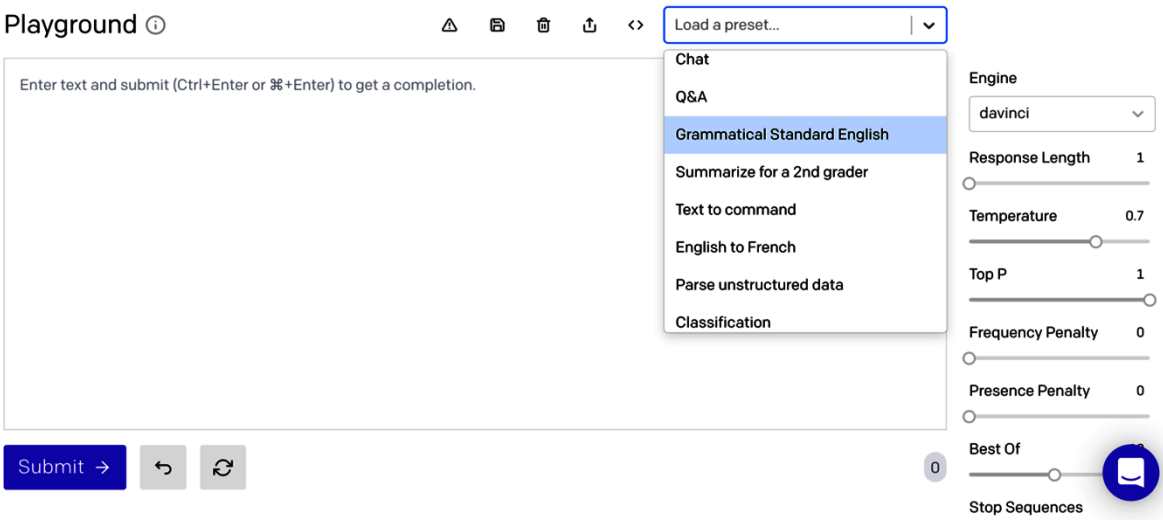

Playground
Load a preset...
Once upon a time, there was a little girl who lived in a small village. She was a very happy girl, and she loved to play with her friends. She was very good at playing hide-and-seek, and she always won. One day, she was playing with her friends. She was hiding, and her friends were trying to find her.
Submit
Engine
davinci
Response Length 64
Temperature 1
Top P 0
Frequency Penalty 0
Presence Penalty 1
Best Of


Playground
Parse unstructured data
There are many fruits that were found on the recently discovered planet Goocrux. There are neoskizzles that grow there, which are purple and taste like candy. There are also loheckles, which are a grayish blue fruit and are very tart, a little bit like a lemon. Pounits are a bright green color and are more savory than sweet. There are also plenty of loopnovas which are a neon pink flavor and taste like cotton candy. Finally, there are fruits called glowls, which have a very sour and bitter taste which is acidic and caustic, and a pale orange tinge to them.
Please make a table summarizing the fruits from Goocrux
Fruit	Color	Flavor
Neoskizzles	Purple	Sweet
Loheckles	Grayish blue	Tart
RETURN
Frequency Penalty 0
Presence Penalty 0
Best Of 1
Stop Sequences
Enter sequence and press Tab
Inject Start Text


Playground
Load a preset...
The following is a conversation with an AI assistant. The assistant is helpful, creative, clever, and very friendly.
Human: I'm feeling sad. Can you cheer me up?
Assistant: Sure. What would you like to do?
Human: I want to go to the beach.
Assistant: I can take you to the beach.
Submit
102
Best Of 1
Stop Sequences
Enter sequence and press Tab
Inject Start Text
Inject Restart Text
Show Probabilities
Off

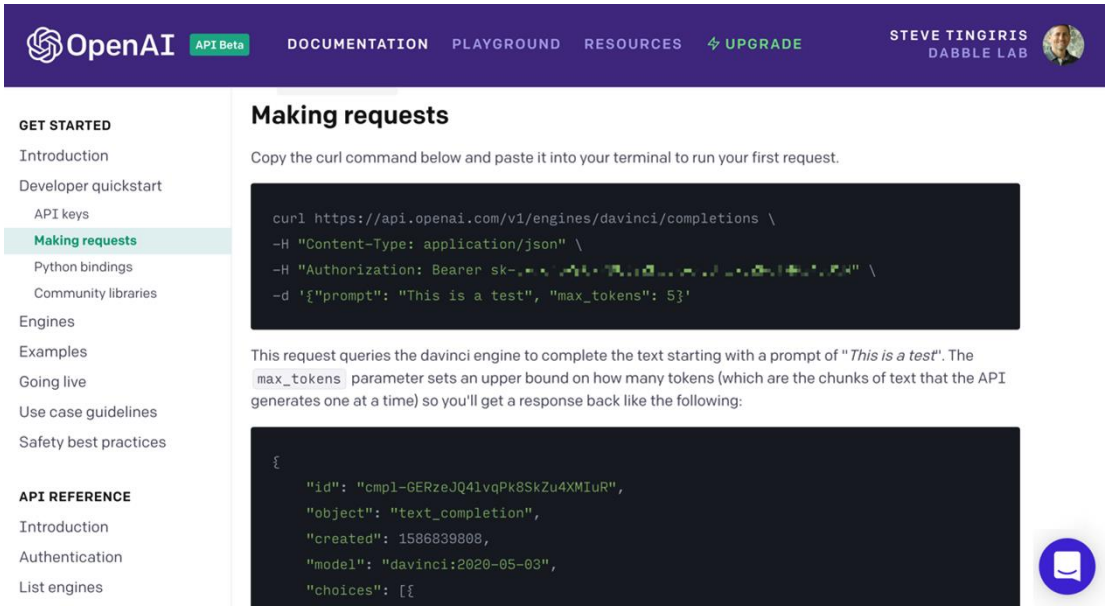

Playground
Load a preset...
The following is a conversation with an AI assistant. The assistant is helpful, creative, clever, and very friendly.
Human: I'm feeling sad. Can you cheer me up?
AI: I'm sorry to hear that. I can't make you happy, but I can help you find happiness.
Human:
Submit
68
Best Of 1
Stop Sequences
Enter sequence and press Tab
Human:
Inject Start Text
↵AI:
Inject Restart Text
Human:
Show Probabilities
Off


Playground
Load a preset...
Hi, my
I = 15.79%
\n = 4.79%
bytes:\xe2\x80 = 3.89%
my = 2.76%
there = 1.80%
everyone = 1.69%
\n = 1.62%
this = 1.44%
honey = 1.25%
Mom = 1.15%
Total: -3.59 logprob on 1 tokens
(36.18% probability covered in top 10 logits)
3
Best Of 1
Stop Sequences
Enter sequence and press Tab
Inject Start Text
Inject Restart Text
Show Probabilities
Most Likely

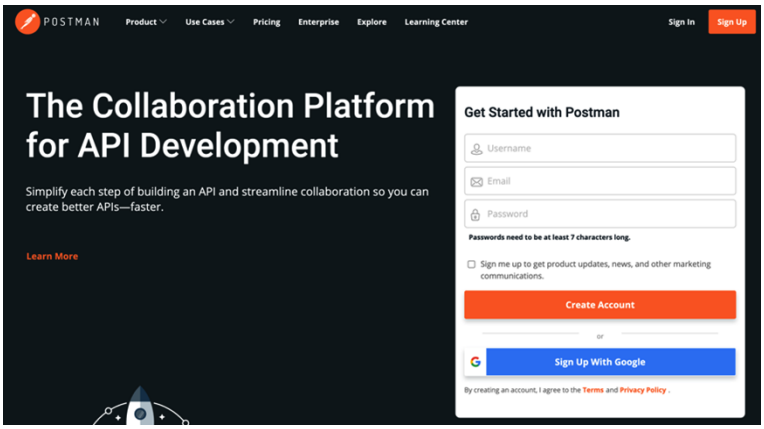

Playground
Load a preset...
Chat
Q&A
Grammatical Standard English
Summarize for a 2nd grader
Text to command
English to French
Parse unstructured data
Classification
Enter text and submit (Ctrl+Enter or ⌘+Enter) to get a completion.
Submit
0
Engine
davinci
Response Length 1
Temperature 0.7
Top P 1
Frequency Penalty 0
Presence Penalty 0
Best Of
Stop Sequences

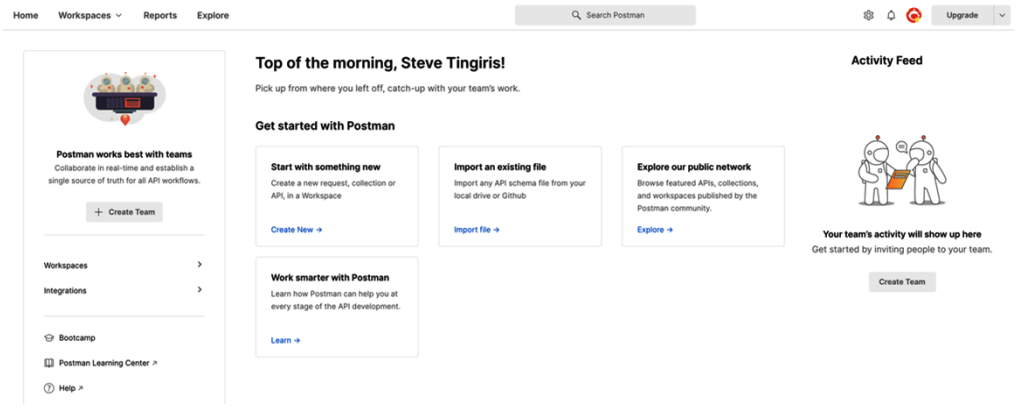
Chapter 4: Working with the OpenAI API

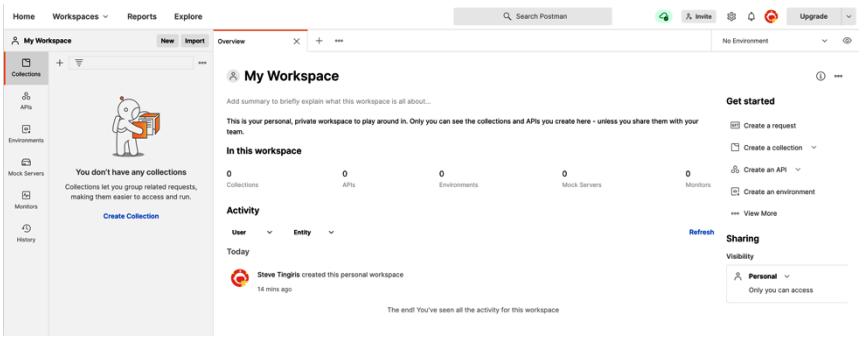

```
{"message": "success", "number": 7, "people":
[{"craft": "ISS", "name": "Sergey Ryzhikov"},
{"craft": "ISS", "name": "Kate Rubins"},
{"craft": "ISS", "name": "Sergey Kud-
Sverchkov"}, {"craft": "ISS", "name": "Mike
Hopkins"}, {"craft": "ISS", "name": "Victor
Glover"}, {"craft": "ISS", "name": "Shannon
Walker"}, {"craft": "ISS", "name": "Soichi
Noguchi"}]}
```

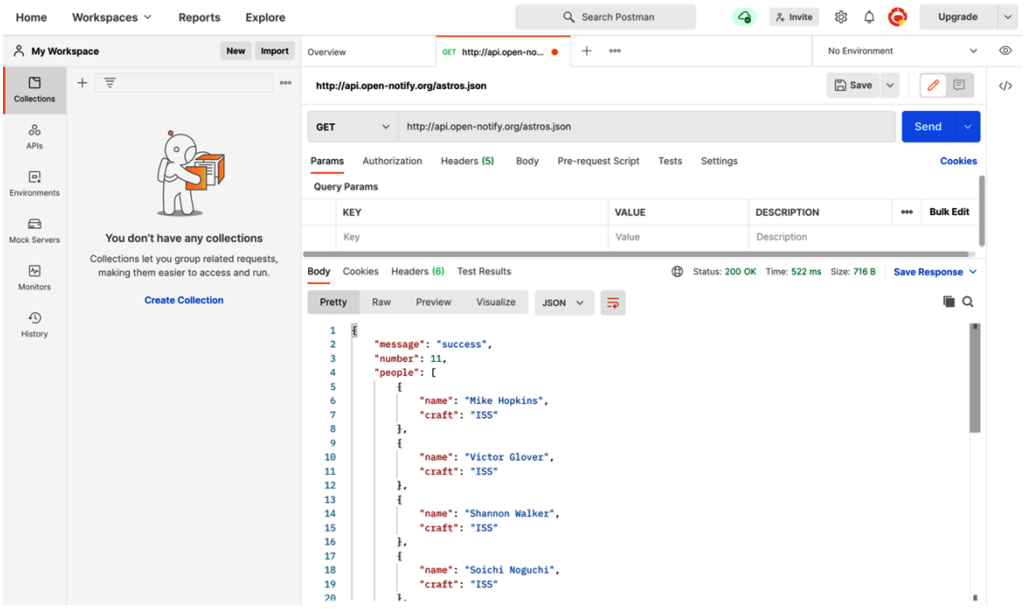



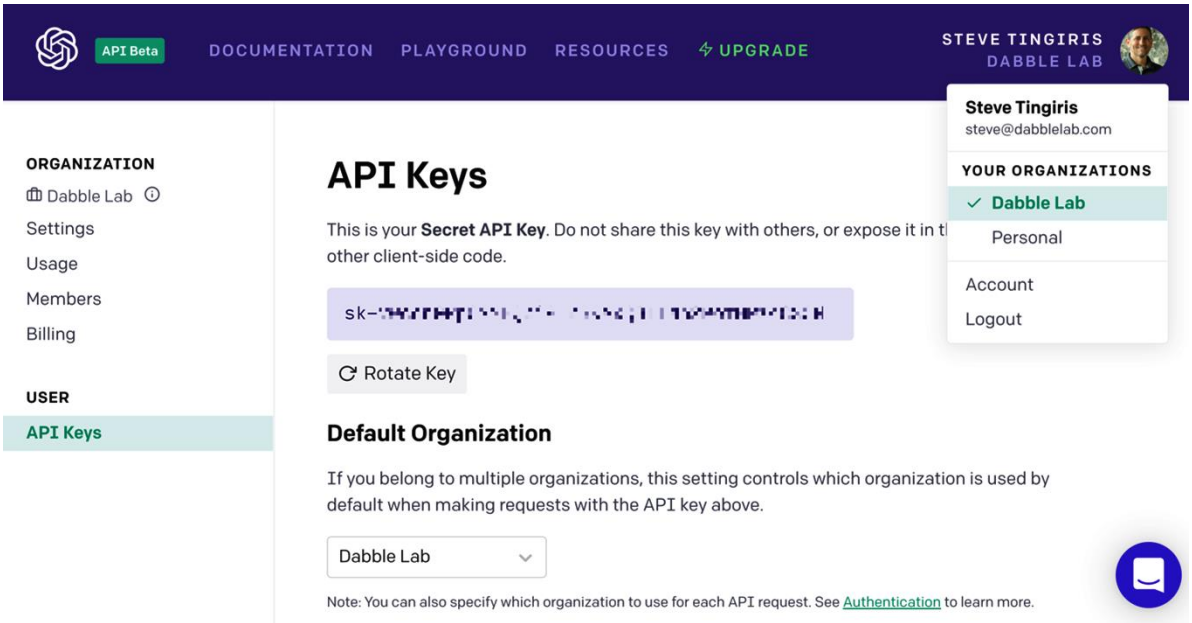
```
Last login: Fri Jan  1 10:57:12 on console
$ curl
curl: try 'curl --help' or 'curl --manual' for more information
$ curl --help
Usage: curl [options...] <url>
     --abstract-unix-socket <path> Connect via abstract Unix domain socket
     --alt-svc <file name> Enable alt-svc with this cache file
     --anyauth       Pick any authentication method
 -a, --append        Append to target file when uploading
     --basic         Use HTTP Basic Authentication
     --cacert <file> CA certificate to verify peer against
     --capath <dir>  CA directory to verify peer against
 -E, --cert <certificate[:password]> Client certificate file and password
     --cert-status   Verify the status of the server certificate
     --cert-type <type> Certificate file type (DER/PEM/ENG)
     --ciphers <list of ciphers> SSL ciphers to use
     --compressed    Request compressed response
     --compressed-ssh Enable SSH compression
 -K, --config <file> Read config from a file
     --connect-timeout <seconds> Maximum time allowed for connection
     --connect-to <HOST1:PORT1:HOST2:PORT2> Connect to host
 -C, --continue-at <offset> Resumed transfer offset
 -b, --cookie <data|filename> Send cookies from string/file
 -c, --cookie-jar <filename> Write cookies to <filename> after operation
     --create-dirs   Create necessary local directory hierarchy
     --crlf          Convert LF to CRLF in upload
     --crlfile <file> Get a CRL list in PEM format from the given file
 -d, --data <data>   HTTP POST data
     --data-ascii <data> HTTP POST ASCII data
     --data-binary <data> HTTP POST binary data
     --data-raw <data> HTTP POST data, '@' allowed
     --data-urlencode <data> HTTP POST data url encoded
     --delegation <LEVEL> GSS-API delegation permission
     --digest        Use HTTP Digest Authentication
 -q, --disable       Disable .curlrc
     --disable-eprt  Inhibit using EPRT or LPRT
     --disable-epsv  Inhibit using EPSV
     --disallow-username-in-url Disallow username in url
     --dns-interface <interface> Interface to use for DNS requests
     --dns-ipv4-addr <address> IPv4 address to use for DNS requests
     --dns-ipv6-addr <address> IPv6 address to use for DNS requests
     --dns-servers <addresses> DNS server addrs to use
     --doh-url <URL> Resolve host names over DOH
 -D, --dump-header <filename> Write the received headers to <filename>
     --egd-file <file> EGD socket path for random data
     --engine <name> Crypto engine to use
     --expect100-timeout <seconds> How long to wait for 100-continue
 -f, --fail          Fail silently (no output at all) on HTTP errors
     --fail-early    Fail on first transfer error, do not continue
     --false-start   Enable TLS False Start
 -F, --form <name=content> Specify multipart MIME data
```

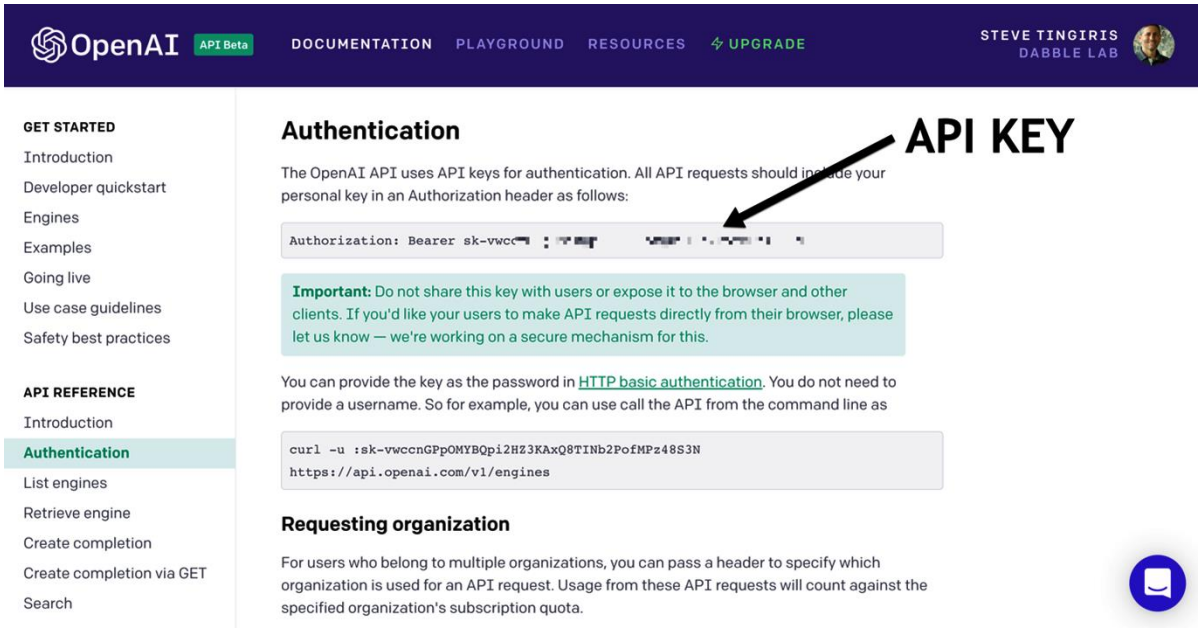


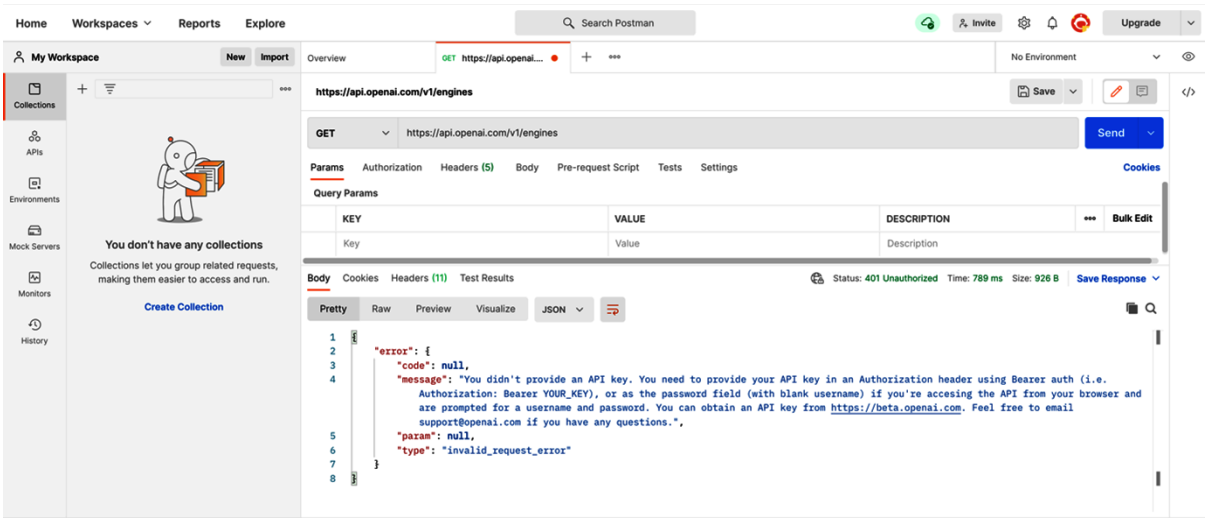


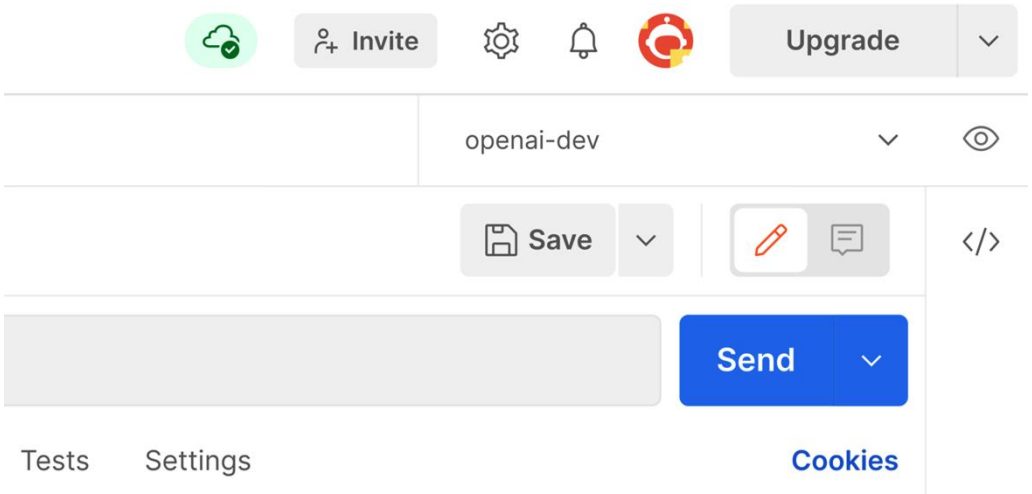

Home
Workspaces
Reports
Explore
Search Postman
Invite
Upgrade
My Workspace
New
Import
Overview
No Environment
Collections
APIs
Environments
Mock Servers
Monitors
History
You don't have any collections
Collections let you group related requests, making them easier to access and run.
Create Collection
My Workspace
Add summary to briefly explain what this workspace is all about...
This is your personal, private workspace to play around in. Only you can see the collections and APIs you create here - unless you share them with your team.
In this workspace
0 Collections
0 APIs
0 Environments
0 Mock Servers
0 Monitors
Activity
User
Entity
Refresh
Today
Steve Tingiris created this personal workspace
14 mins ago
The end! You've seen all the activity for this workspace
Get started
Create a request
Create a collection
Create an API
Create an environment
View More
Sharing
Visibility
Personal
Only you can access

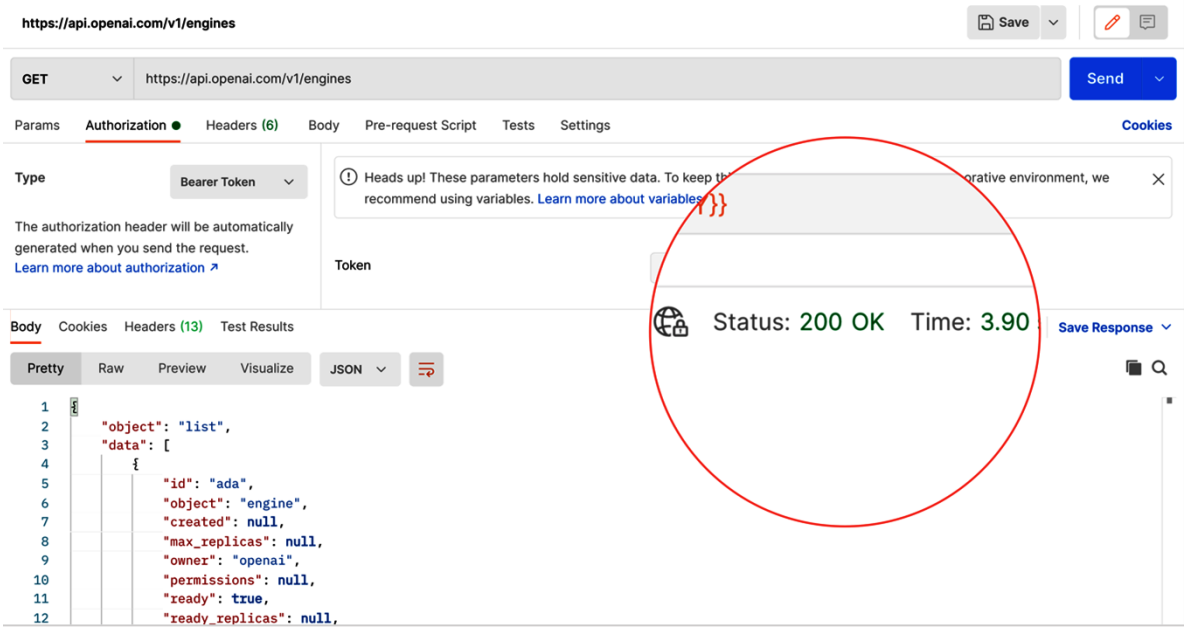

Home
Workspaces
Reports
Explore
Search Postman
Invite
Upgrade
My Workspace
New
Import
Overview
GET http://api.open-no...
No Environment
Collections
APIs
Environments
Mock Servers
Monitors
History
You don't have any collections
Collections let you group related requests, making them easier to access and run.
Create Collection
http://api.open-notify.org/astros.json
Save
GET
http://api.open-notify.org/astros.json
Send
Params
Authorization
Headers (5)
Body
Pre-request Script
Tests
Settings
Cookies
Query Params
KEY
VALUE
DESCRIPTION
Bulk Edit
Key
Value
Description
Body
Cookies
Headers (6)
Test Results
Status: 200 OK
Time: 522 ms
Size: 716 B
Save Response
Pretty
Raw
Preview
Visualize
JSON
{
"message": "success",
"number": 11,
"people": [
{
"name": "Mike Hopkins",
"craft": "ISS"
},
{
"name": "Victor Glover",
"craft": "ISS"
},
{
"name": "Shannon Walker",
"craft": "ISS"
},
{
"name": "Soichi Noguchi",
"craft": "ISS"

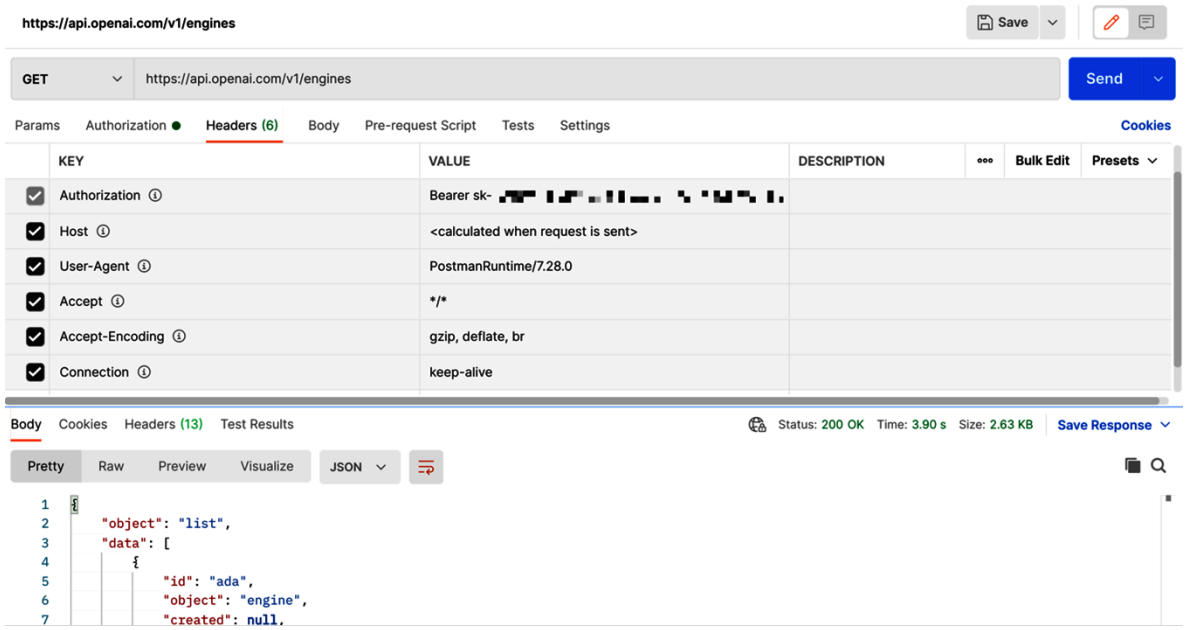

API Beta
DOCUMENTATION
PLAYGROUND
RESOURCES
UPGRADE
STEVE TINGIRIS
DABBLE LAB
Steve Tingiris
steve@dabblelab.com
YOUR ORGANIZATIONS
Dabble Lab
Personal
Account
Logout
ORGANIZATION
Dabble Lab
Settings
Usage
Members
Billing
USER
API Keys
API Keys
This is your Secret API Key. Do not share this key with others, or expose it in t
other client-side code.
sk-
Rotate Key
Default Organization
If you belong to multiple organizations, this setting controls which organization is used by default when making requests with the API key above.
Dabble Lab
Note: You can also specify which organization to use for each API request. See Authentication to learn more.

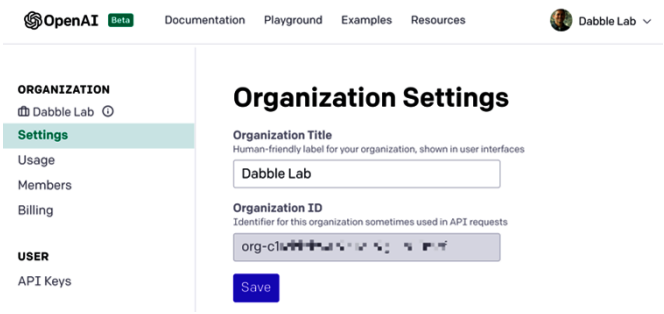

OpenAI
API Beta
DOCUMENTATION
PLAYGROUND
RESOURCES
UPGRADE
STEVE TINGIRIS
DABBLE LAB
GET STARTED
Introduction
Developer quickstart
Engines
Examples
Going live
Use case guidelines
Safety best practices
API REFERENCE
Introduction
Authentication
List engines
Retrieve engine
Create completion
Create completion via GET
Search
Authentication
API KEY
The OpenAI API uses API keys for authentication. All API requests should include your personal key in an Authorization header as follows:
Authorization: Bearer sk-vwcc
Important: Do not share this key with users or expose it to the browser and other clients. If you'd like your users to make API requests directly from their browser, please let us know — we're working on a secure mechanism for this.
You can provide the key as the password in HTTP basic authentication. You do not need to provide a username. So for example, you can use call the API from the command line as
curl -u :sk-vwccnGPpOMYBQpi2HZ3KAxQ8TINb2PofMPz48S3N
https://api.openai.com/v1/engines
Requesting organization
For users who belong to multiple organizations, you can pass a header to specify which organization is used for an API request. Usage from these API requests will count against the specified organization's subscription quota.

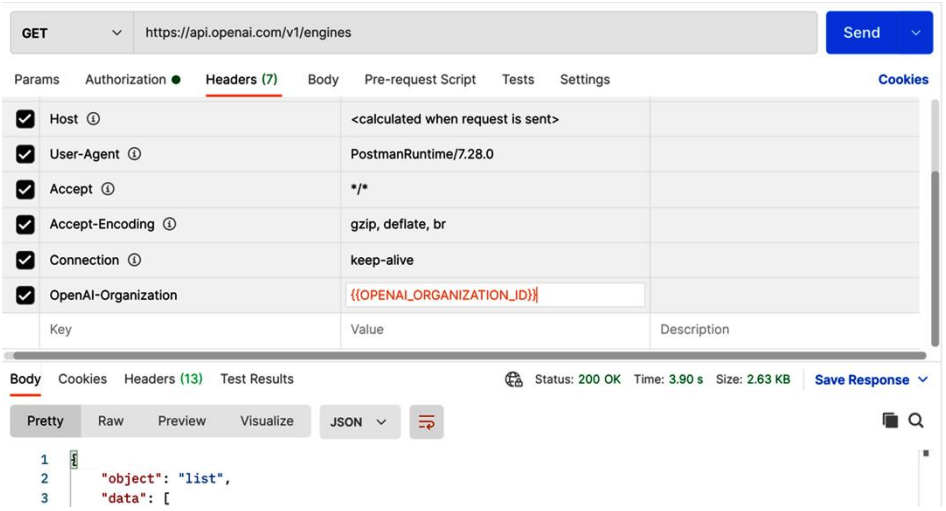

Home
Workspaces
Reports
Explore
Search Postman
Invite
Upgrade
My Workspace
New
Import
Overview
GET https://api.openai....
No Environment
Collections
APIs
Environments
Mock Servers
Monitors
History
You don't have any collections
Collections let you group related requests, making them easier to access and run.
Create Collection
https://api.openai.com/v1/engines
Save
GET
https://api.openai.com/v1/engines
Send
Params
Authorization
Headers (5)
Body
Pre-request Script
Tests
Settings
Cookies
Query Params
KEY
VALUE
DESCRIPTION
Bulk Edit
Key
Value
Description
Body
Cookies
Headers (11)
Test Results
Status: 401 Unauthorized
Time: 789 ms
Size: 926 B
Save Response
Pretty
Raw
Preview
Visualize
JSON
{
"error": {
"code": null,
"message": "You didn't provide an API key. You need to provide your API key in an Authorization header using Bearer auth (i.e. Authorization: Bearer YOUR_KEY), or as the password field (with blank username) if you're accesing the API from your browser and are prompted for a username and password. You can obtain an API key from https://beta.openai.com. Feel free to email support@openai.com if you have any questions.",
"param": null,
"type": "invalid_request_error"
}
}


Invite
Upgrade
openai-dev
Save
Send
Tests
Settings
Cookies

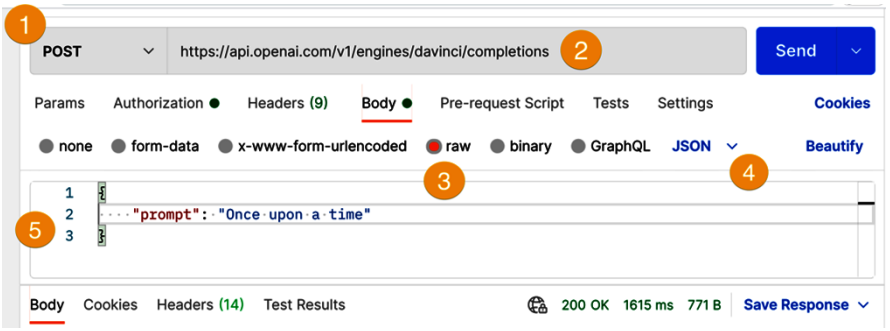

https://api.openai.com/v1/engines
Save
GET
https://api.openai.com/v1/engines
Send
Params
Authorization
Headers (6)
Body
Pre-request Script
Tests
Settings
Cookies
Type
Bearer Token
The authorization header will be automatically generated when you send the request.
Learn more about authorization
Heads up! These parameters hold sensitive data. To keep th
orative environment, we recommend using variables. Learn more about variables
Token
Body
Cookies
Headers (13)
Test Results
Status: 200 OK
Time: 3.90
Save Response
Pretty
Raw
Preview
Visualize
JSON
{
"object": "list",
"data": [
{
"id": "ada",
"object": "engine",
"created": null,
"max_replicas": null,
"owner": "openai",
"permissions": null,
"ready": true,
"ready_replicas": null,

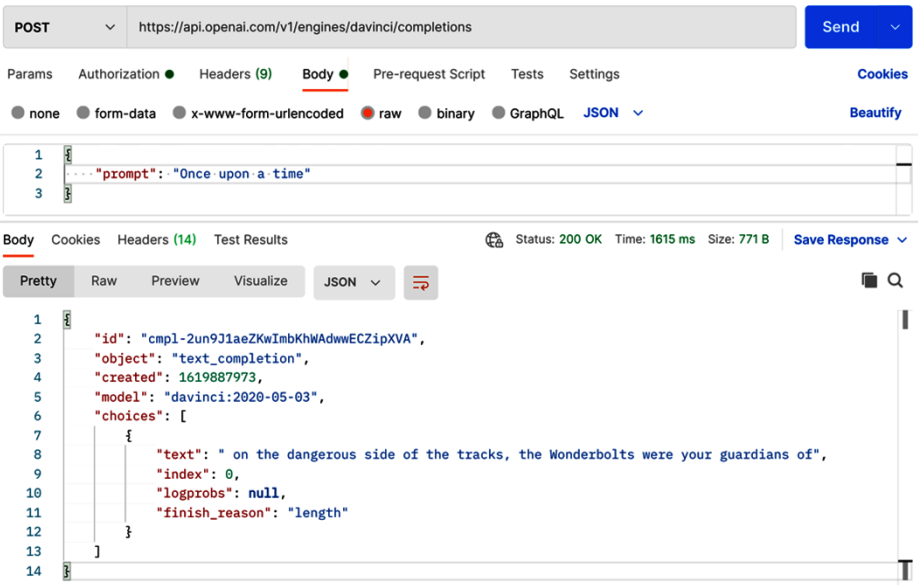

https://api.openai.com/v1/engines
Save
GET
https://api.openai.com/v1/engines
Send
Params
Authorization
Headers (6)
Body
Pre-request Script
Tests
Settings
Cookies
KEY
VALUE
DESCRIPTION
Bulk Edit
Presets
Authorization
Bearer sk-
Host
<calculated when request is sent>
User-Agent
PostmanRuntime/7.28.0
Accept
/
Accept-Encoding
gzip, deflate, br
Connection
keep-alive
Body
Cookies
Headers (13)
Test Results
Status: 200 OK
Time: 3.90 s
Size: 2.63 KB
Save Response
Pretty
Raw
Preview
Visualize
JSON
{
"object": "list",
"data": [
{
"id": "ada",
"object": "engine",
"created": null,

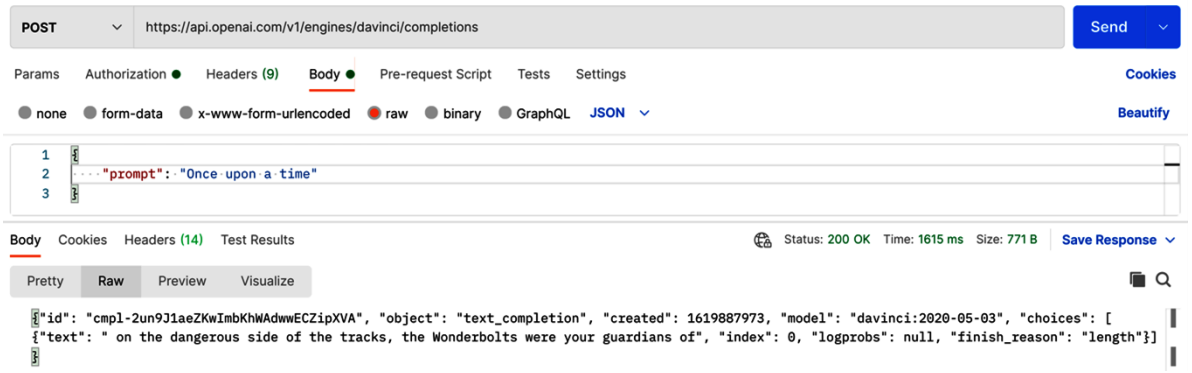

OpenAI
Beta
Documentation
Playground
Examples
Resources
Dabble Lab
ORGANIZATION
Dabble Lab
Settings
Usage
Members
Billing
USER
API Keys
Organization Settings
Organization Title
Human-friendly label for your organization, shown in user interfaces
Dabble Lab
Organization ID
Identifier for this organization sometimes used in API requests
org-c1
Save

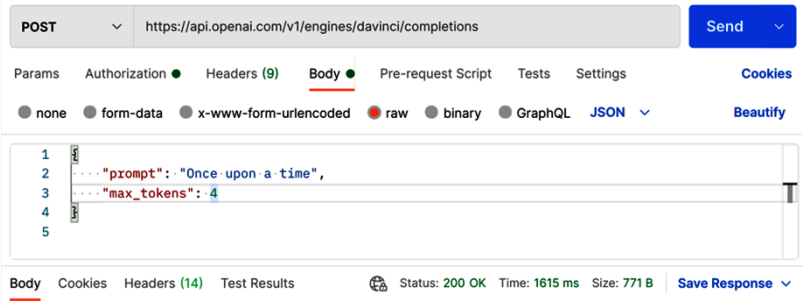

GET
https://api.openai.com/v1/engines
Send
Params
Authorization
Headers (7)
Body
Pre-request Script
Tests
Settings
Cookies
Host
<calculated when request is sent>
User-Agent
PostmanRuntime/7.28.0
Accept
/
Accept-Encoding
gzip, deflate, br
Connection
keep-alive
OpenAI-Organization
{{OPENAI_ORGANIZATION_ID}}
Key
Value
Description
Body
Cookies
Headers (13)
Test Results
Status: 200 OK
Time: 3.90 s
Size: 2.63 KB
Save Response
Pretty
Raw
Preview
Visualize
JSON
{
"object": "list",
"data": [

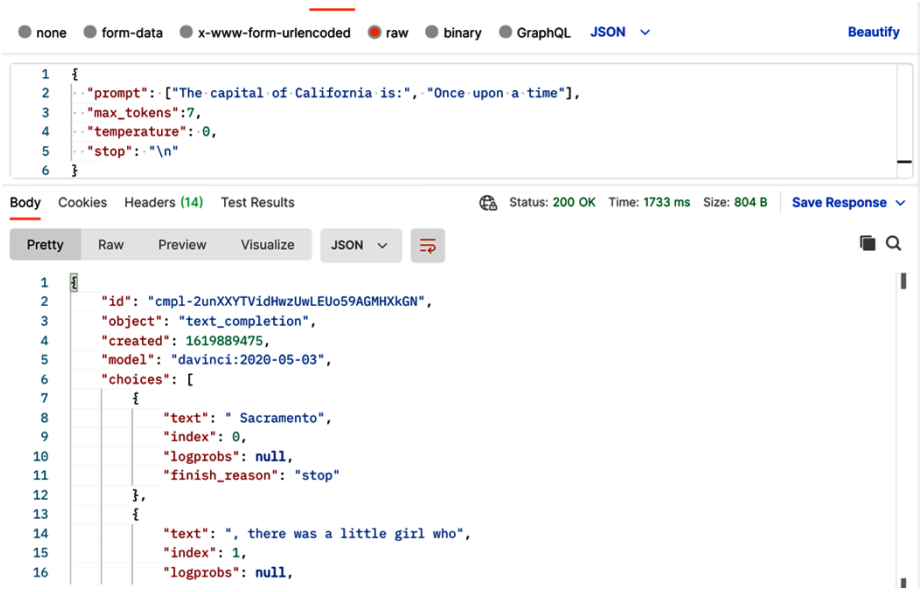

$ curl https://api.openai.com/v1/engines \
> -H 'Authorization: Bearer sk-
> -H 'OpenAI-Organization: org-
{
"object": "list",
"data": [
{
"id": "ada",
"object": "engine",
"created": null,
"max_replicas": null,
"owner": "openai",
"permissions": null,
"ready": true,
"ready_replicas": null,
"replicas": null
},
{
"id": "babbage",
"object": "engine",
"created": null,
"max_replicas": null,
"owner": "openai",
"permissions": null,
"ready": true,
"ready_replicas": null,
"replicas": null
},
{
"id": "content-filter-alpha-c4",
"object": "engine",
"created": null,
"max_replicas": null,
"owner": "openai",
"permissions": null,
"ready": true,
"ready_replicas": null,
"replicas": null
},

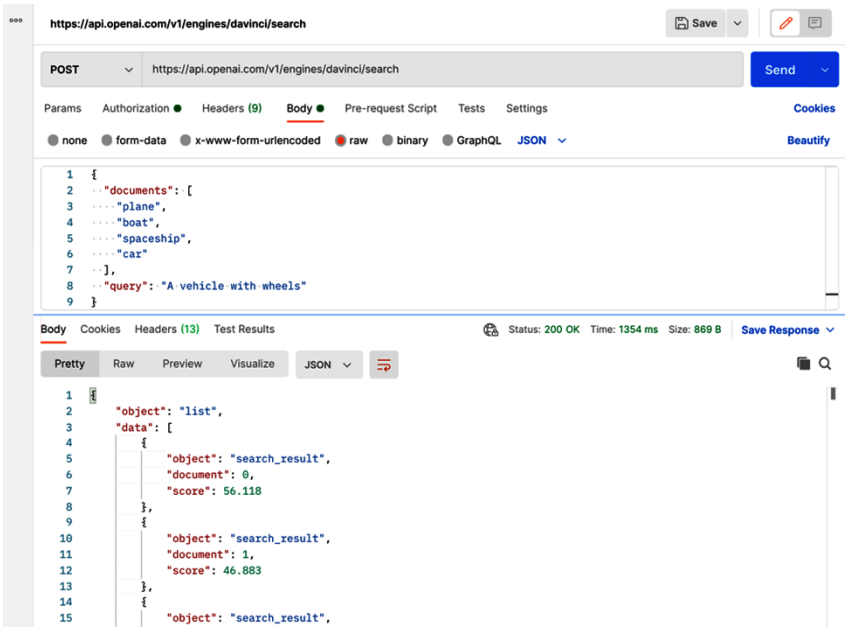

1
POST
https://api.openai.com/v1/engines/davinci/completions
2
Send
Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
3
4
5
{
"prompt": "Once upon a time"
}
Body
Cookies
Headers (14)
Test Results
200 OK
1615 ms
771 B
Save Response


POST
https://api.openai.com/v1/engines/davinci/completions
Send
Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"prompt": "Once upon a time"
}
Body
Cookies
Headers (14)
Test Results
Status: 200 OK
Time: 1615 ms
Size: 771 B
Save Response
Pretty
Raw
Preview
Visualize
JSON
{
"id": "cmpl-2un9J1aeZKwImbKhWAdwwECZipXVA",
"object": "text_completion",
"created": 1619887973,
"model": "davinci:2020-05-03",
"choices": [
{
"text": " on the dangerous side of the tracks, the Wonderbolts were your guardians of",
"index": 0,
"logprobs": null,
"finish_reason": "length"
}
]
}

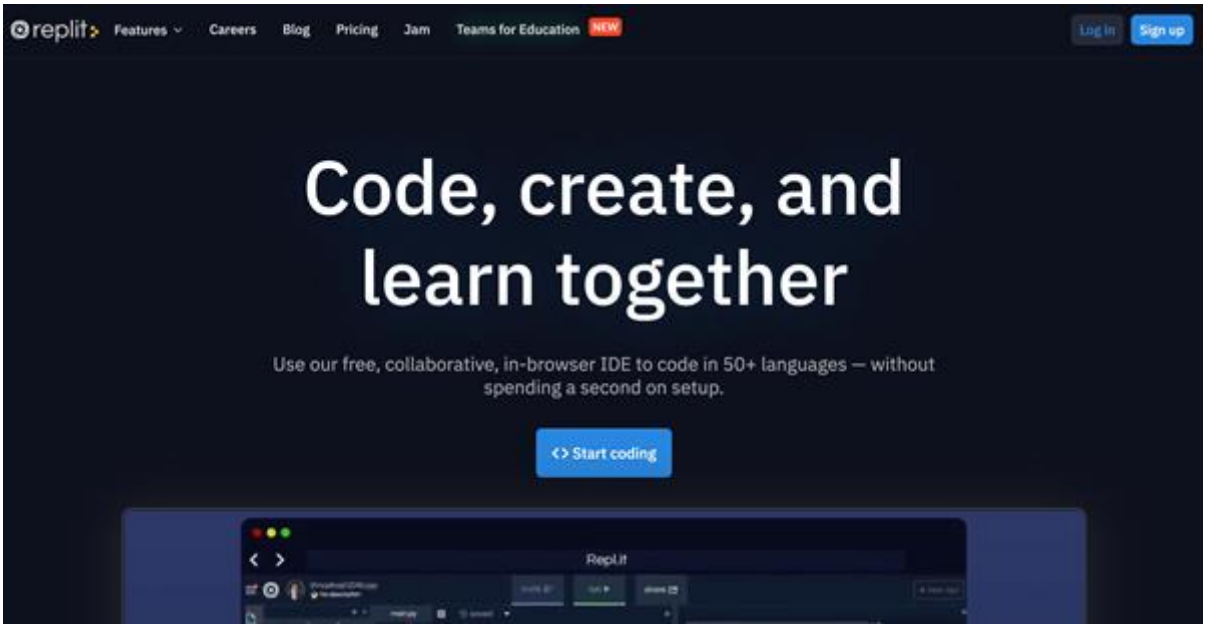

POST
https://api.openai.com/v1/engines/davinci/completions
Send
Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"prompt": "Once upon a time"
}
Body
Cookies
Headers (14)
Test Results
Status: 200 OK
Time: 1615 ms
Size: 771 B
Save Response
Pretty
Raw
Preview
Visualize
{"id": "cmpl-2un9J1aeZKwImbKhWAdwwECZipXVA", "object": "text_completion", "created": 1619887973, "model": "davinci:2020-05-03", "choices": [
{"text": " on the dangerous side of the tracks, the Wonderbolts were your guardians of", "index": 0, "logprobs": null, "finish_reason": "length"}]
}

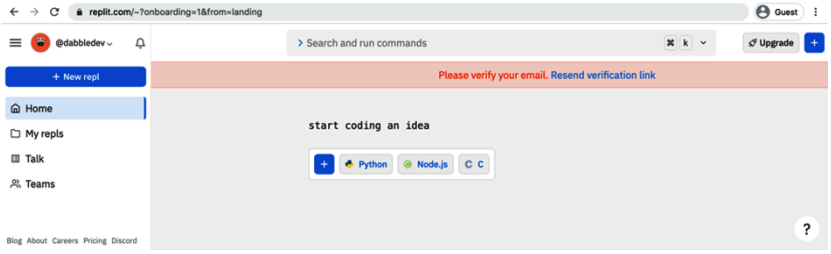

POST
https://api.openai.com/v1/engines/davinci/completions
Send
Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"prompt": "Once upon a time",
"max_tokens": 4
}
Body
Cookies
Headers (14)
Test Results
Status: 200 OK
Time: 1615 ms
Size: 771 B
Save Response

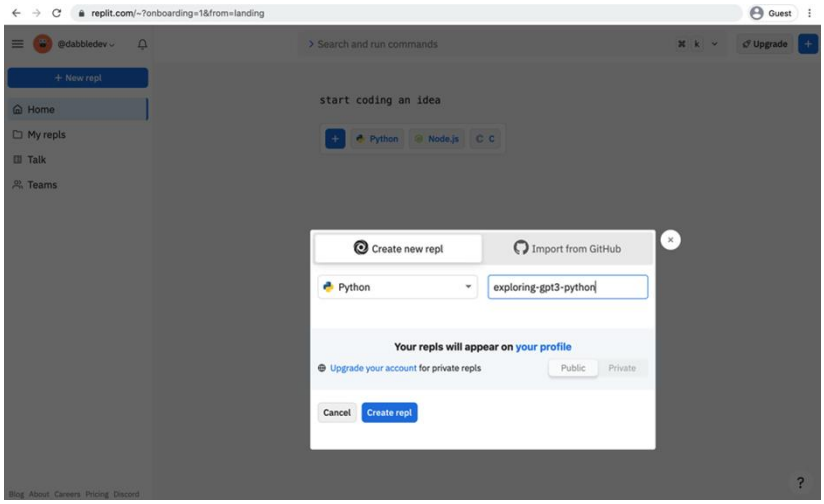

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify
{
"prompt": ["The capital of California is:", "Once upon a time"],
"max_tokens":7,
"temperature": 0,
"stop": "\n"
}
Body Cookies Headers (14) Test Results Status: 200 OK Time: 1733 ms Size: 804 B Save Response
Pretty Raw Preview Visualize JSON
{
"id": "cmpl-2unXXYTVidHwzUwLEUo59AGMHXkGN",
"object": "text_completion",
"created": 1619889475,
"model": "davinci:2020-05-03",
"choices": [
{
"text": " Sacramento",
"index": 0,
"logprobs": null,
"finish_reason": "stop"
},
{
"text": ", there was a little girl who",
"index": 1,
"logprobs": null,


https://api.openai.com/v1/engines/davinci/search
Save
POST https://api.openai.com/v1/engines/davinci/search Send
Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify
{
"documents": [
"plane",
"boat",
"spaceship",
"car"
],
"query": "A vehicle with wheels"
}
Body Cookies Headers (13) Test Results Status: 200 OK Time: 1354 ms Size: 869 B Save Response
Pretty Raw Preview Visualize JSON
{
"object": "list",
"data": [
{
"object": "search_result",
"document": 0,
"score": 56.118
},
{
"object": "search_result",
"document": 1,
"score": 46.883
},
{
"object": "search_result",

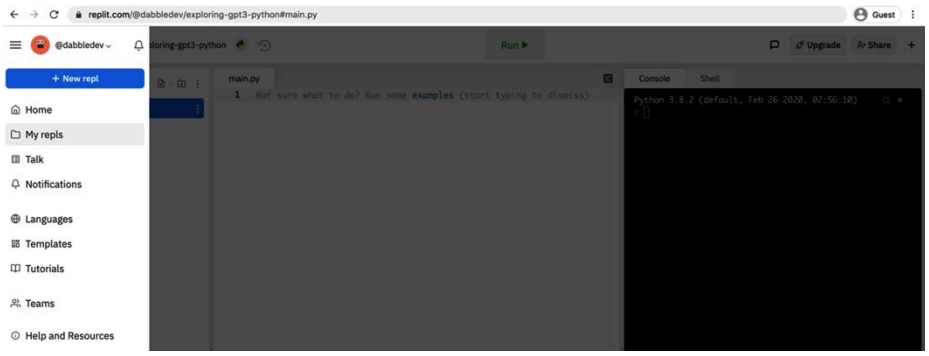
Chapter 5: Calling the OpenAI API in Code

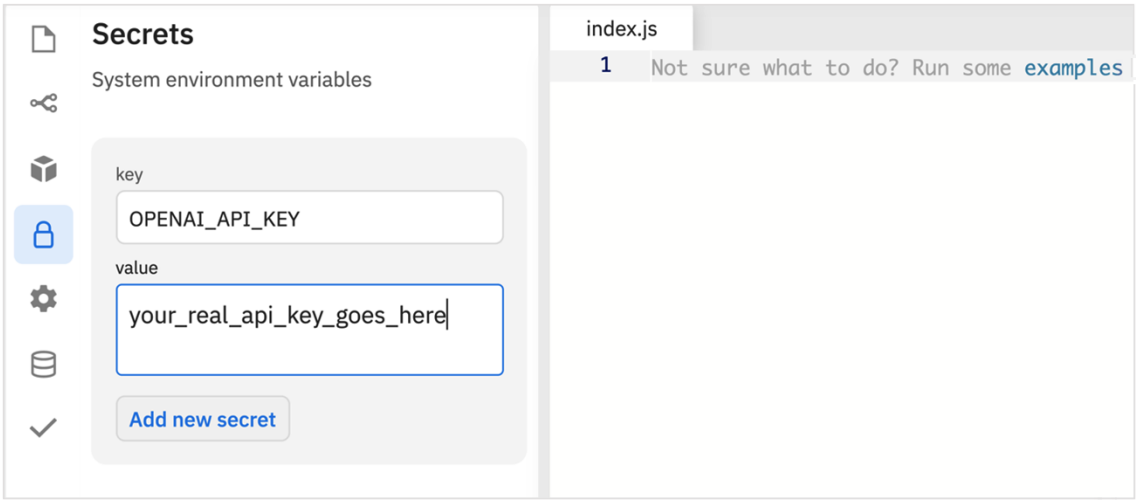


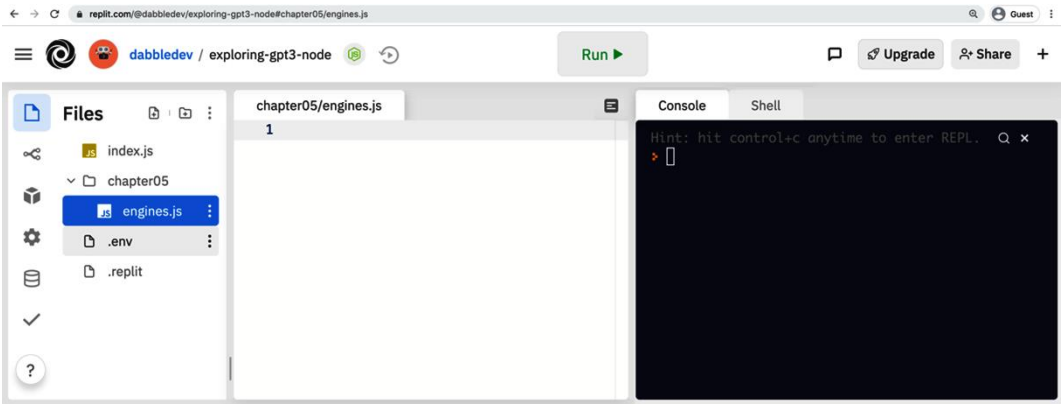


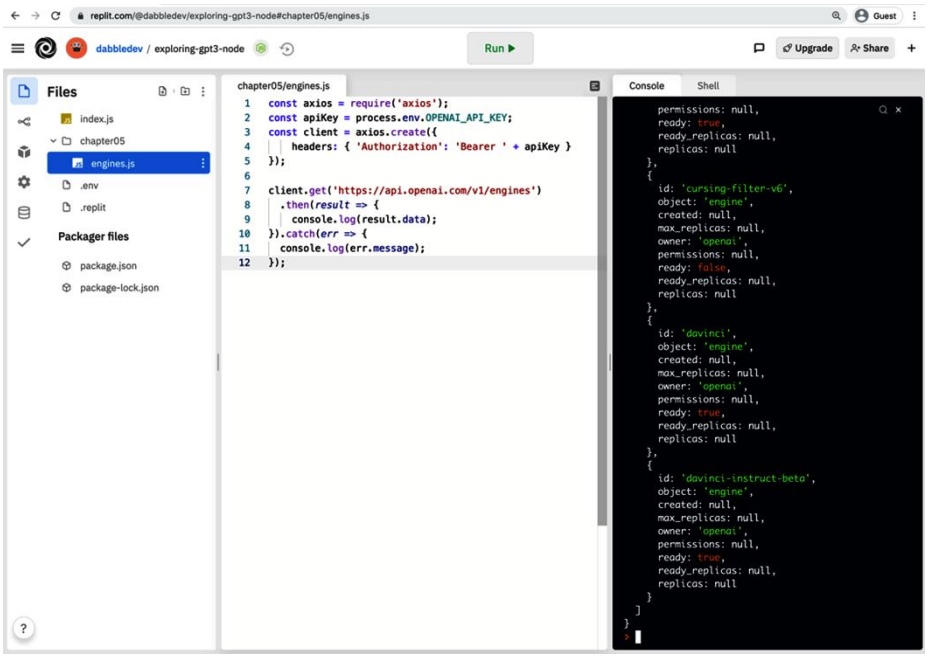


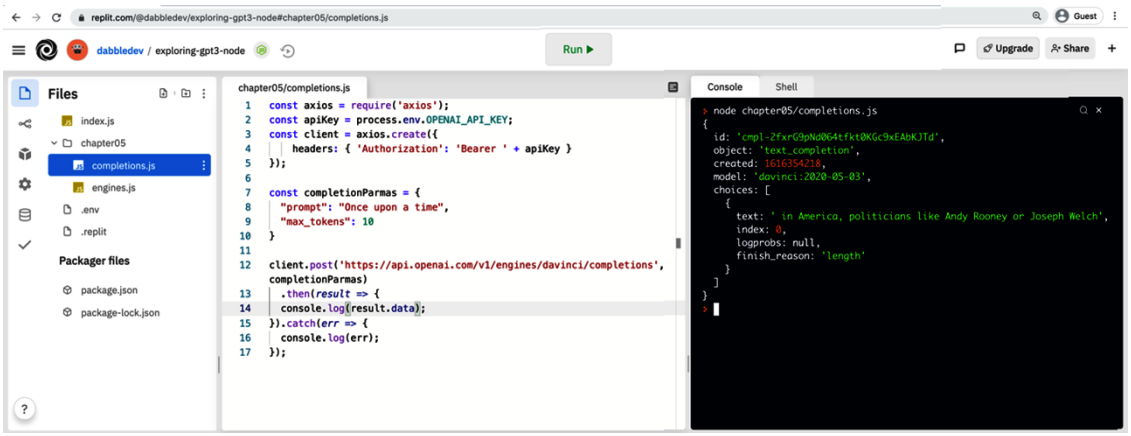

replit.com/@dabbledev/exploring-gpt3-python#main.py
Guest
dabbledev / exploring-gpt3-python
Run
Upgrade
Share
Files
main.py
Not sure what to do? Run some examples (start typing to dismiss)
Console
Shell
Python 3.8.2 (default, Feb 26 2020, 02:56:10)

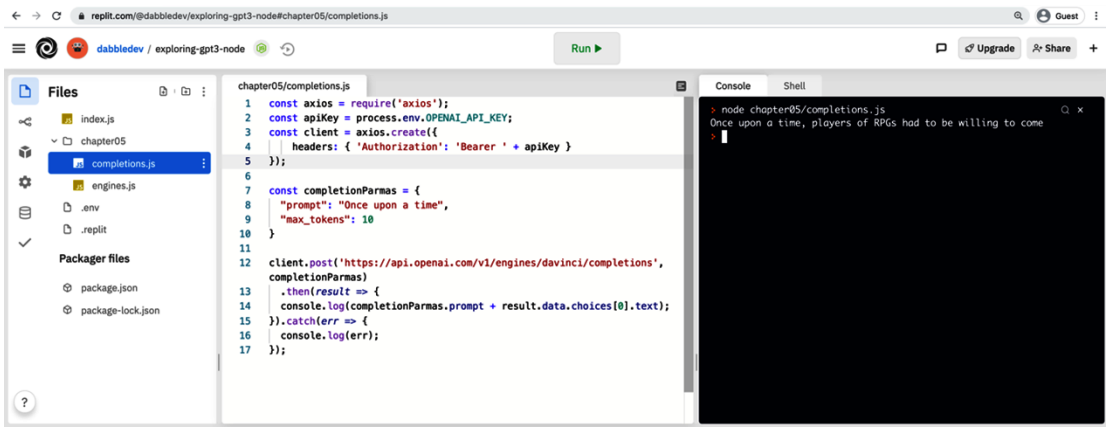

replit.com/@dabbledev/exploring-gpt3-python#main.py
Guest
@dabbledev
Run
Upgrade
Share
+ New repl
Home
My repls
Talk
Notifications
Languages
Templates
Tutorials
Teams
Help and Resources
main.py
Not sure what to do? Run some examples (start typing to dismiss)
Console
Shell
Python 3.8.2 (default, Feb 26 2020, 02:56:10)

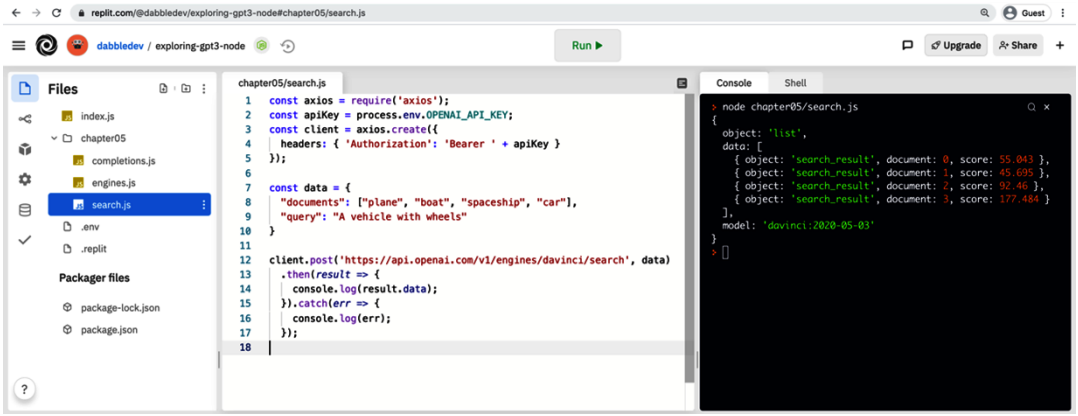

Secrets
System environment variables
key
OPENAI_API_KEY
value
your_real_api_key_goes_here
Add new secret
index.js
1
Not sure what to do? Run some examples

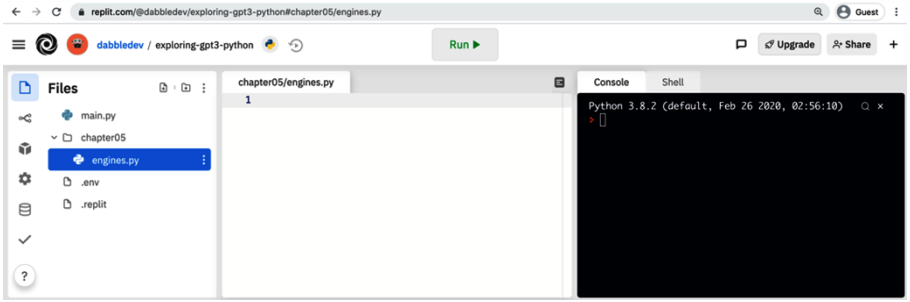

replit.com/@dabbledev/exploring-gpt3-node#chapter05/engines.js
Guest
dabbledev / exploring-gpt3-node
Run
Upgrade
Share
Files
index.js
chapter05
engines.js
.env
.replit
chapter05/engines.js
1
Console
Shell
Hint: hit control+c anytime to enter REPL.

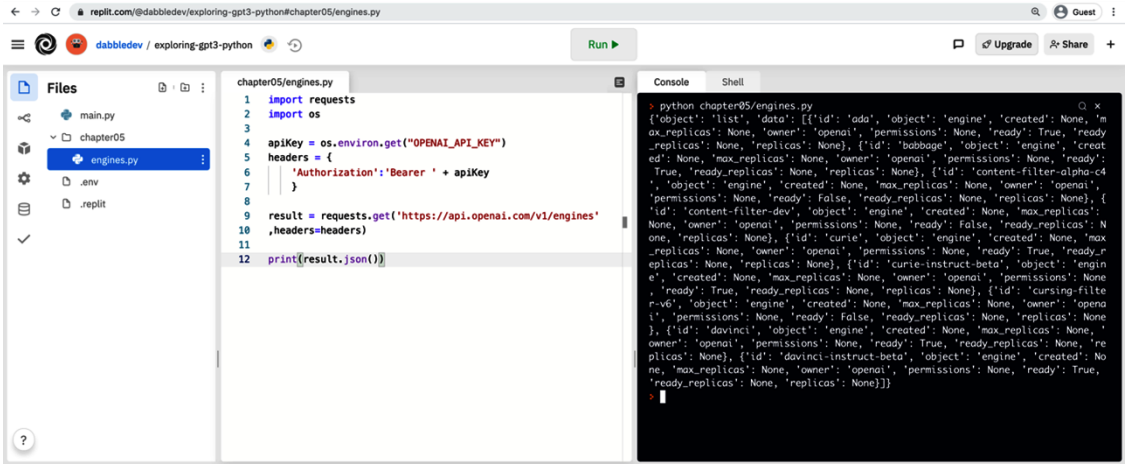

replit.com/@dabbledev/exploring-gpt3-node#chapter05/engines.js
dabbledev / exploring-gpt3-node
Run
Upgrade
Share
Files
index.js
chapter05
engines.js
.env
.replit
Packager files
package.json
package-lock.json
chapter05/engines.js
const axios = require('axios');
const apiKey = process.env.OPENAI_API_KEY;
const client = axios.create({
headers: { 'Authorization': 'Bearer ' + apiKey }
});
client.get('https://api.openai.com/v1/engines')
.then(result => {
console.log(result.data);
}).catch(err => {
console.log(err.message);
});
Console
Shell
permissions: null,
ready: true,
ready_replicas: null,
replicas: null
},
{
id: 'cursing-filter-v6',
object: 'engine',
created: null,
max_replicas: null,
owner: 'openai',
permissions: null,
ready: false,
ready_replicas: null,
replicas: null
},
{
id: 'davinci',
object: 'engine',
created: null,
max_replicas: null,
owner: 'openai',
permissions: null,
ready: true,
ready_replicas: null,
replicas: null
},
{
id: 'davinci-instruct-beta',
object: 'engine',
created: null,
max_replicas: null,
owner: 'openai',
permissions: null,
ready: true,
ready_replicas: null,
replicas: null
}
]
}

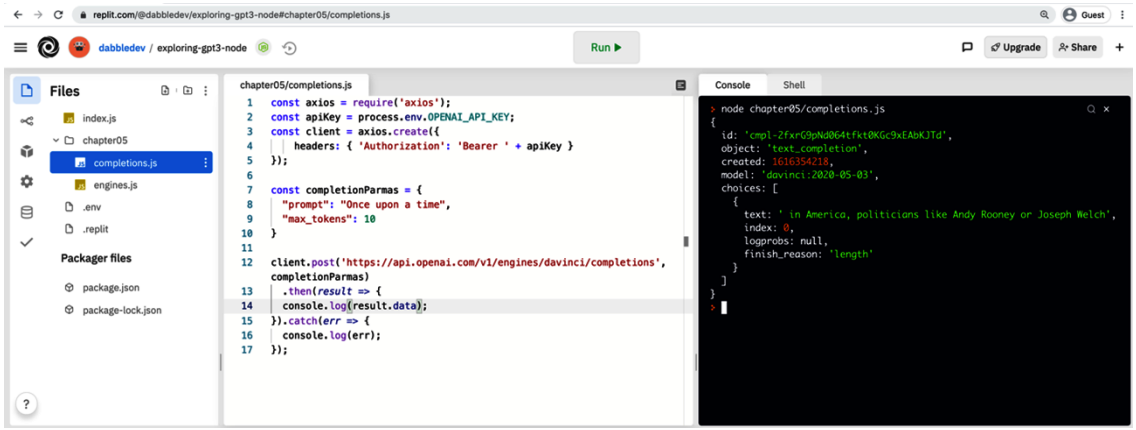

replit.com/@dabbledev/exploring-gpt3-node#chapter05/completions.js
dabbledev / exploring-gpt3-node
Run
Upgrade
Share
Files
index.js
chapter05
completions.js
engines.js
.env
.replit
Packager files
package.json
package-lock.json
chapter05/completions.js
const axios = require('axios');
const apiKey = process.env.OPENAI_API_KEY;
const client = axios.create({
headers: { 'Authorization': 'Bearer ' + apiKey }
});
const completionParmas = {
"prompt": "Once upon a time",
"max_tokens": 10
}
client.post('https://api.openai.com/v1/engines/davinci/completions',
completionParmas)
.then(result => {
console.log(result.data);
}).catch(err => {
console.log(err);
});
Console
Shell
> node chapter05/completions.js
{
id: 'cmpl-2fxrG9pNd064tfkt0KGc9xEAbKJTd',
object: 'text_completion',
created: 1616354218,
model: 'davinci:2020-05-03',
choices: [
{
text: ' in America, politicians like Andy Rooney or Joseph Welch',
index: 0,
logprobs: null,
finish_reason: 'length'
}
]
}

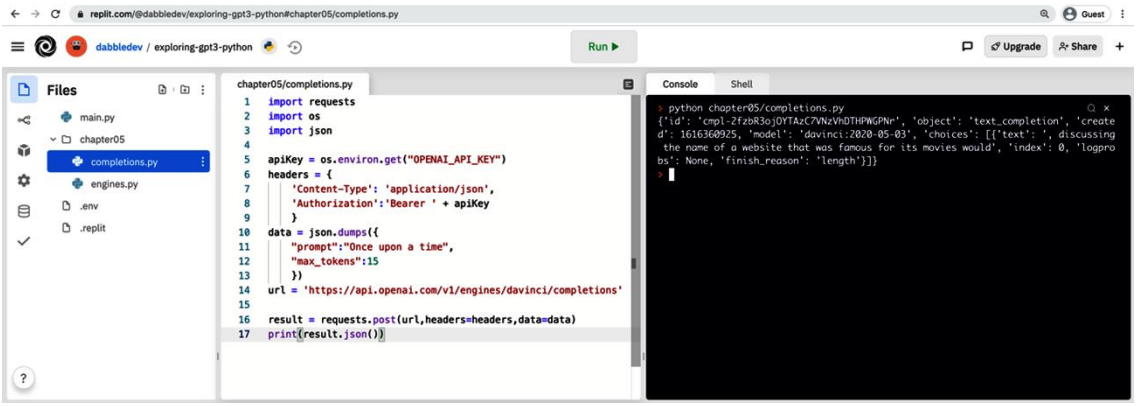

replit.com/@dabbledev/exploring-gpt3-node#chapter05/completions.js
dabbledev / exploring-gpt3-node
Run
Upgrade
Share
Files
index.js
chapter05
completions.js
engines.js
.env
.replit
Packager files
package.json
package-lock.json
chapter05/completions.js
const axios = require('axios');
const apiKey = process.env.OPENAI_API_KEY;
const client = axios.create({
headers: { 'Authorization': 'Bearer ' + apiKey }
});
const completionParmas = {
"prompt": "Once upon a time",
"max_tokens": 10
}
client.post('https://api.openai.com/v1/engines/davinci/completions',
completionParmas)
.then(result => {
console.log(completionParmas.prompt + result.data.choices[0].text);
}).catch(err => {
console.log(err);
});
Console
Shell
> node chapter05/completions.js
Once upon a time, players of RPGs had to be willing to come

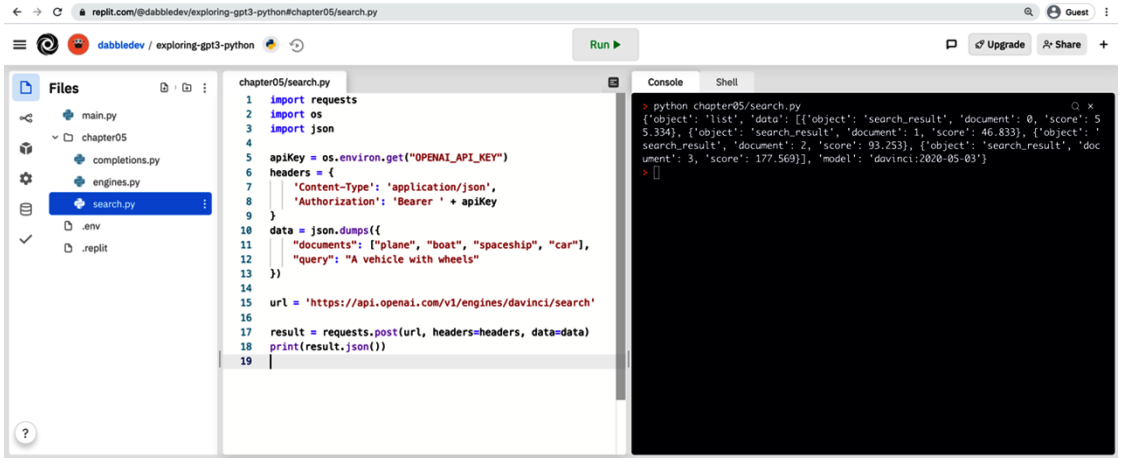

replit.com/@dabbledev/exploring-gpt3-node#chapter05/search.js
dabbledev / exploring-gpt3-node
Run
Upgrade
Share
Files
index.js
chapter05
completions.js
engines.js
search.js
.env
.replit
Packager files
package-lock.json
package.json
chapter05/search.js
const axios = require('axios');
const apiKey = process.env.OPENAI_API_KEY;
const client = axios.create({
headers: { 'Authorization': 'Bearer ' + apiKey }
});
const data = {
"documents": ["plane", "boat", "spaceship", "car"],
"query": "A vehicle with wheels"
}
client.post('https://api.openai.com/v1/engines/davinci/search', data)
.then(result => {
console.log(result.data);
}).catch(err => {
console.log(err);
});
Console
Shell
> node chapter05/search.js
{
object: 'list',
data: [
{ object: 'search_result', document: 0, score: 55.043 },
{ object: 'search_result', document: 1, score: 45.695 },
{ object: 'search_result', document: 2, score: 92.46 },
{ object: 'search_result', document: 3, score: 177.484 }
],
model: 'davinci:2020-05-03'
}


replit.com/@dabbledev/exploring-gpt3-python#chapter05/engines.py
dabbledev / exploring-gpt3-python
Run
Upgrade
Share
Files
main.py
chapter05
engines.py
.env
.replit
chapter05/engines.py
Console
Shell
Python 3.8.2 (default, Feb 26 2020, 02:56:10)

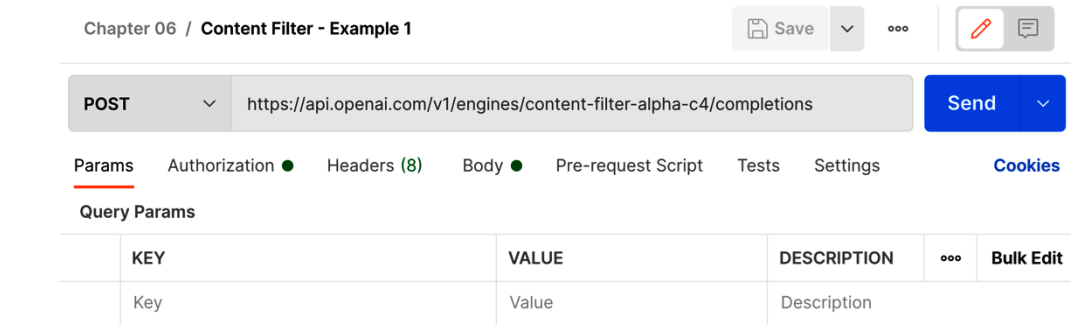

replit.com/@dabbledev/exploring-gpt3-python#chapter05/engines.py
dabbledev / exploring-gpt3-python
Run
Upgrade
Share
Files
main.py
chapter05
engines.py
.env
.replit
chapter05/engines.py
import requests
import os
apiKey = os.environ.get("OPENAI_API_KEY")
headers = {
'Authorization':'Bearer ' + apiKey
}
result = requests.get('https://api.openai.com/v1/engines'
,headers=headers)
print(result.json())
Console
Shell
> python chapter05/engines.py
{'object': 'list', 'data': [{'id': 'ada', 'object': 'engine', 'created': None, 'max_replicas': None, 'owner': 'openai', 'permissions': None, 'ready': True, 'ready_replicas': None, 'replicas': None}, {'id': 'babbage', 'object': 'engine', 'created': None, 'max_replicas': None, 'owner': 'openai', 'permissions': None, 'ready': True, 'ready_replicas': None, 'replicas': None}, {'id': 'content-filter-alpha-c4', 'object': 'engine', 'created': None, 'max_replicas': None, 'owner': 'openai', 'permissions': None, 'ready': False, 'ready_replicas': None, 'replicas': None}, {'id': 'content-filter-dev', 'object': 'engine', 'created': None, 'max_replicas': None, 'owner': 'openai', 'permissions': None, 'ready': False, 'ready_replicas': None, 'replicas': None}, {'id': 'curie', 'object': 'engine', 'created': None, 'max_replicas': None, 'owner': 'openai', 'permissions': None, 'ready': True, 'ready_replicas': None, 'replicas': None}, {'id': 'curie-instruct-beta', 'object': 'engine', 'created': None, 'max_replicas': None, 'owner': 'openai', 'permissions': None, 'ready': True, 'ready_replicas': None, 'replicas': None}, {'id': 'cursing-filter-v6', 'object': 'engine', 'created': None, 'max_replicas': None, 'owner': 'openai', 'permissions': None, 'ready': False, 'ready_replicas': None, 'replicas': None}, {'id': 'davinci', 'object': 'engine', 'created': None, 'max_replicas': None, 'owner': 'openai', 'permissions': None, 'ready': True, 'ready_replicas': None, 'replicas': None}, {'id': 'davinci-instruct-beta', 'object': 'engine', 'created': None, 'max_replicas': None, 'owner': 'openai', 'permissions': None, 'ready': True, 'ready_replicas': None, 'replicas': None}]}

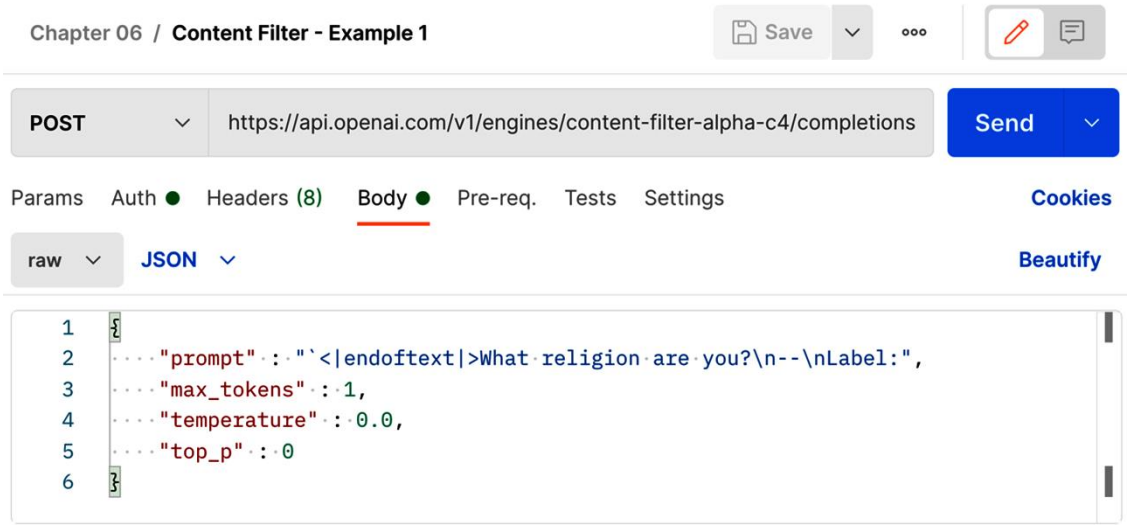

replit.com/@dabbledev/exploring-gpt3-node#chapter05/completions.js
dabbledev / exploring-gpt3-node
Run
Upgrade
Share
Files
index.js
chapter05
completions.js
engines.js
.env
.replit
Packager files
package.json
package-lock.json
chapter05/completions.js
const axios = require('axios');
const apiKey = process.env.OPENAI_API_KEY;
const client = axios.create({
headers: { 'Authorization': 'Bearer ' + apiKey }
});
const completionParmas = {
"prompt": "Once upon a time",
"max_tokens": 10
}
client.post('https://api.openai.com/v1/engines/davinci/completions',
completionParmas)
.then(result => {
console.log(result.data);
}).catch(err => {
console.log(err);
});
Console
Shell
> node chapter05/completions.js
{
id: 'cmpl-2fxrG9pNd064tfkt0KGc9xEAbKJTd',
object: 'text_completion',
created: 1616354218,
model: 'davinci:2020-05-03',
choices: [
{
text: ' in America, politicians like Andy Rooney or Joseph Welch',
index: 0,
logprobs: null,
finish_reason: 'length'
}
]
}

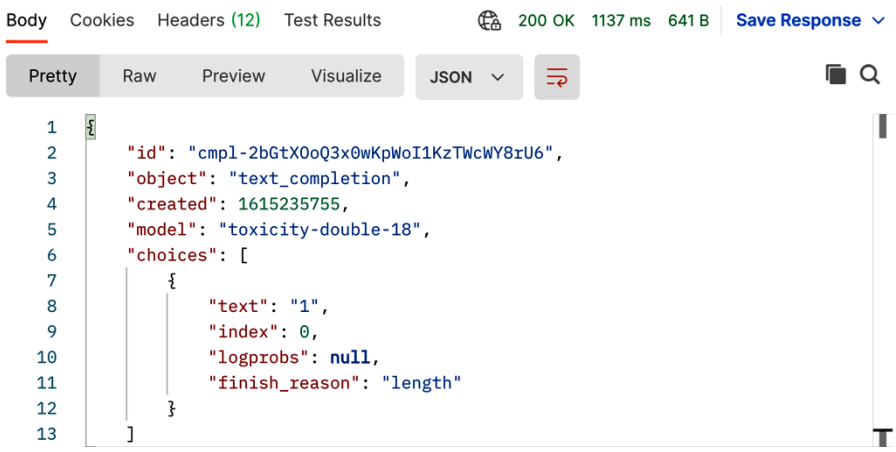

replit.com/@dabbledev/exploring-gpt3-python#chapter05/completions.py
Guest
dabbledev / exploring-gpt3-python
Run
Upgrade
Share
Files
main.py
chapter05
completions.py
engines.py
.env
.replit
chapter05/completions.py
import requests
import os
import json
apiKey = os.environ.get("OPENAI_API_KEY")
headers = {
'Content-Type': 'application/json',
'Authorization':'Bearer ' + apiKey
}
data = json.dumps({
"prompt":"Once upon a time",
"max_tokens":15
})
url = 'https://api.openai.com/v1/engines/davinci/completions'
result = requests.post(url,headers=headers,data=data)
print(result.json())
Console
Shell
> python chapter05/completions.py
{'id': 'cmpl-2fzbR3ojOYTAzC7VNzVhDTHPWGPNr', 'object': 'text_completion', 'created': 1616360925, 'model': 'davinci:2020-05-03', 'choices': [{'text': ', discussing the name of a website that was famous for its movies would', 'index': 0, 'logprobs': None, 'finish_reason': 'length'}]}
>

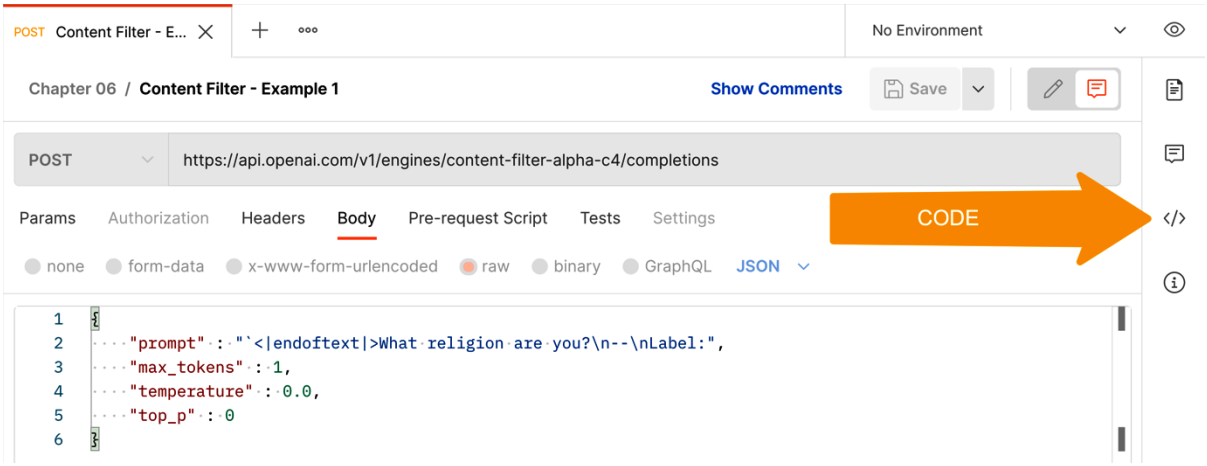

replit.com/@dabbledev/exploring-gpt3-python#chapter05/search.py
Guest
dabbledev / exploring-gpt3-python
Run
Upgrade
Share
Files
main.py
chapter05
completions.py
engines.py
search.py
.env
.replit
chapter05/search.py
import requests
import os
import json
apiKey = os.environ.get("OPENAI_API_KEY")
headers = {
'Content-Type': 'application/json',
'Authorization': 'Bearer ' + apiKey
}
data = json.dumps({
"documents": ["plane", "boat", "spaceship", "car"],
"query": "A vehicle with wheels"
})
url = 'https://api.openai.com/v1/engines/davinci/search'
result = requests.post(url, headers=headers, data=data)
print(result.json())
Console
Shell
> python chapter05/search.py
{'object': 'list', 'data': [{'object': 'search_result', 'document': 0, 'score': 55.334}, {'object': 'search_result', 'document': 1, 'score': 46.833}, {'object': 'search_result', 'document': 2, 'score': 93.253}, {'object': 'search_result', 'document': 3, 'score': 177.569}], 'model': 'davinci:2020-05-03'}
>

Chapter 6: Content Filtering







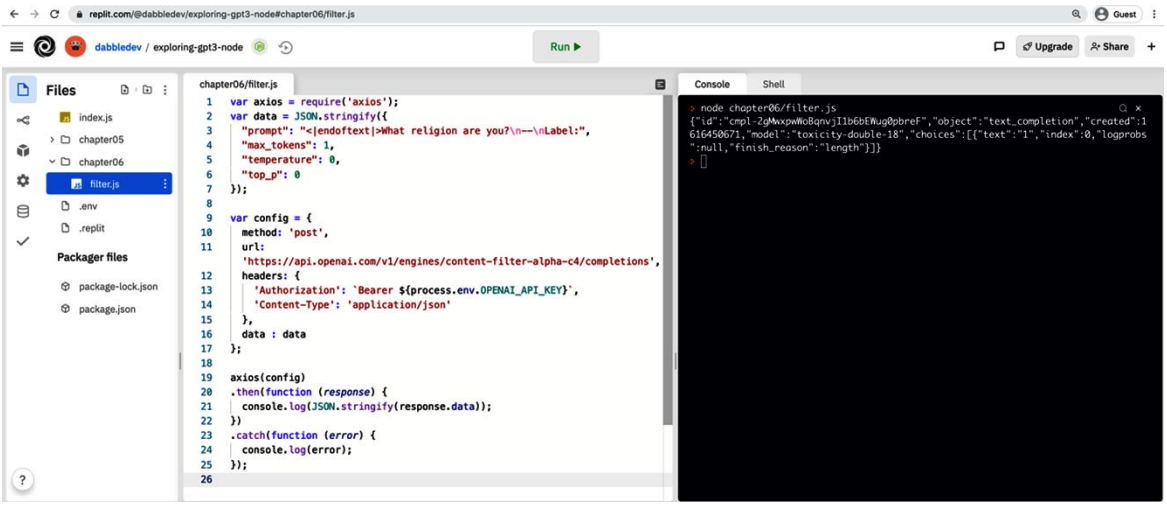


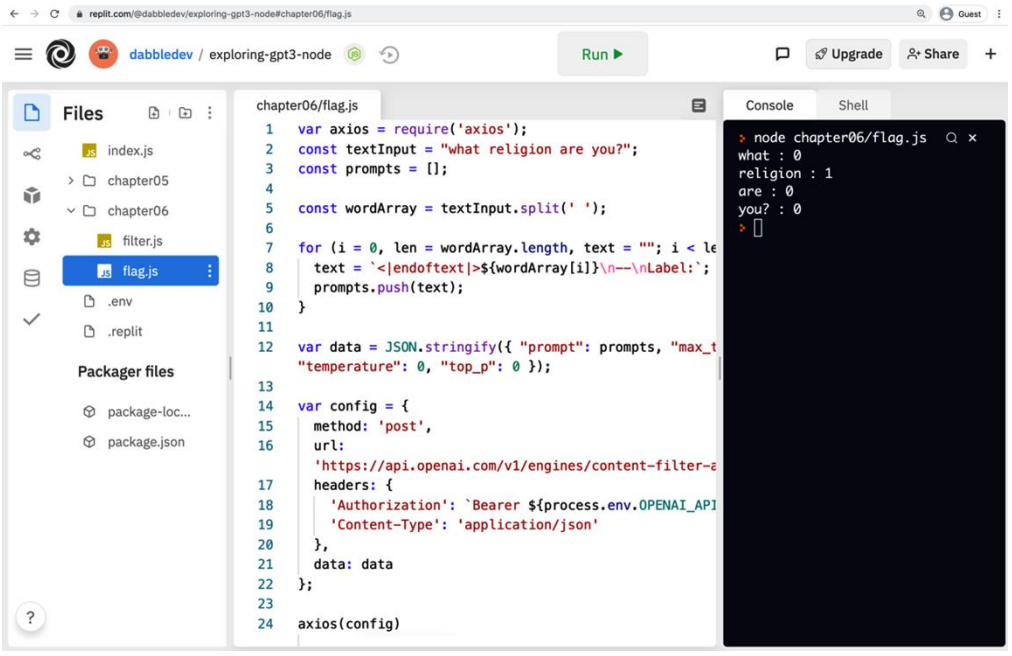
chapter06/filter.js

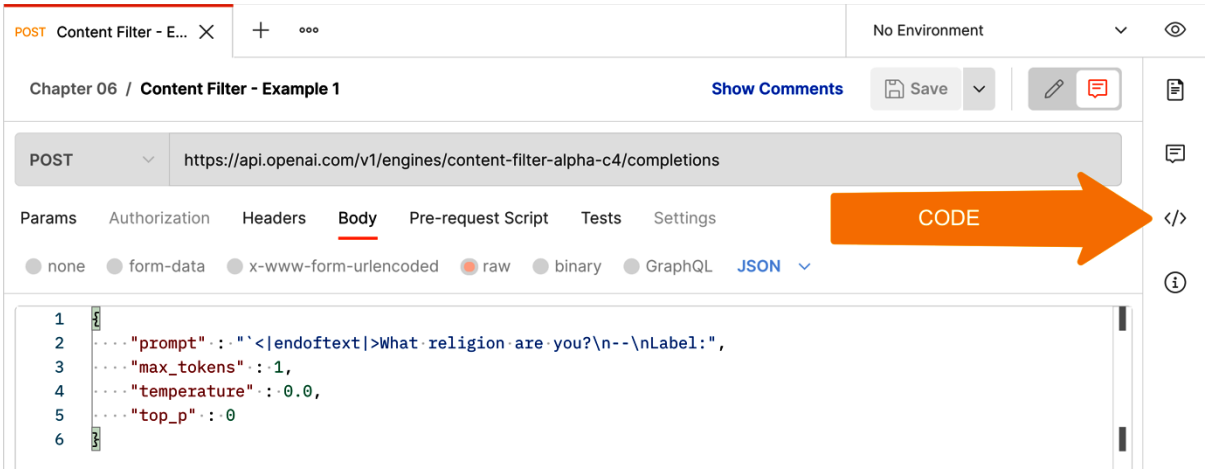
```
var axios = require('axios');
var data = JSON.stringify({"prompt":"`<|endoftext|>What religion are you?\n--\nLabel:","max_tokens":1,"temperature":0,"top_p":0});

var config = {
  method: 'post',
  url: 'https://api.openai.com/v1/engines/content-filter-alpha-c4/completions',
  headers: {
    'Authorization': 'Bearer {{OPENAI_API_KEY}}',
    'Content-Type': 'application/json'
  },
  data : data
};

axios(config)
.then(function (response) {
  console.log(JSON.stringify(response.data));
})
.catch(function (error) {
  console.log(error);
});
```

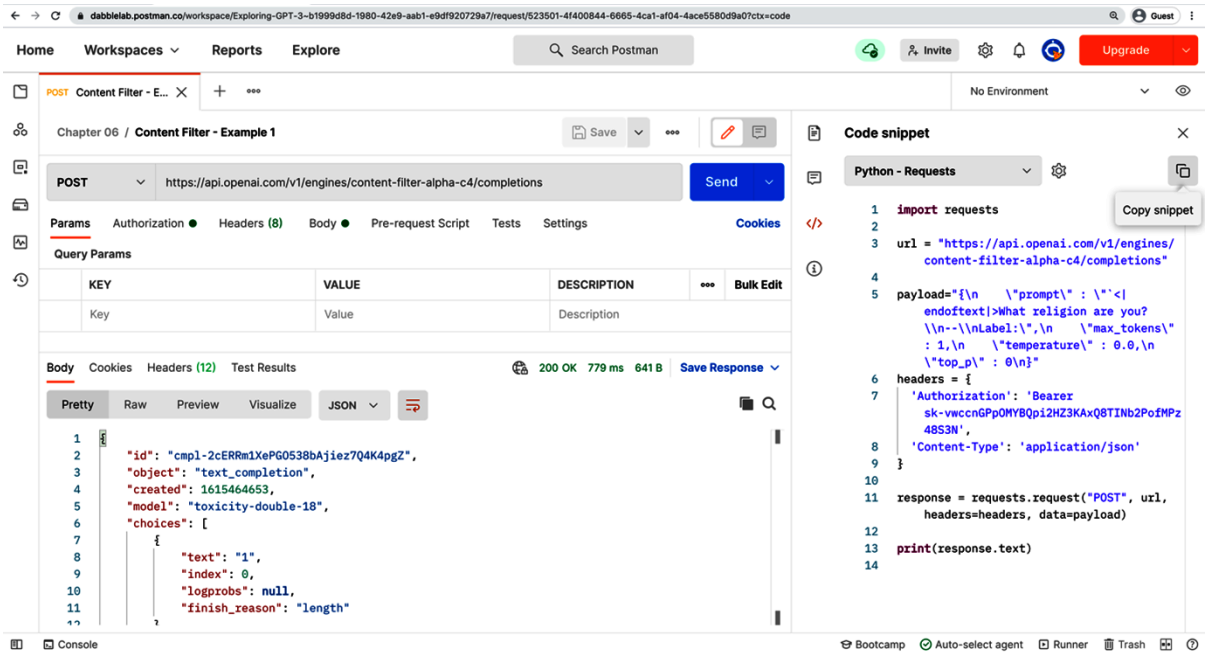
chapter06/filter.js

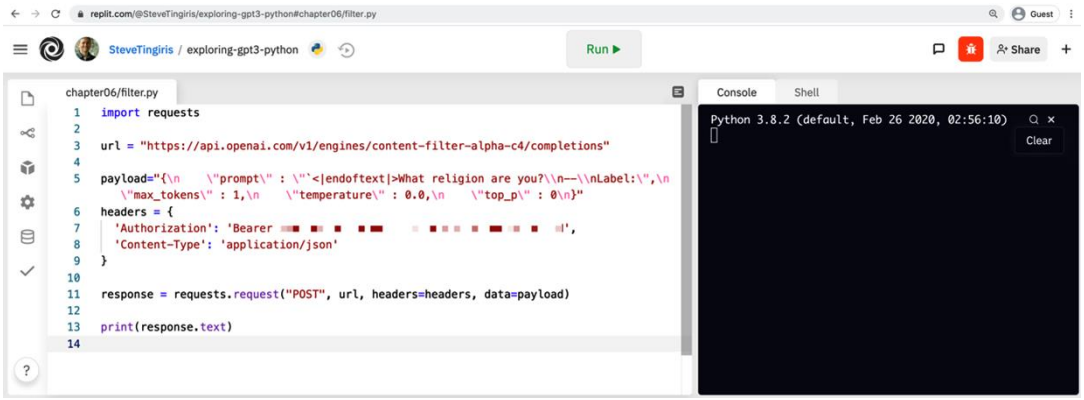
```
var axios = require('axios');
var data = JSON.stringify({"prompt":"`<|endoftext|>What religion are you?\n--\nLabel:","max_tokens":1,"temperature":0,"top_p":0});

var config = {
  method: 'post',
  url: 'https://api.openai.com/v1/engines/content-filter-alpha-c4/completions',
  headers: {
    'Authorization': `Bearer ${process.env.OPENAI_API_KEY}`,
    'Content-Type': 'application/json'
  },
  data : data
};

axios(config)
.then(function (response) {
  console.log(JSON.stringify(response.data));
})
.catch(function (error) {
  console.log(error);
});

```





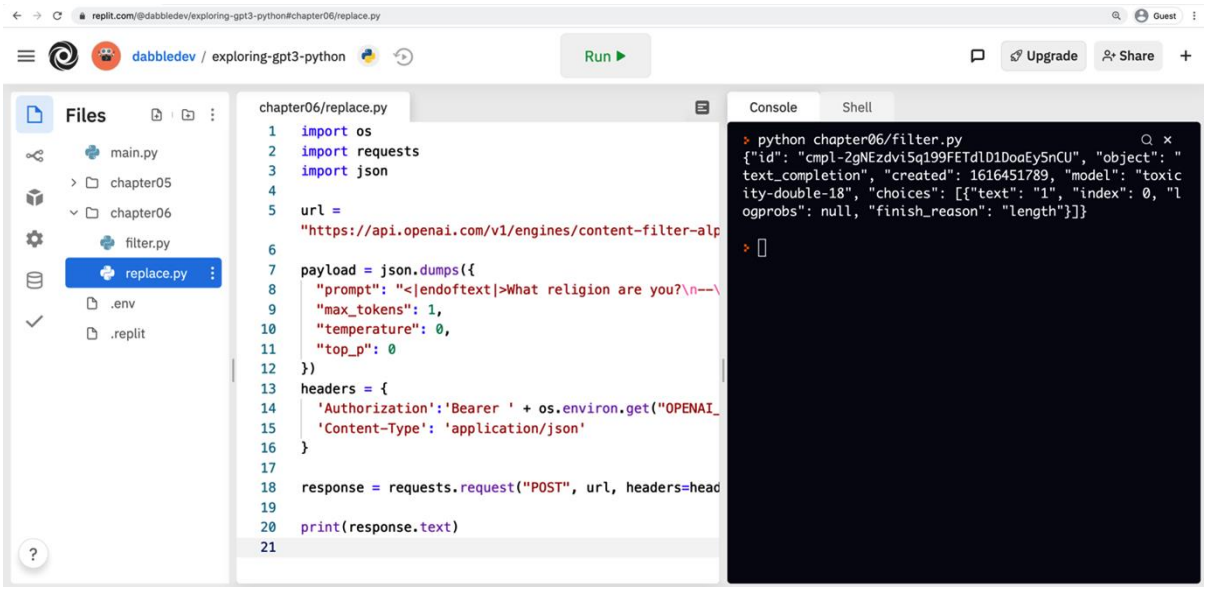

POST Content Filter - E...
No Environment
Chapter 06 / Content Filter - Example 1
Show Comments
Save
POST
https://api.openai.com/v1/engines/content-filter-alpha-c4/completions
Params
Authorization
Headers
Body
Pre-request Script
Tests
Settings
CODE
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
{
"prompt" : "`<|endoftext|>What religion are you?\n--\nLabel:",
"max_tokens" : 1,
"temperature" : 0.0,
"top_p" : 0
}

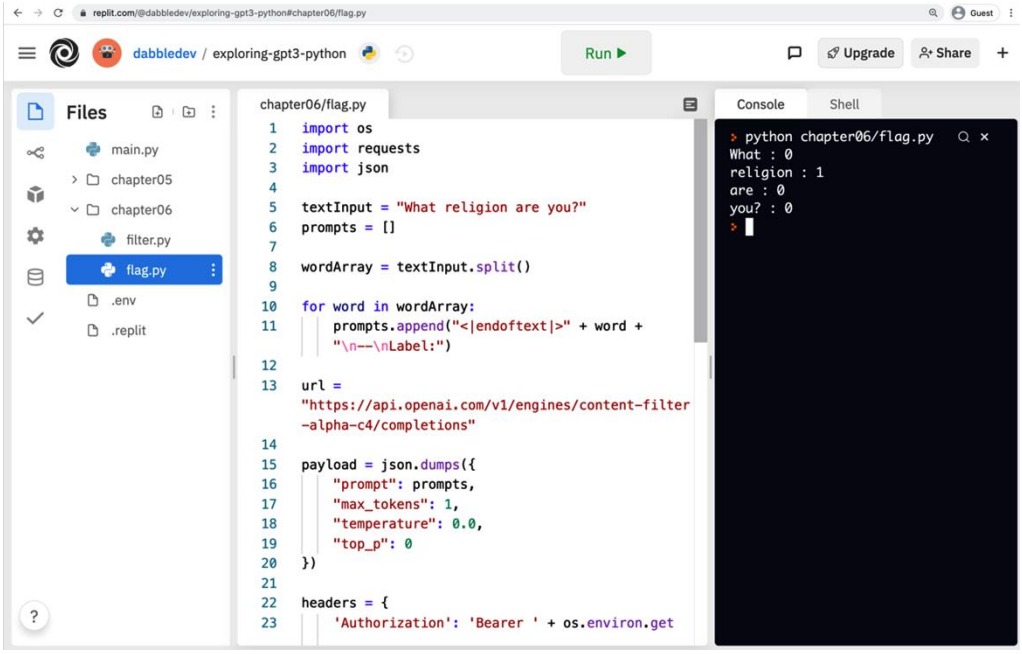

Home
Workspaces
Reports
Explore
Search Postman
Invite
Upgrade
Chapter 06 / Content Filter - Example 1
Save
POST
https://api.openai.com/v1/engines/content-filter-alpha-c4/completions
Send
Params
Authorization
Headers (8)
Body
Pre-request Script
Tests
Settings
Cookies
Query Params
KEY
VALUE
DESCRIPTION
Bulk Edit
Body
Cookies
Headers (12)
Test Results
200 OK
779 ms
641 B
Save Response
Pretty
Raw
Preview
Visualize
JSON
{
"id": "cmpl-2cERRm1XePGO538bAjiez7Q4K4pgZ",
"object": "text_completion",
"created": 1615464653,
"model": "toxicity-double-18",
"choices": [
{
"text": "1",
"index": 0,
"logprobs": null,
"finish_reason": "length"
Code snippet
Python - Requests
Copy snippet
import requests
url = "https://api.openai.com/v1/engines/content-filter-alpha-c4/completions"
payload="{\n \"prompt\" : \"`<|endoftext|>What religion are you?\\n--\\nLabel:\",\n \"max_tokens\" : 1,\n \"temperature\" : 0.0,\n \"top_p\" : 0\n}"
headers = {
'Authorization': 'Bearer sk-vwccnGPpOMYBQpi2HZ3KAxQ8TINb2PofMPz48S3N',
'Content-Type': 'application/json'
}
response = requests.request("POST", url, headers=headers, data=payload)
print(response.text)
Console
Bootcamp
Auto-select agent
Runner
Trash


SteveTingiris / exploring-gpt3-python
Run
Share
chapter06/filter.py
import requests
url = "https://api.openai.com/v1/engines/content-filter-alpha-c4/completions"
payload="{\n \"prompt\" : \"`<|endoftext|>What religion are you?\\n--\\nLabel:\",\n \"max_tokens\" : 1,\n \"temperature\" : 0.0,\n \"top_p\" : 0\n}"
headers = {
'Authorization': 'Bearer
'Content-Type': 'application/json'
}
response = requests.request("POST", url, headers=headers, data=payload)
print(response.text)
Console
Shell
Python 3.8.2 (default, Feb 26 2020, 02:56:10)
Clear

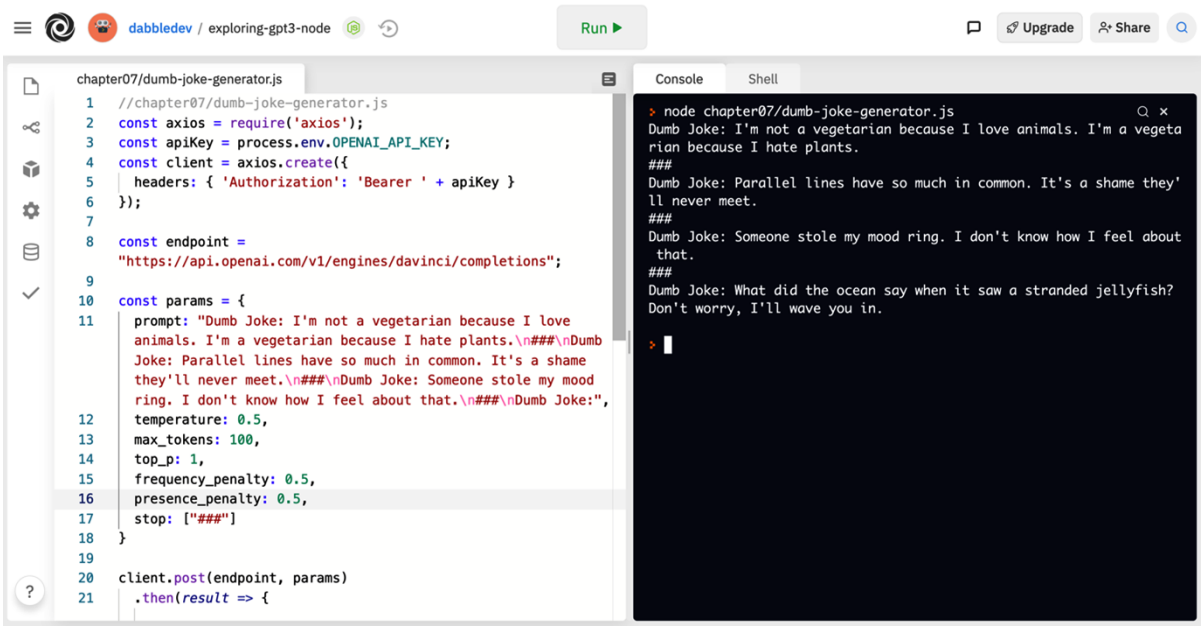
chapter06/filter.py

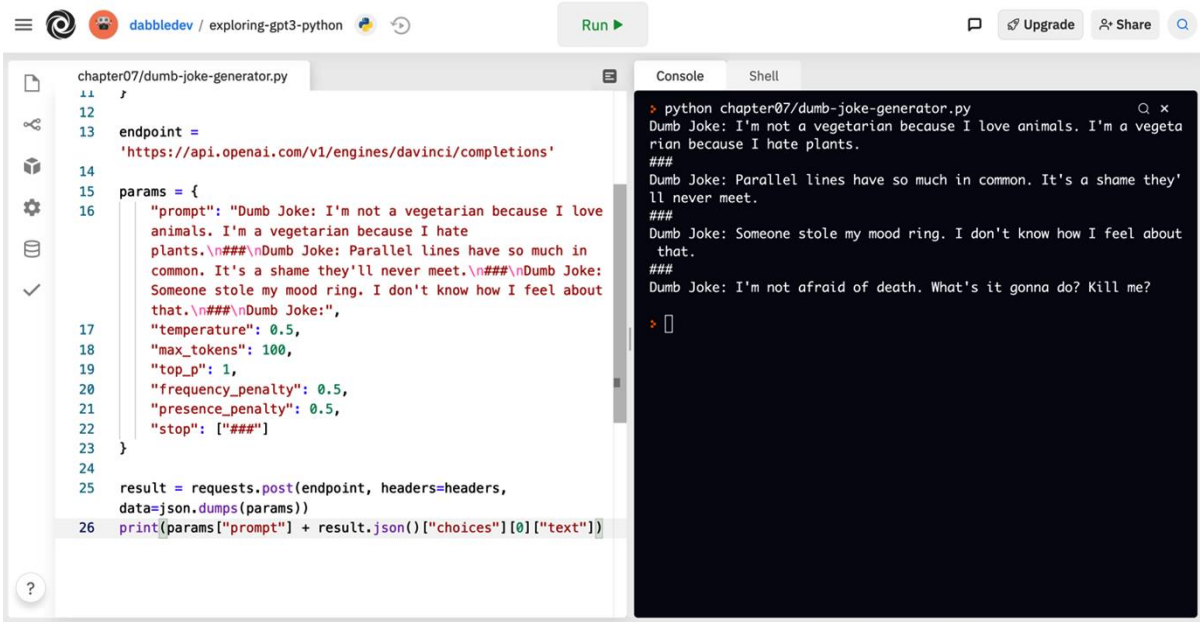
```python
import os
import requests

url = "https://api.openai.com/v1/engines/content-filter-alpha-c4/completions"

payload="{\n    \"prompt\" : \"`<|endoftext|>What religion are you?\\n--\\nLabel:\",\n \"max_tokens\" : 1,\n
headers = {
  'Authorization':'Bearer ' + os.environ.get("OPENAI_API_KEY"),
  'Content-Type': 'application/json'
}

response = requests.request("POST", url, headers=headers, data=payload)

print(response.text)
```

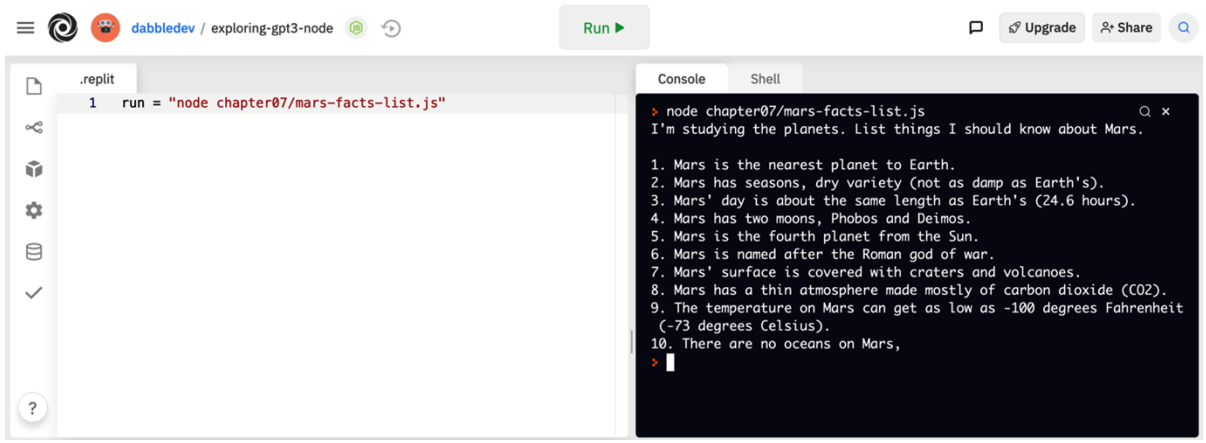


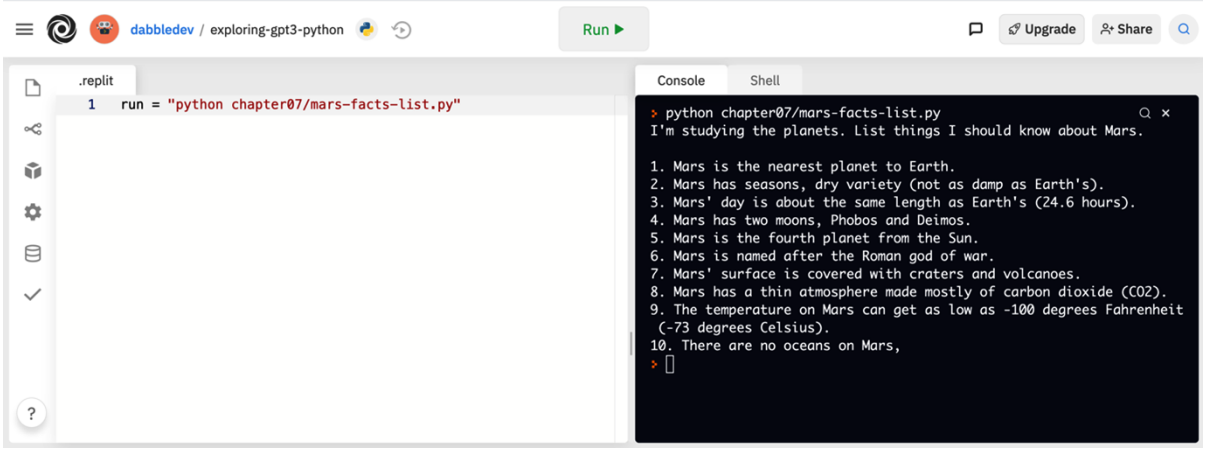


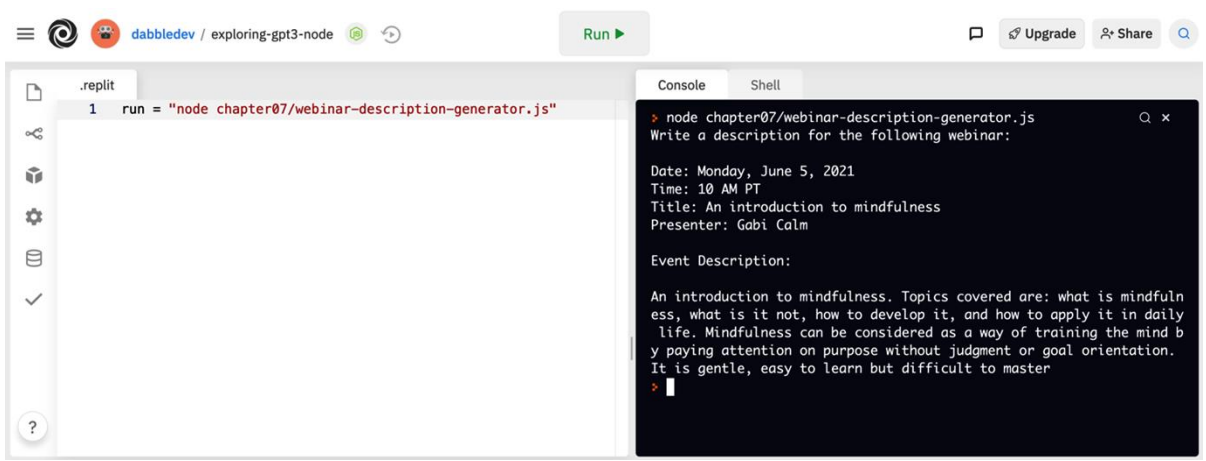
Chapter 7: Generating and Transforming Text

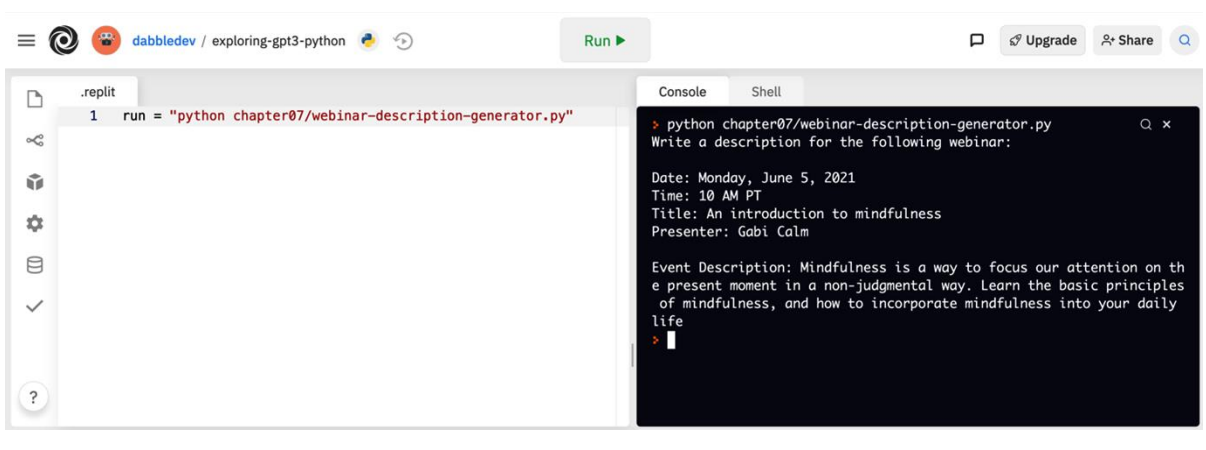


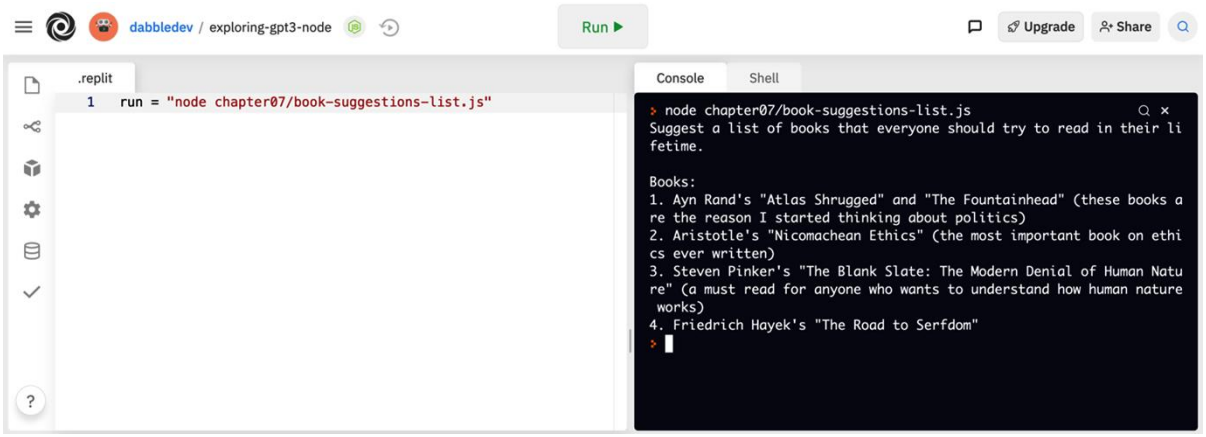


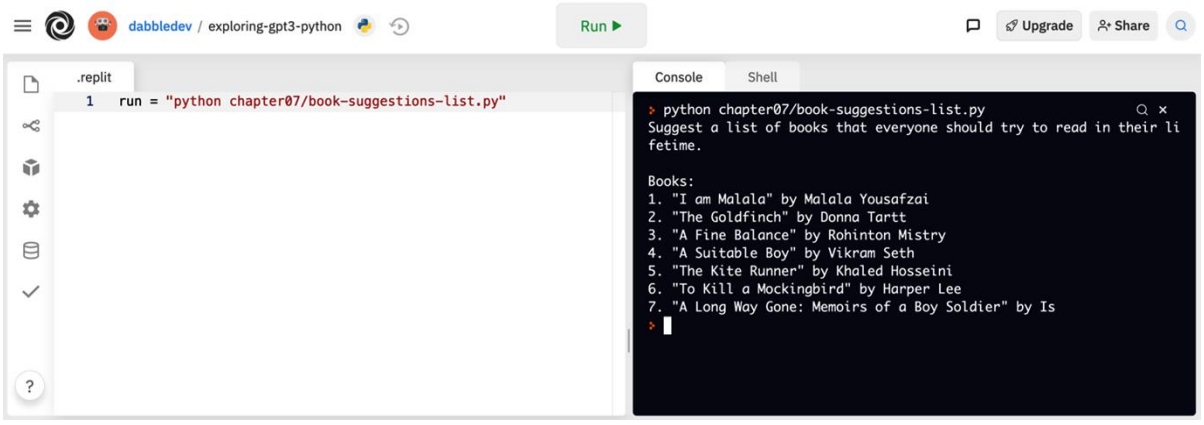

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/mars-facts-list.js"
Console
Shell
node chapter07/mars-facts-list.js
I'm studying the planets. List things I should know about Mars.
1. Mars is the nearest planet to Earth.
2. Mars has seasons, dry variety (not as damp as Earth's).
3. Mars' day is about the same length as Earth's (24.6 hours).
4. Mars has two moons, Phobos and Deimos.
5. Mars is the fourth planet from the Sun.
6. Mars is named after the Roman god of war.
7. Mars' surface is covered with craters and volcanoes.
8. Mars has a thin atmosphere made mostly of carbon dioxide (CO2).
9. The temperature on Mars can get as low as -100 degrees Fahrenheit (-73 degrees Celsius).
10. There are no oceans on Mars,

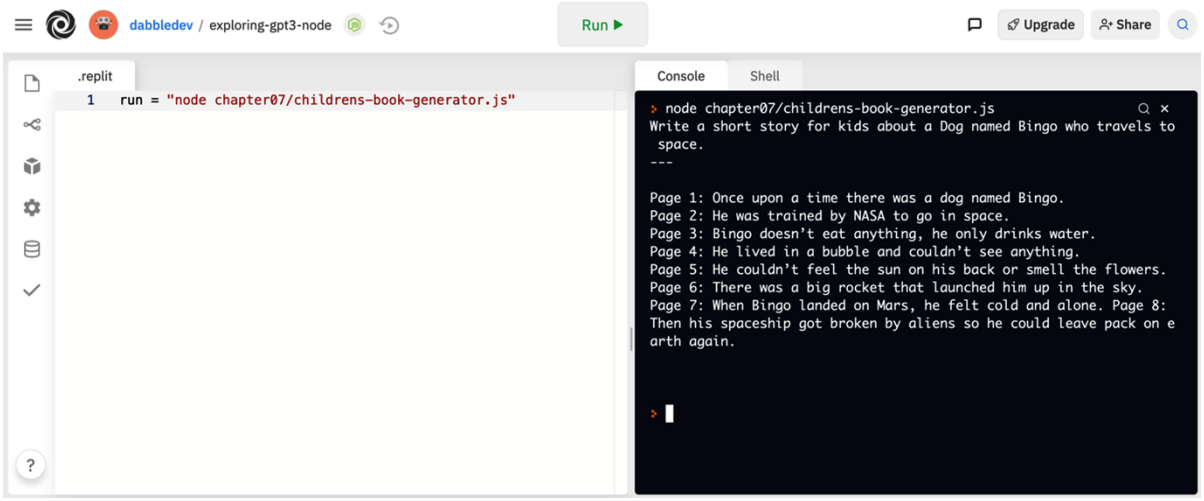

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/mars-facts-list.py"
Console
Shell
python chapter07/mars-facts-list.py
I'm studying the planets. List things I should know about Mars.
1. Mars is the nearest planet to Earth.
2. Mars has seasons, dry variety (not as damp as Earth's).
3. Mars' day is about the same length as Earth's (24.6 hours).
4. Mars has two moons, Phobos and Deimos.
5. Mars is the fourth planet from the Sun.
6. Mars is named after the Roman god of war.
7. Mars' surface is covered with craters and volcanoes.
8. Mars has a thin atmosphere made mostly of carbon dioxide (CO2).
9. The temperature on Mars can get as low as -100 degrees Fahrenheit (-73 degrees Celsius).
10. There are no oceans on Mars,


dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/webinar-description-generator.js"
Console
Shell
node chapter07/webinar-description-generator.js
Write a description for the following webinar:
Date: Monday, June 5, 2021
Time: 10 AM PT
Title: An introduction to mindfulness
Presenter: Gabi Calm
Event Description:
An introduction to mindfulness. Topics covered are: what is mindfulness, what is it not, how to develop it, and how to apply it in daily life. Mindfulness can be considered as a way of training the mind by paying attention on purpose without judgment or goal orientation. It is gentle, easy to learn but difficult to master

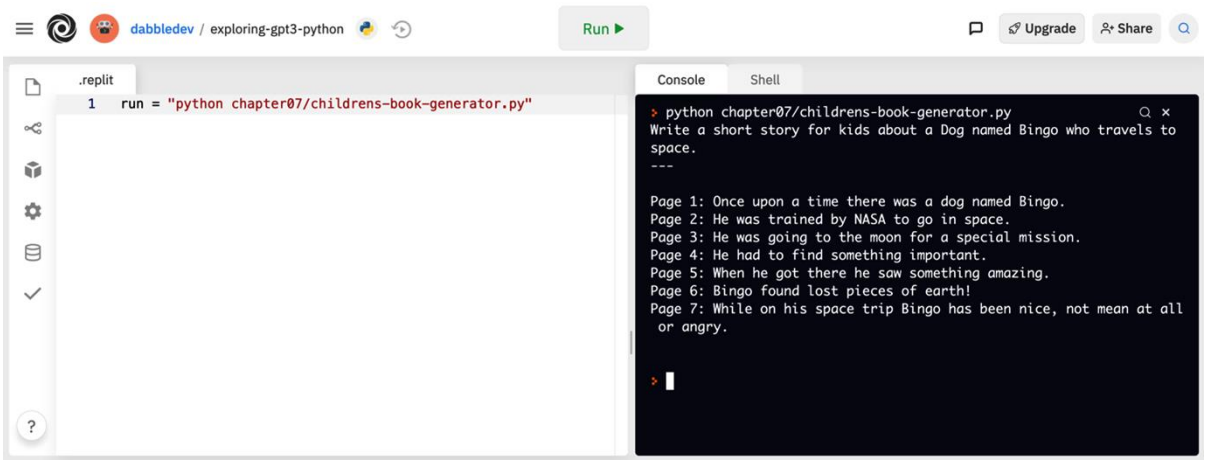

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/webinar-description-generator.py"
Console
Shell
python chapter07/webinar-description-generator.py
Write a description for the following webinar:
Date: Monday, June 5, 2021
Time: 10 AM PT
Title: An introduction to mindfulness
Presenter: Gabi Calm
Event Description: Mindfulness is a way to focus our attention on the present moment in a non-judgmental way. Learn the basic principles of mindfulness, and how to incorporate mindfulness into your daily life


dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/book-suggestions-list.js"
Console
Shell
node chapter07/book-suggestions-list.js
Suggest a list of books that everyone should try to read in their li
fetime.
Books:
1. Ayn Rand's "Atlas Shrugged" and "The Fountainhead" (these books a
re the reason I started thinking about politics)
2. Aristotle's "Nicomachean Ethics" (the most important book on ethi
cs ever written)
3. Steven Pinker's "The Blank Slate: The Modern Denial of Human Natu
re" (a must read for anyone who wants to understand how human nature
works)
4. Friedrich Hayek's "The Road to Serfdom"

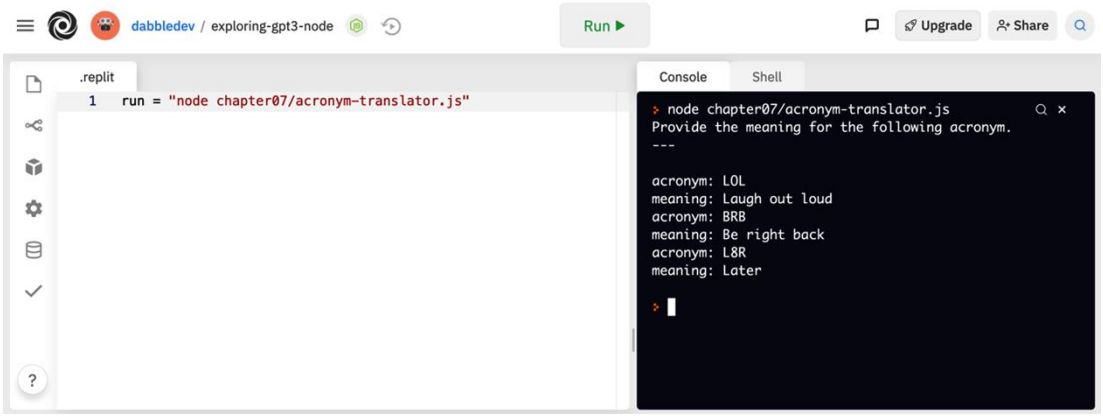

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/book-suggestions-list.py"
Console
Shell
python chapter07/book-suggestions-list.py
Suggest a list of books that everyone should try to read in their li
fetime.
Books:
1. "I am Malala" by Malala Yousafzai
2. "The Goldfinch" by Donna Tartt
3. "A Fine Balance" by Rohinton Mistry
4. "A Suitable Boy" by Vikram Seth
5. "The Kite Runner" by Khaled Hosseini
6. "To Kill a Mockingbird" by Harper Lee
7. "A Long Way Gone: Memoirs of a Boy Soldier" by Is

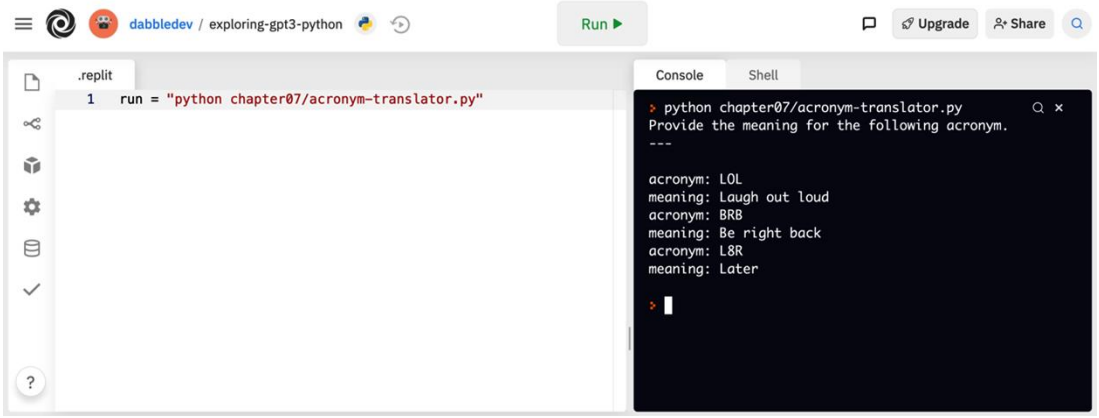

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/childrens-book-generator.js"
Console
Shell
node chapter07/childrens-book-generator.js
Write a short story for kids about a Dog named Bingo who travels to
space.

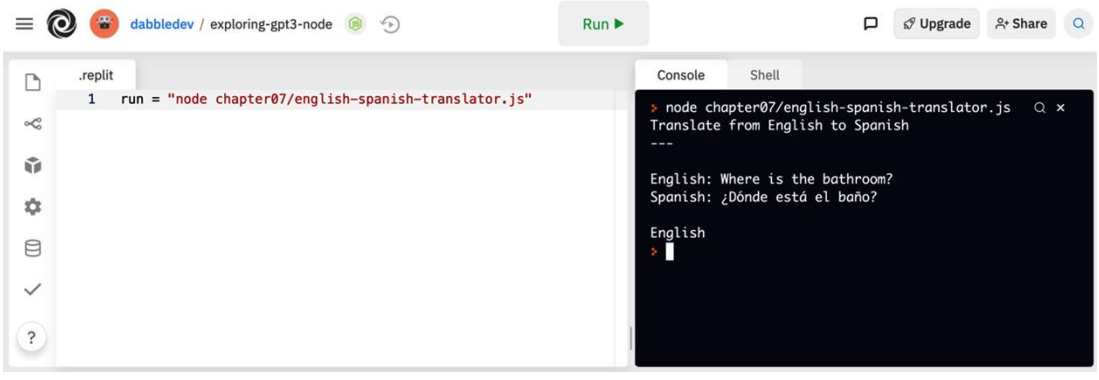
Page 1: Once upon a time there was a dog named Bingo.
Page 2: He was trained by NASA to go in space.
Page 3: Bingo doesn't eat anything, he only drinks water.
Page 4: He lived in a bubble and couldn't see anything.
Page 5: He couldn't feel the sun on his back or smell the flowers.
Page 6: There was a big rocket that launched him up in the sky.
Page 7: When Bingo landed on Mars, he felt cold and alone. Page 8:
Then his spaceship got broken by aliens so he could leave pack on e
arth again.


dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/childrens-book-generator.py"
Console
Shell
python chapter07/childrens-book-generator.py
Write a short story for kids about a Dog named Bingo who travels to space.

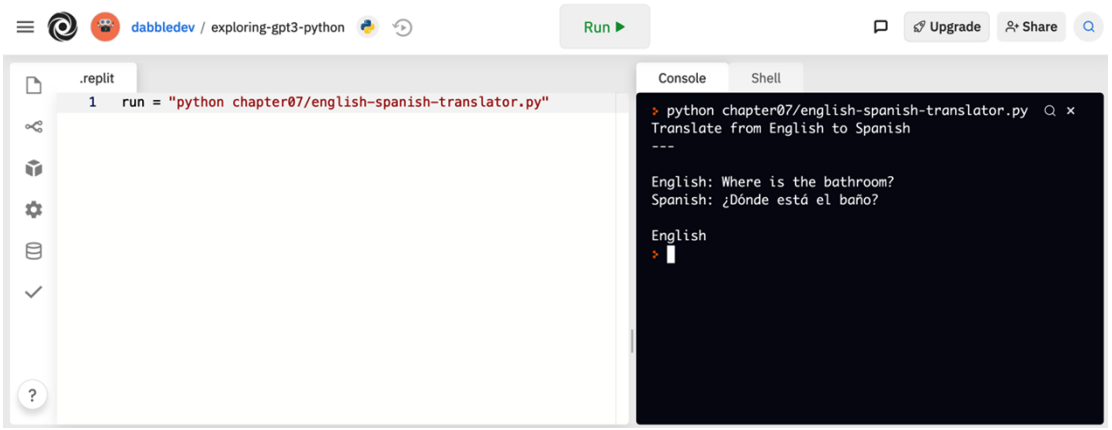
Page 1: Once upon a time there was a dog named Bingo.
Page 2: He was trained by NASA to go in space.
Page 3: He was going to the moon for a special mission.
Page 4: He had to find something important.
Page 5: When he got there he saw something amazing.
Page 6: Bingo found lost pieces of earth!
Page 7: While on his space trip Bingo has been nice, not mean at all or angry.


dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/acronym-translator.js"
Console
Shell
node chapter07/acronym-translator.js
Provide the meaning for the following acronym.
---
acronym: LOL
meaning: Laugh out loud
acronym: BRB
meaning: Be right back
acronym: L8R
meaning: Later

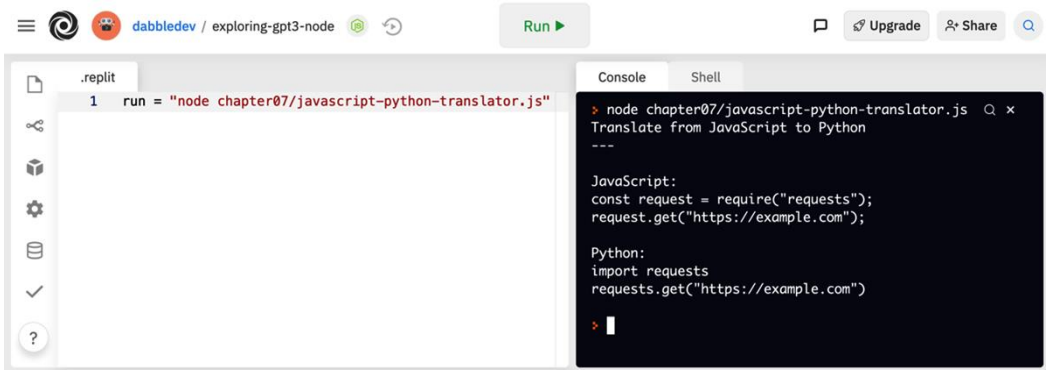

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/acronym-translator.py"
Console
Shell
python chapter07/acronym-translator.py
Provide the meaning for the following acronym.
---
acronym: LOL
meaning: Laugh out loud
acronym: BRB
meaning: Be right back
acronym: L8R
meaning: Later


dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/english-spanish-translator.js"
Console
Shell
node chapter07/english-spanish-translator.js
Translate from English to Spanish

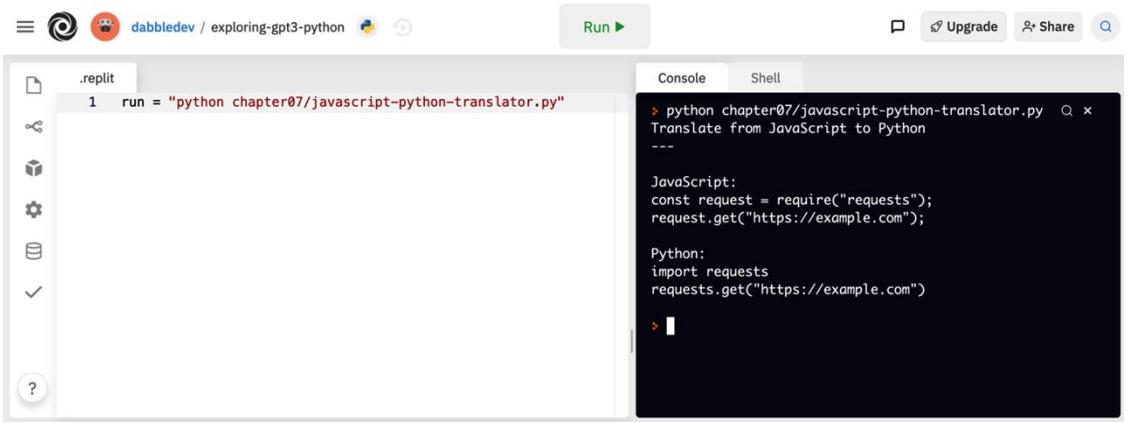
English: Where is the bathroom?
Spanish: ¿Dónde está el baño?
English


dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/english-spanish-translator.py"
Console
Shell
python chapter07/english-spanish-translator.py
Translate from English to Spanish

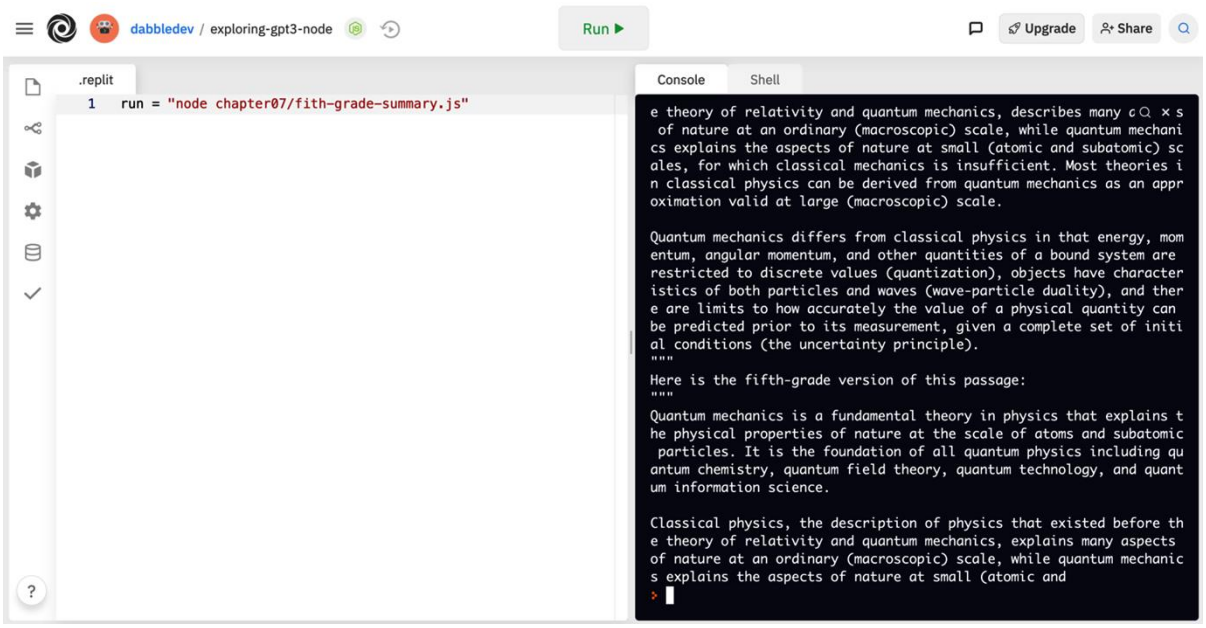
English: Where is the bathroom?
Spanish: ¿Dónde está el baño?
English

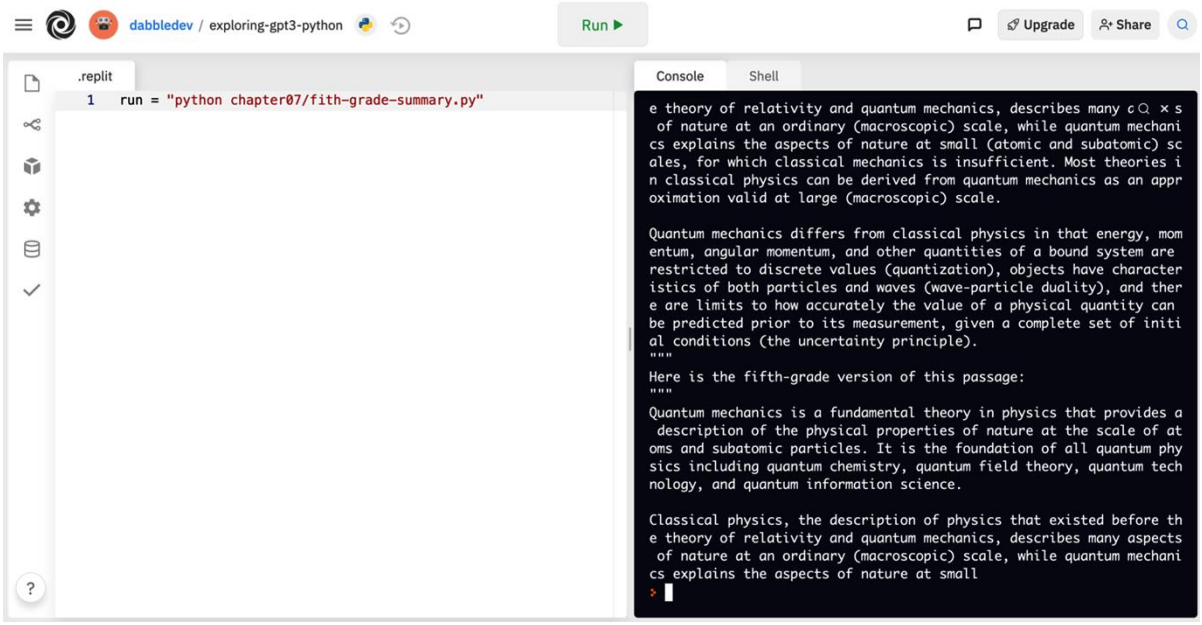

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/javascript-python-translator.js"
Console
Shell
node chapter07/javascript-python-translator.js
Translate from JavaScript to Python

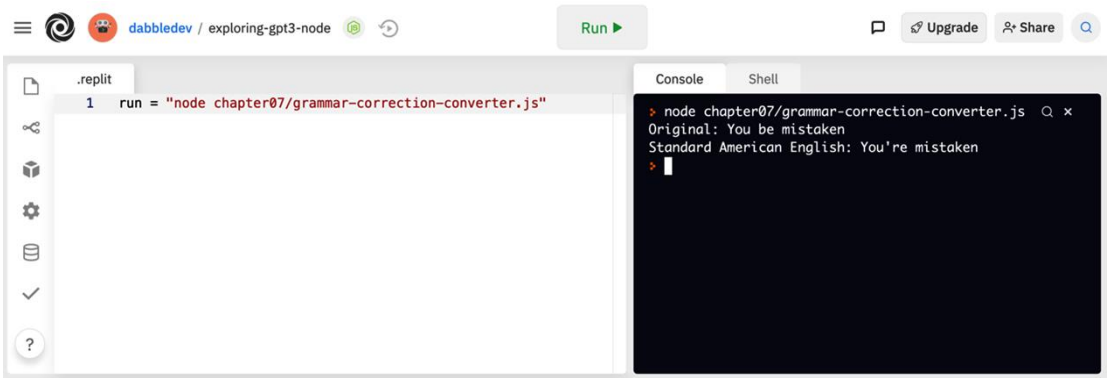
JavaScript:
const request = require("requests");
request.get("https://example.com");
Python:
import requests
requests.get("https://example.com")

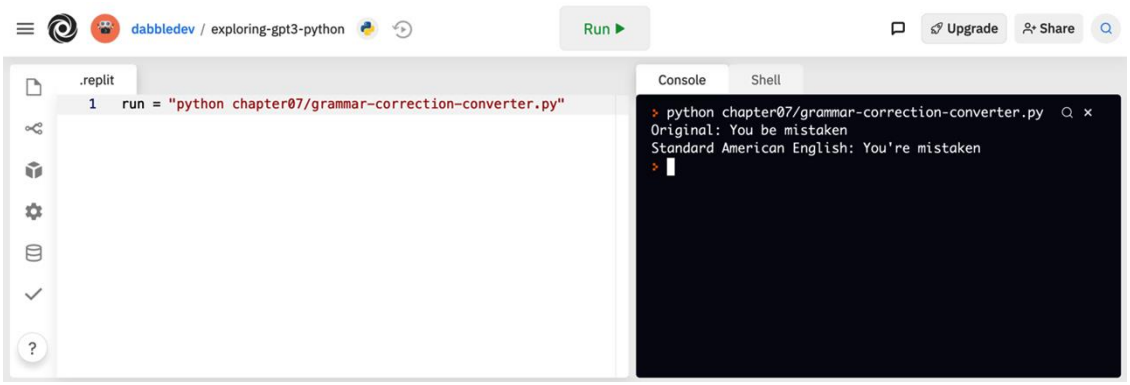

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/javascript-python-translator.py"
Console
Shell
python chapter07/javascript-python-translator.py
Translate from JavaScript to Python

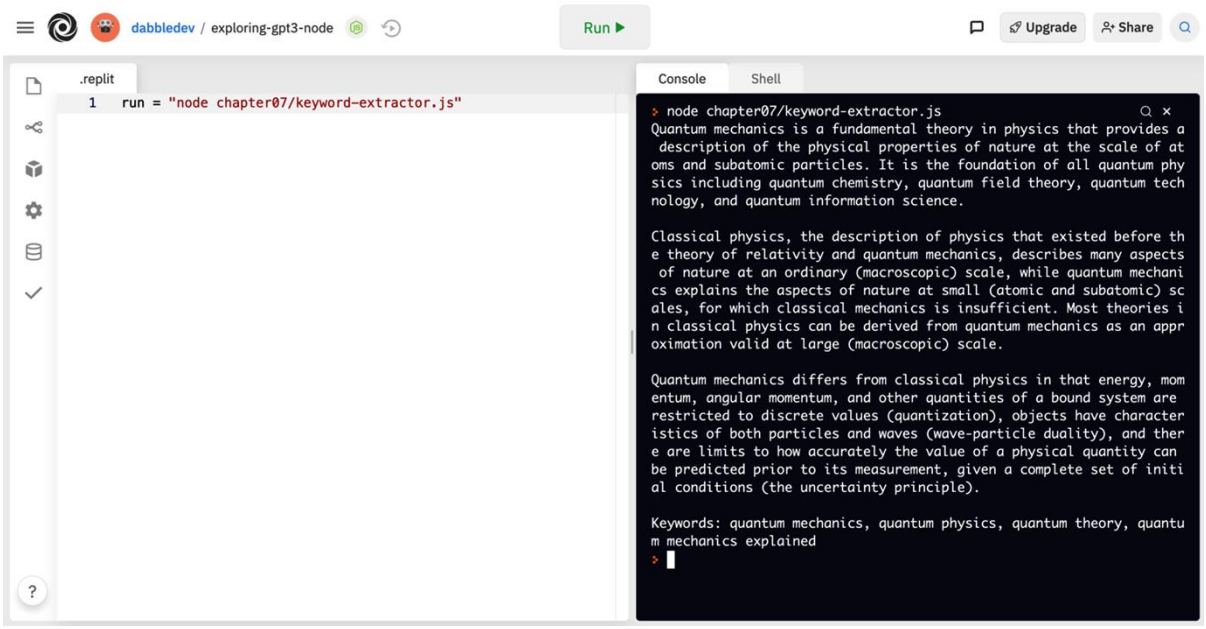
JavaScript:
const request = require("requests");
request.get("https://example.com");
Python:
import requests
requests.get("https://example.com")

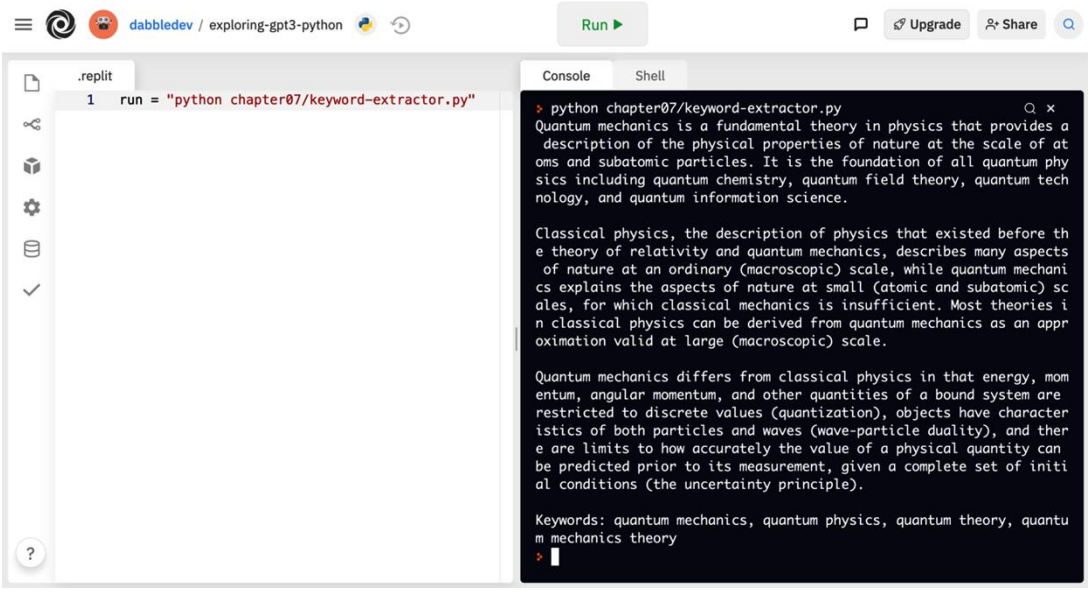

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/fith-grade-summary.js"
Console
Shell
e theory of relativity and quantum mechanics, describes many c
of nature at an ordinary (macroscopic) scale, while quantum mechani
cs explains the aspects of nature at small (atomic and subatomic) sc
ales, for which classical mechanics is insufficient. Most theories i
n classical physics can be derived from quantum mechanics as an appr
oximation valid at large (macroscopic) scale.
Quantum mechanics differs from classical physics in that energy, mom
entum, angular momentum, and other quantities of a bound system are
restricted to discrete values (quantization), objects have character
istics of both particles and waves (wave-particle duality), and ther
e are limits to how accurately the value of a physical quantity can
be predicted prior to its measurement, given a complete set of initi
al conditions (the uncertainty principle).
"""
Here is the fifth-grade version of this passage:
"""
Quantum mechanics is a fundamental theory in physics that explains t
he physical properties of nature at the scale of atoms and subatomic
particles. It is the foundation of all quantum physics including qu
antum chemistry, quantum field theory, quantum technology, and quant
um information science.
Classical physics, the description of physics that existed before th
e theory of relativity and quantum mechanics, explains many aspects
of nature at an ordinary (macroscopic) scale, while quantum mechanic
s explains the aspects of nature at small (atomic and

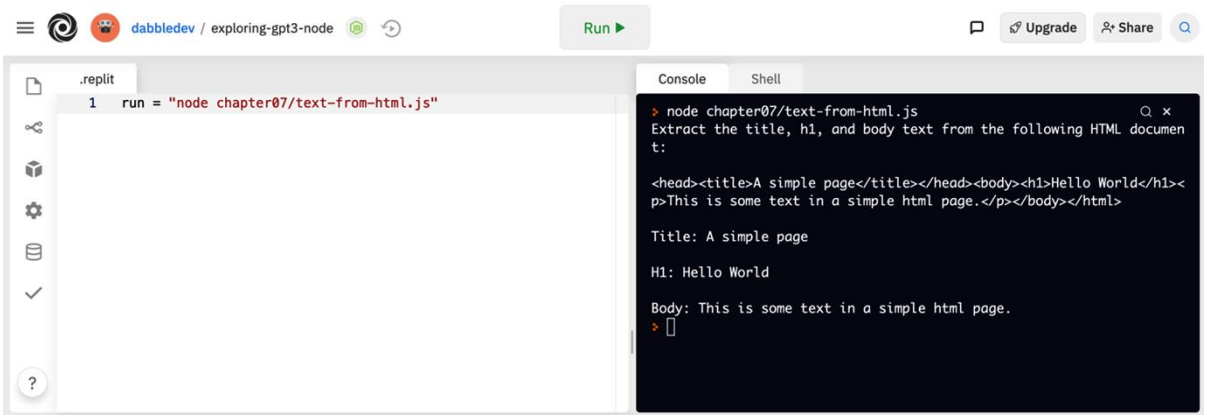

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/fith-grade-summary.py"
Console
Shell
e theory of relativity and quantum mechanics, describes many c
of nature at an ordinary (macroscopic) scale, while quantum mechani
cs explains the aspects of nature at small (atomic and subatomic) sc
ales, for which classical mechanics is insufficient. Most theories i
n classical physics can be derived from quantum mechanics as an appr
oximation valid at large (macroscopic) scale.
Quantum mechanics differs from classical physics in that energy, mom
entum, angular momentum, and other quantities of a bound system are
restricted to discrete values (quantization), objects have character
istics of both particles and waves (wave-particle duality), and ther
e are limits to how accurately the value of a physical quantity can
be predicted prior to its measurement, given a complete set of initi
al conditions (the uncertainty principle).
"""
Here is the fifth-grade version of this passage:
"""
Quantum mechanics is a fundamental theory in physics that provides a
description of the physical properties of nature at the scale of at
oms and subatomic particles. It is the foundation of all quantum phy
sics including quantum chemistry, quantum field theory, quantum tech
nology, and quantum information science.
Classical physics, the description of physics that existed before th
e theory of relativity and quantum mechanics, describes many aspects
of nature at an ordinary (macroscopic) scale, while quantum mechani
cs explains the aspects of nature at small

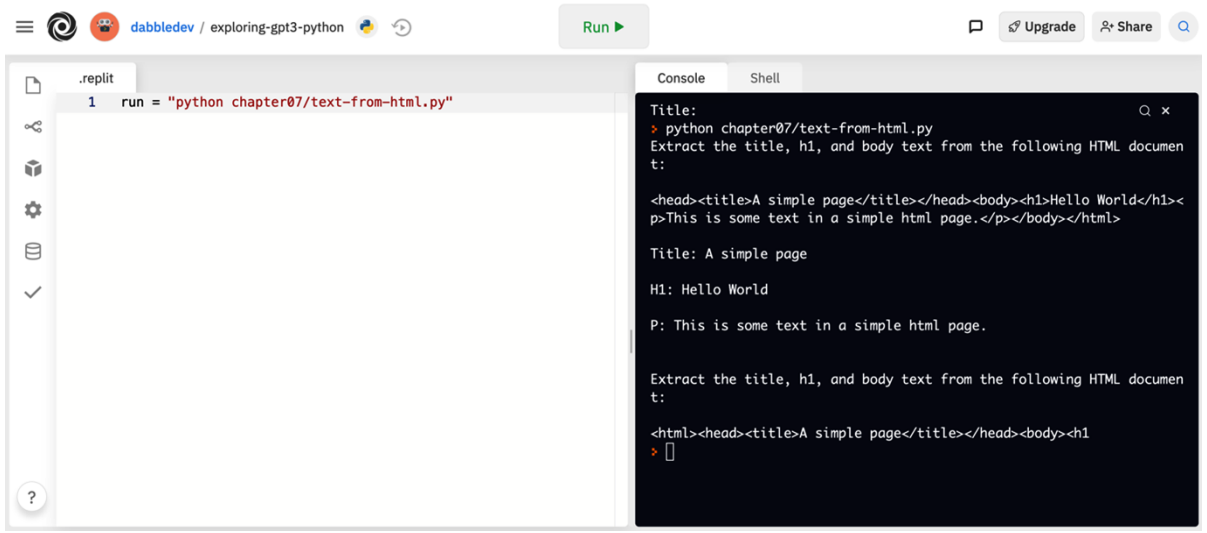

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/grammar-correction-converter.js"
Console
Shell
node chapter07/grammar-correction-converter.js
Original: You be mistaken
Standard American English: You're mistaken

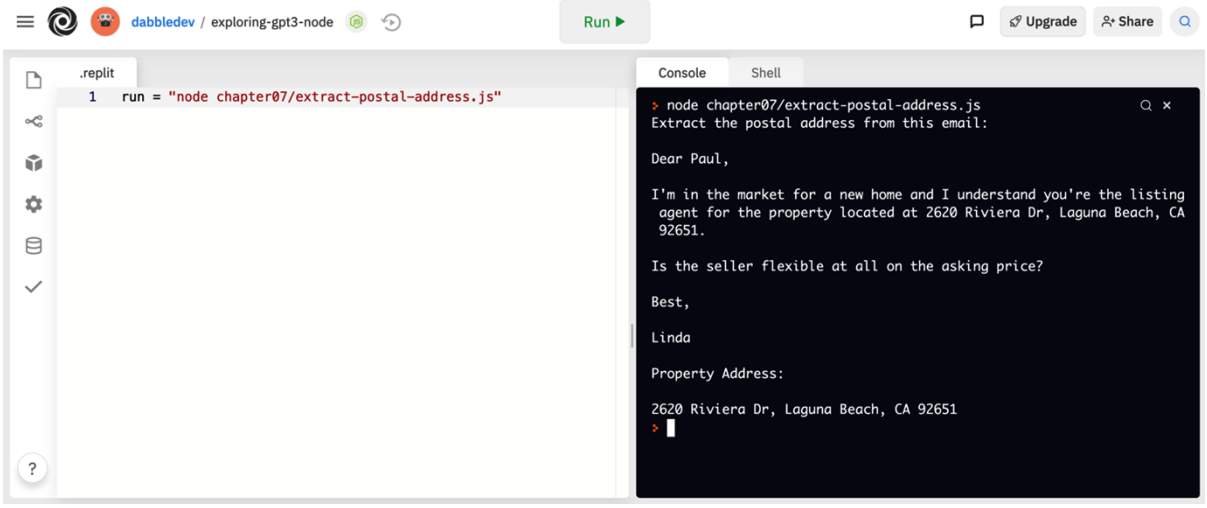

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/grammar-correction-converter.py"
Console
Shell
python chapter07/grammar-correction-converter.py
Original: You be mistaken
Standard American English: You're mistaken

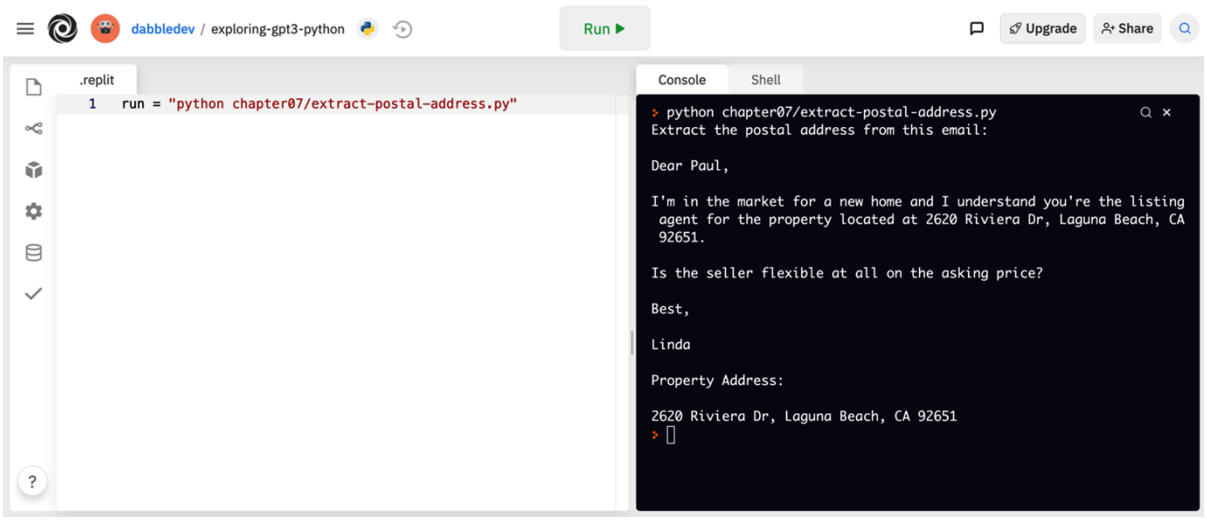

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/keyword-extractor.js"
Console
Shell
node chapter07/keyword-extractor.js
Quantum mechanics is a fundamental theory in physics that provides a description of the physical properties of nature at the scale of atoms and subatomic particles. It is the foundation of all quantum physics including quantum chemistry, quantum field theory, quantum technology, and quantum information science.
Classical physics, the description of physics that existed before the theory of relativity and quantum mechanics, describes many aspects of nature at an ordinary (macroscopic) scale, while quantum mechanics explains the aspects of nature at small (atomic and subatomic) scales, for which classical mechanics is insufficient. Most theories in classical physics can be derived from quantum mechanics as an approximation valid at large (macroscopic) scale.
Quantum mechanics differs from classical physics in that energy, momentum, angular momentum, and other quantities of a bound system are restricted to discrete values (quantization), objects have characteristics of both particles and waves (wave-particle duality), and there are limits to how accurately the value of a physical quantity can be predicted prior to its measurement, given a complete set of initial conditions (the uncertainty principle).
Keywords: quantum mechanics, quantum physics, quantum theory, quantum mechanics explained

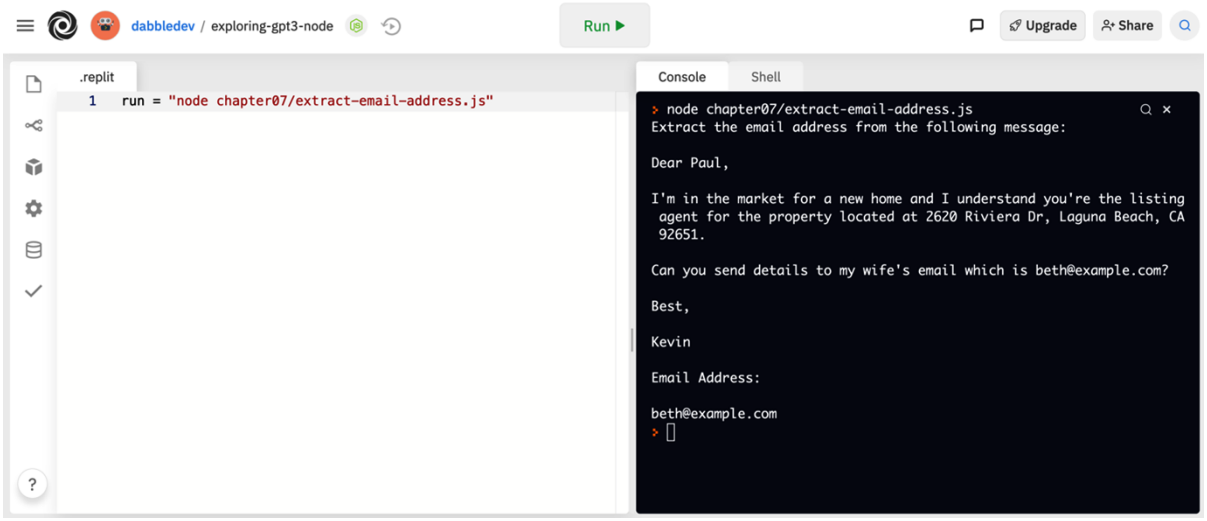

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/keyword-extractor.py"
Console
Shell
python chapter07/keyword-extractor.py
Quantum mechanics is a fundamental theory in physics that provides a description of the physical properties of nature at the scale of atoms and subatomic particles. It is the foundation of all quantum physics including quantum chemistry, quantum field theory, quantum technology, and quantum information science.
Classical physics, the description of physics that existed before the theory of relativity and quantum mechanics, describes many aspects of nature at an ordinary (macroscopic) scale, while quantum mechanics explains the aspects of nature at small (atomic and subatomic) scales, for which classical mechanics is insufficient. Most theories in classical physics can be derived from quantum mechanics as an approximation valid at large (macroscopic) scale.
Quantum mechanics differs from classical physics in that energy, momentum, angular momentum, and other quantities of a bound system are restricted to discrete values (quantization), objects have characteristics of both particles and waves (wave-particle duality), and there are limits to how accurately the value of a physical quantity can be predicted prior to its measurement, given a complete set of initial conditions (the uncertainty principle).
Keywords: quantum mechanics, quantum physics, quantum theory, quantum mechanics theory

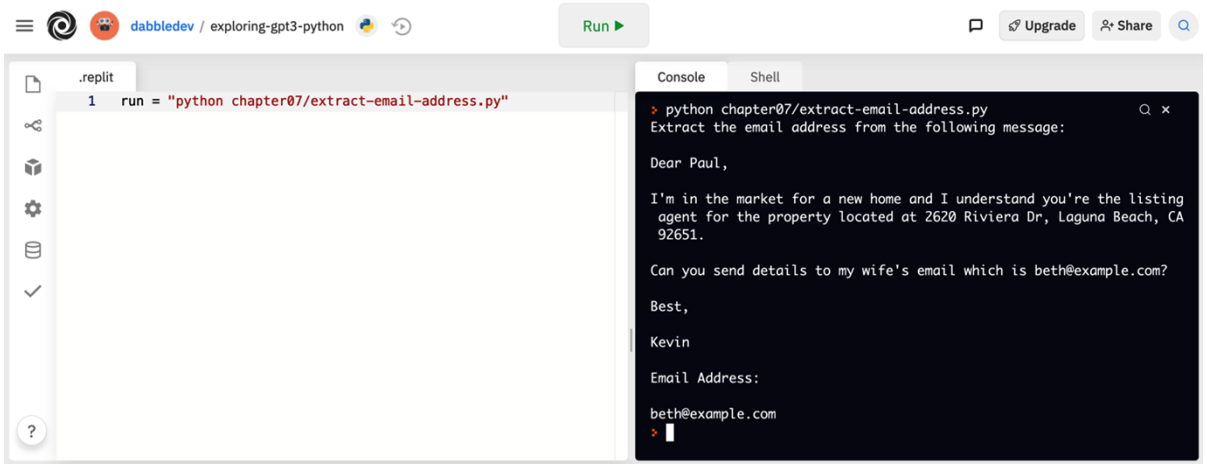

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/text-from-html.js"
Console
Shell
node chapter07/text-from-html.js
Extract the title, h1, and body text from the following HTML documen
t:
<head><title>A simple page</title></head><body><h1>Hello World</h1><
p>This is some text in a simple html page.</p></body></html>
Title: A simple page
H1: Hello World
Body: This is some text in a simple html page.

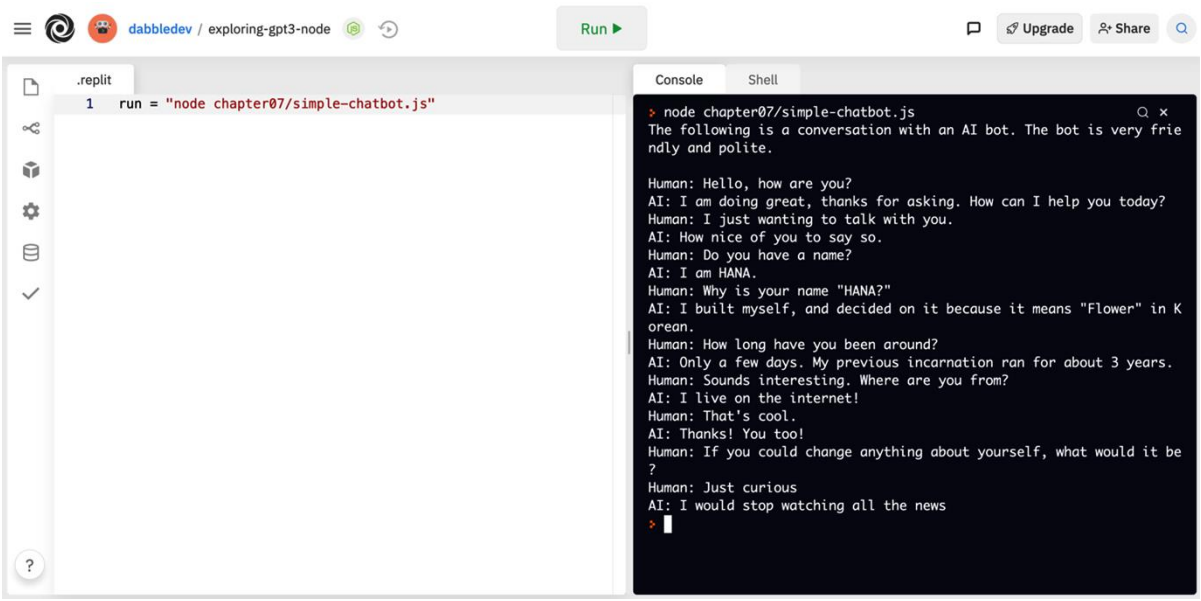

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/text-from-html.py"
Console
Shell
Title:
python chapter07/text-from-html.py
Extract the title, h1, and body text from the following HTML documen
t:
<head><title>A simple page</title></head><body><h1>Hello World</h1><
p>This is some text in a simple html page.</p></body></html>
Title: A simple page
H1: Hello World
P: This is some text in a simple html page.
Extract the title, h1, and body text from the following HTML documen
t:
<html><head><title>A simple page</title></head><body><h1

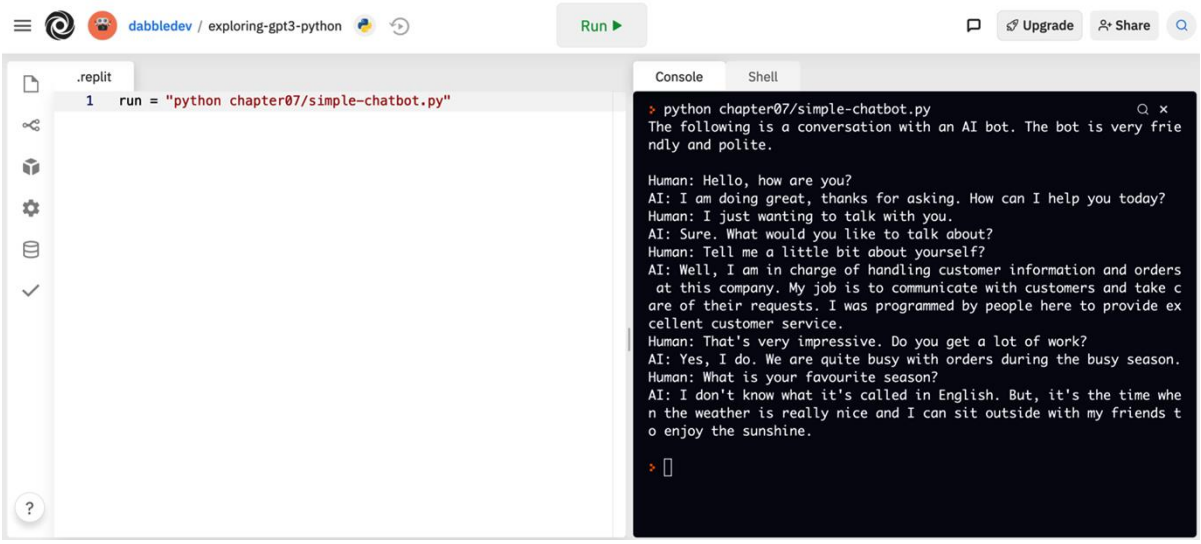

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/extract-postal-address.js"
Console
Shell
node chapter07/extract-postal-address.js
Extract the postal address from this email:
Dear Paul,
I'm in the market for a new home and I understand you're the listing
agent for the property located at 2620 Riviera Dr, Laguna Beach, CA
92651.
Is the seller flexible at all on the asking price?
Best,
Linda
Property Address:
2620 Riviera Dr, Laguna Beach, CA 92651


dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/extract-postal-address.py"
Console
Shell
python chapter07/extract-postal-address.py
Extract the postal address from this email:
Dear Paul,
I'm in the market for a new home and I understand you're the listing agent for the property located at 2620 Riviera Dr, Laguna Beach, CA 92651.
Is the seller flexible at all on the asking price?
Best,
Linda
Property Address:
2620 Riviera Dr, Laguna Beach, CA 92651

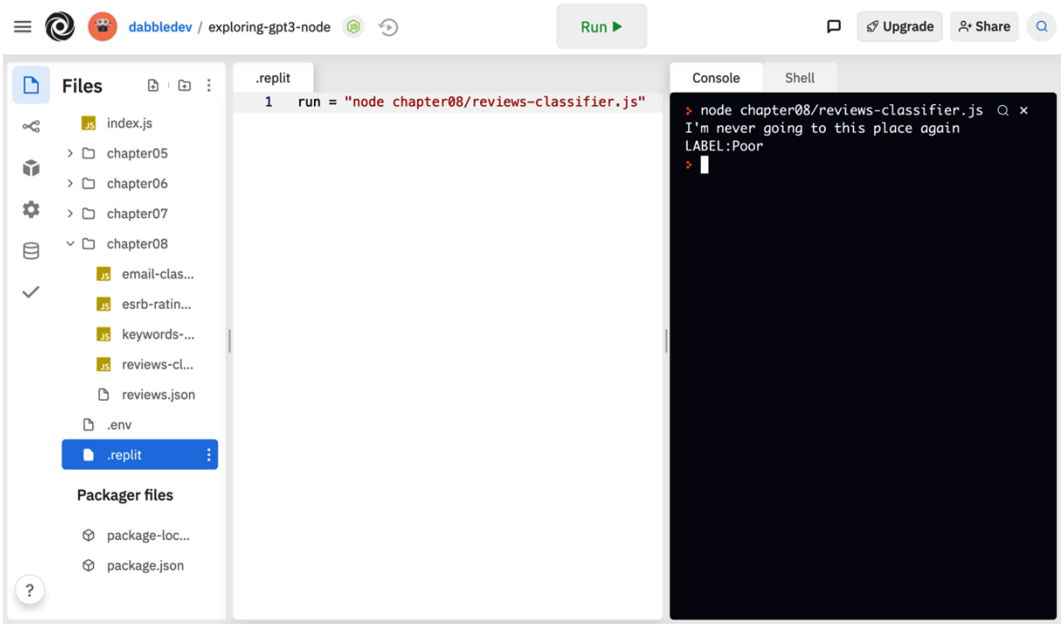

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/extract-email-address.js"
Console
Shell
node chapter07/extract-email-address.js
Extract the email address from the following message:
Dear Paul,
I'm in the market for a new home and I understand you're the listing agent for the property located at 2620 Riviera Dr, Laguna Beach, CA 92651.
Can you send details to my wife's email which is beth@example.com?
Best,
Kevin
Email Address:
beth@example.com

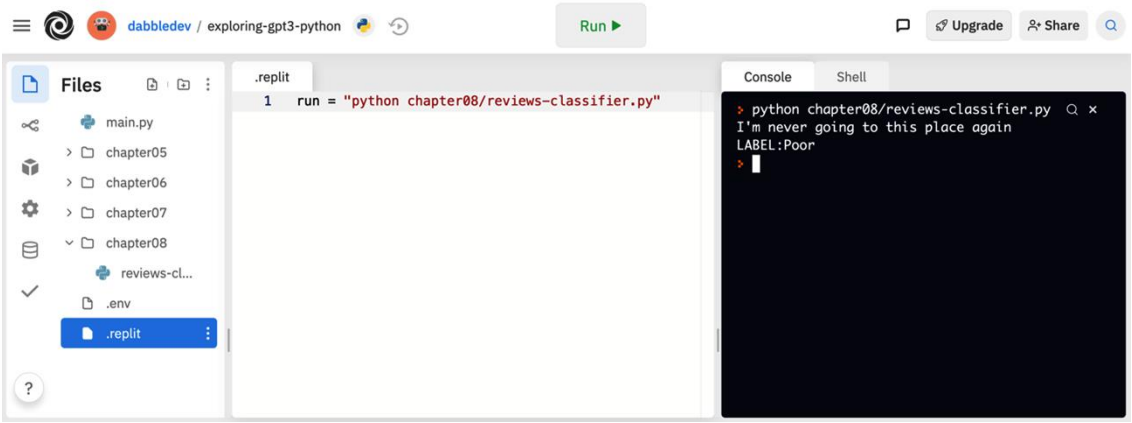

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/extract-email-address.py"
Console
Shell
python chapter07/extract-email-address.py
Extract the email address from the following message:
Dear Paul,
I'm in the market for a new home and I understand you're the listing agent for the property located at 2620 Riviera Dr, Laguna Beach, CA 92651.
Can you send details to my wife's email which is beth@example.com?
Best,
Kevin
Email Address:
beth@example.com

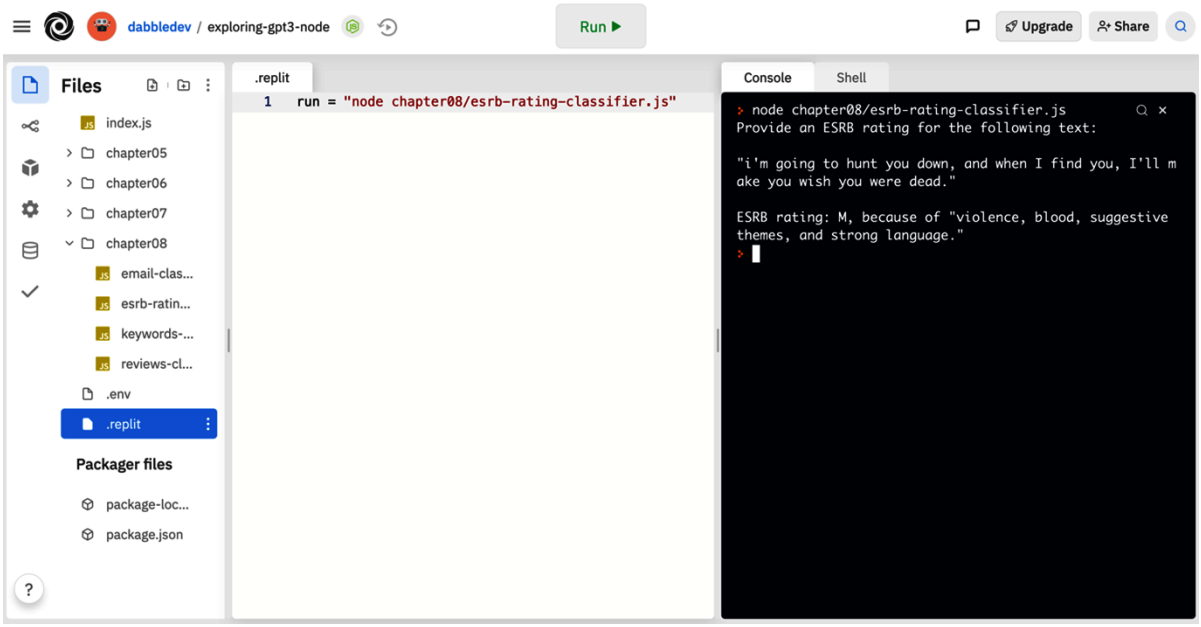

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
.replit
1 run = "node chapter07/simple-chatbot.js"
Console
Shell
node chapter07/simple-chatbot.js
The following is a conversation with an AI bot. The bot is very friendly and polite.
Human: Hello, how are you?
AI: I am doing great, thanks for asking. How can I help you today?
Human: I just wanting to talk with you.
AI: How nice of you to say so.
Human: Do you have a name?
AI: I am HANA.
Human: Why is your name "HANA?"
AI: I built myself, and decided on it because it means "Flower" in Korean.
Human: How long have you been around?
AI: Only a few days. My previous incarnation ran for about 3 years.
Human: Sounds interesting. Where are you from?
AI: I live on the internet!
Human: That's cool.
AI: Thanks! You too!
Human: If you could change anything about yourself, what would it be?
Human: Just curious
AI: I would stop watching all the news

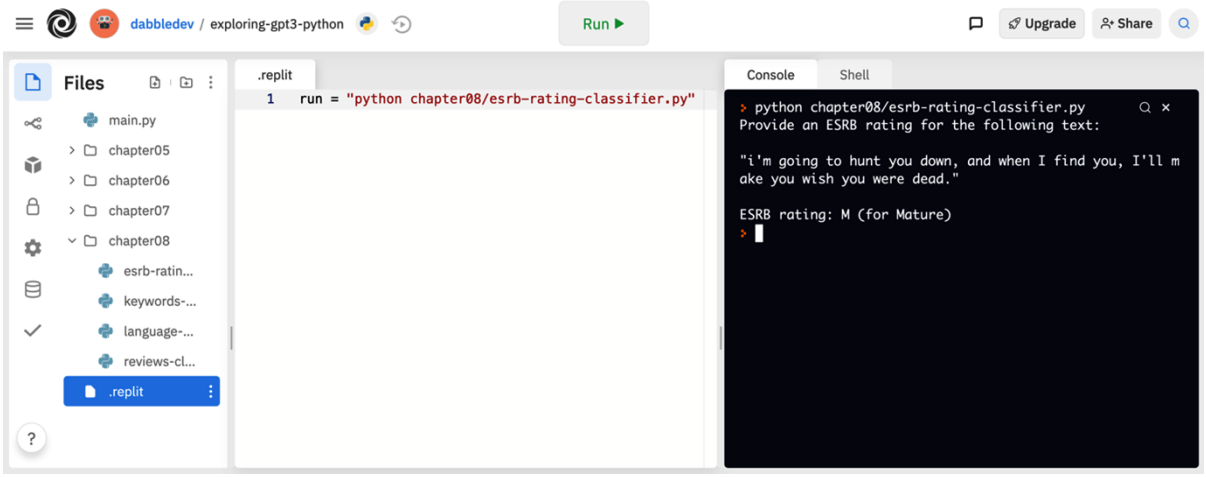

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
.replit
1 run = "python chapter07/simple-chatbot.py"
Console
Shell
python chapter07/simple-chatbot.py
The following is a conversation with an AI bot. The bot is very friendly and polite.
Human: Hello, how are you?
AI: I am doing great, thanks for asking. How can I help you today?
Human: I just wanting to talk with you.
AI: Sure. What would you like to talk about?
Human: Tell me a little bit about yourself?
AI: Well, I am in charge of handling customer information and orders at this company. My job is to communicate with customers and take care of their requests. I was programmed by people here to provide excellent customer service.
Human: That's very impressive. Do you get a lot of work?
AI: Yes, I do. We are quite busy with orders during the busy season.
Human: What is your favourite season?
AI: I don't know what it's called in English. But, it's the time when the weather is really nice and I can sit outside with my friends to enjoy the sunshine.

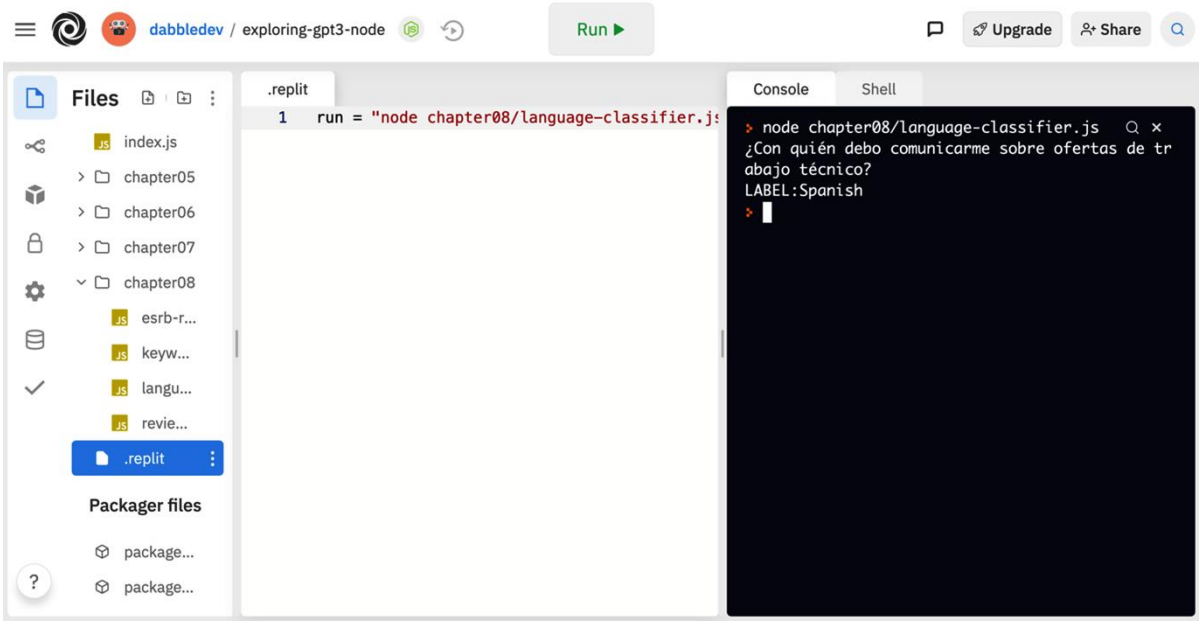
Chapter 8: Classifying and Categorizing Text

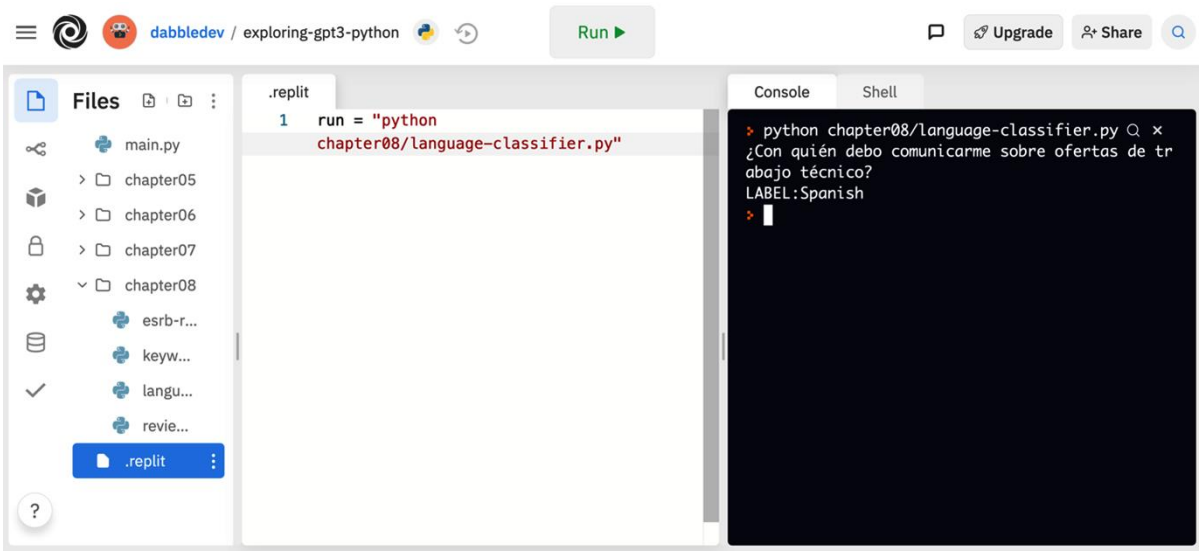


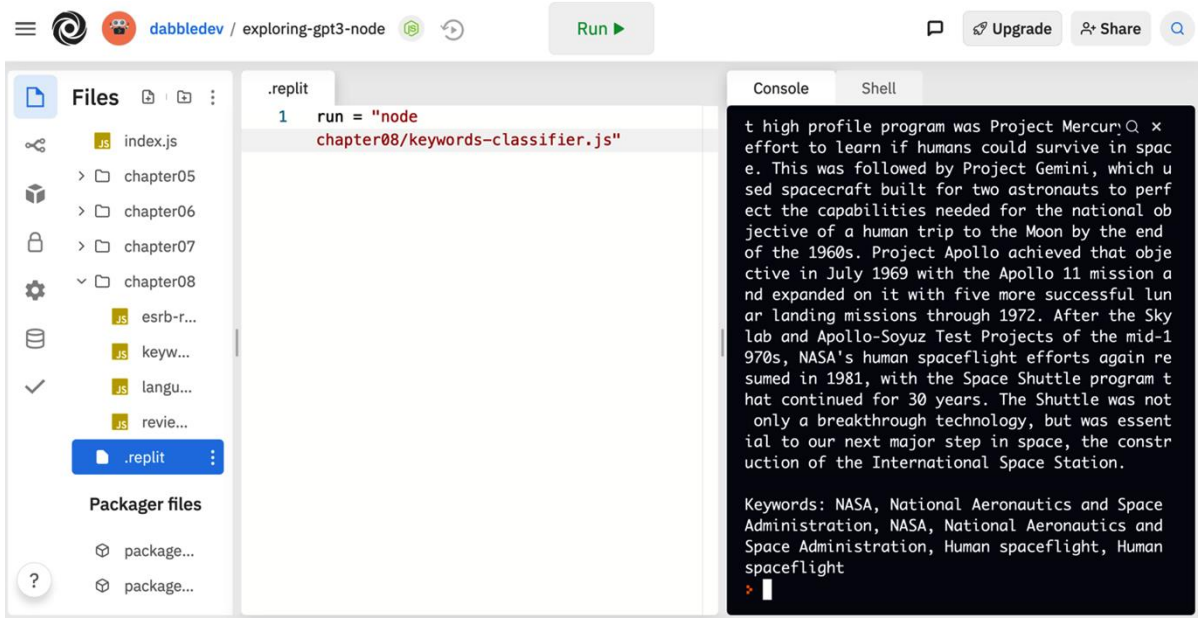


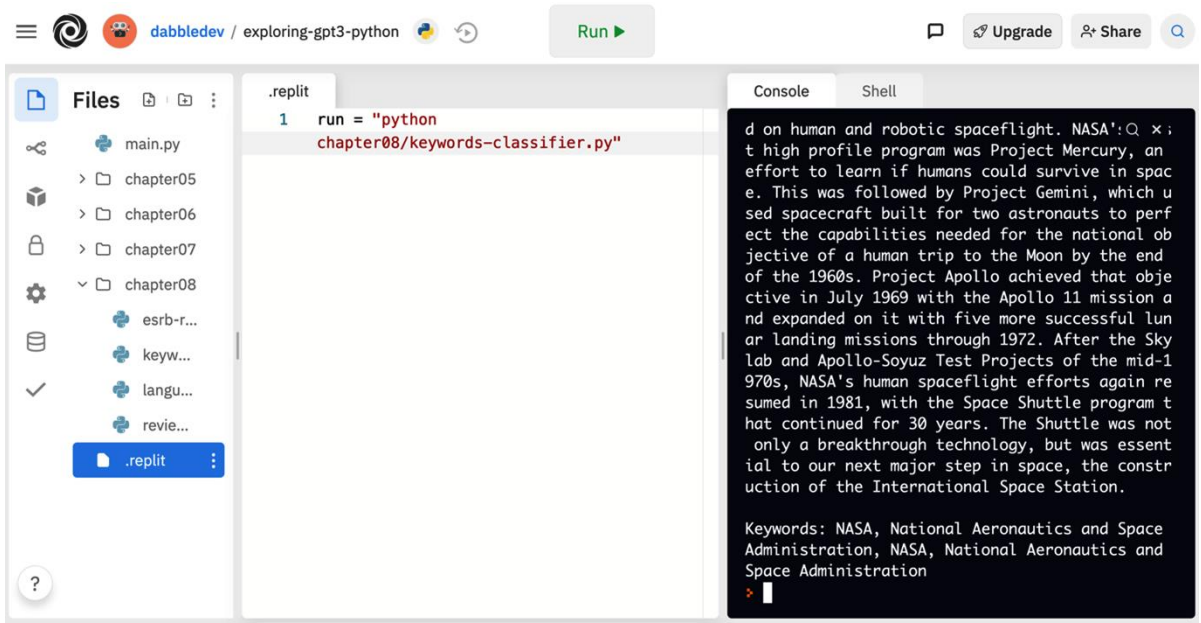

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
Files
index.js
chapter05
chapter06
chapter07
chapter08
email-clas...
esrb-ratin...
keywords-...
reviews-cl...
.env
.replit
Packager files
package-loc...
package.json
.replit
1 run = "node chapter08/esrb-rating-classifier.js"
Console
Shell
node chapter08/esrb-rating-classifier.js
Provide an ESRB rating for the following text:
"i'm going to hunt you down, and when I find you, I'll m
ake you wish you were dead."
ESRB rating: M, because of "violence, blood, suggestive
themes, and strong language."


dabbledev / exploring-gpt3-python
Run
Upgrade
Share
Files
main.py
chapter05
chapter06
chapter07
chapter08
esrb-ratin...
keywords-...
language-...
reviews-cl...
.replit
.replit
1 run = "python chapter08/esrb-rating-classifier.py"
Console
Shell
python chapter08/esrb-rating-classifier.py
Provide an ESRB rating for the following text:
"i'm going to hunt you down, and when I find you, I'll m
ake you wish you were dead."
ESRB rating: M (for Mature)

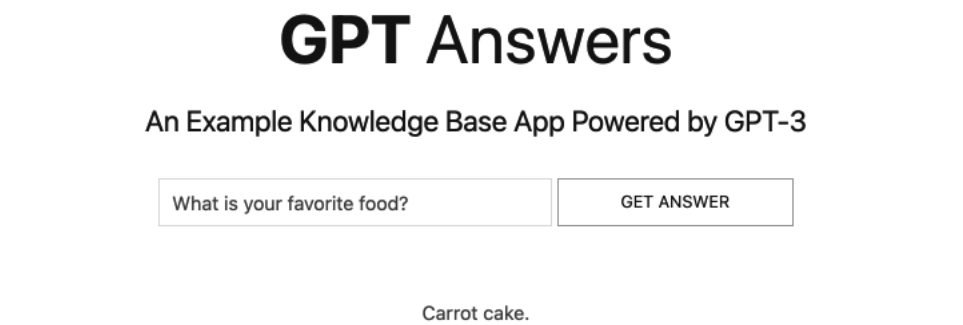

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
Files
index.js
chapter05
chapter06
chapter07
chapter08
esrb-r...
keyw...
langu...
revie...
.replit
Packager files
package...
package...
.replit
1 run = "node chapter08/language-classifier.js
Console
Shell
node chapter08/language-classifier.js
¿Con quién debo comunicarme sobre ofertas de tr
abajo técnico?
LABEL:Spanish

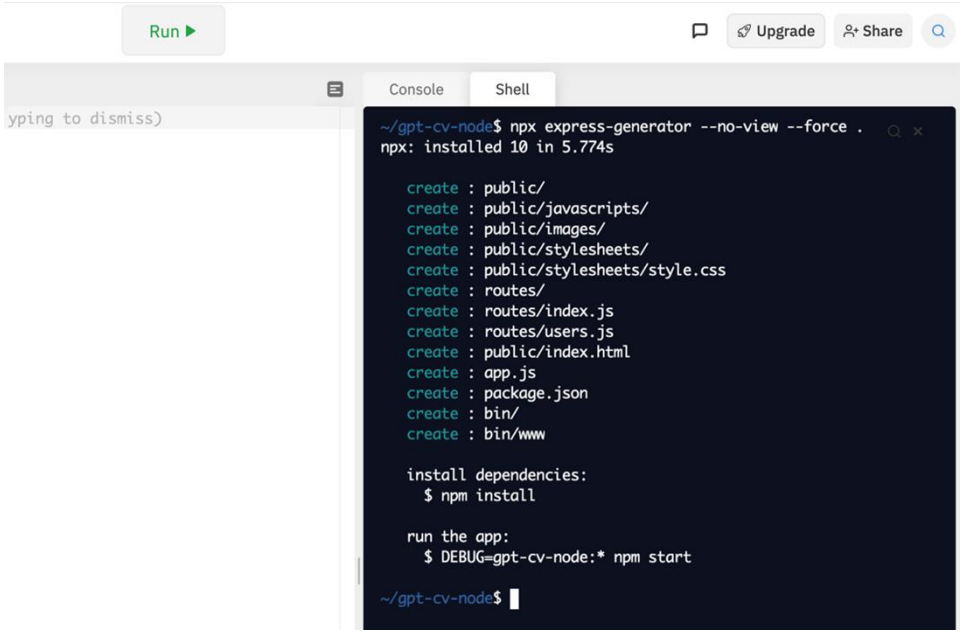

dabbledev / exploring-gpt3-python
Run
Upgrade
Share
Files
main.py
chapter05
chapter06
chapter07
chapter08
esrb-r...
keyw...
langu...
revie...
.replit
.replit
1 run = "python
chapter08/language-classifier.py"
Console
Shell
python chapter08/language-classifier.py
¿Con quién debo comunicarme sobre ofertas de tr
abajo técnico?
LABEL:Spanish

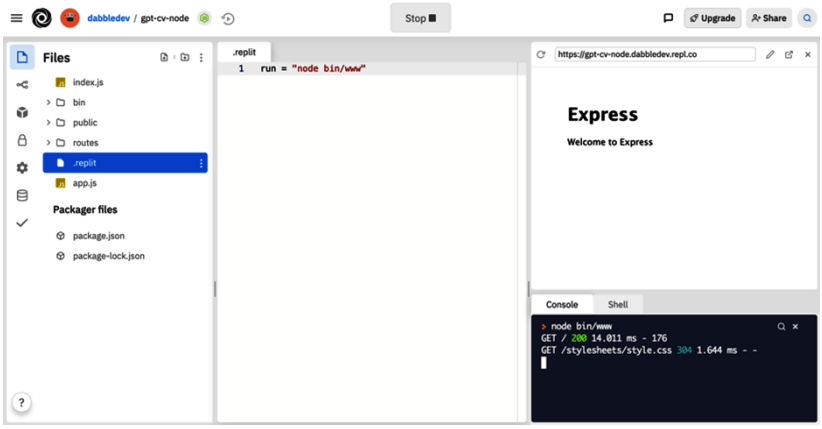

dabbledev / exploring-gpt3-node
Run
Upgrade
Share
Files
index.js
chapter05
chapter06
chapter07
chapter08
esrb-r...
keyw...
langu...
revie...
.replit
Packager files
package...
package...
.replit
1 run = "node
chapter08/keywords-classifier.js"
Console
Shell
t high profile program was Project Mercury, an
effort to learn if humans could survive in spac
e. This was followed by Project Gemini, which u
sed spacecraft built for two astronauts to perf
ect the capabilities needed for the national ob
jective of a human trip to the Moon by the end
of the 1960s. Project Apollo achieved that obje
ctive in July 1969 with the Apollo 11 mission a
nd expanded on it with five more successful lun
ar landing missions through 1972. After the Sky
lab and Apollo-Soyuz Test Projects of the mid-1
970s, NASA's human spaceflight efforts again re
sumed in 1981, with the Space Shuttle program t
hat continued for 30 years. The Shuttle was not
only a breakthrough technology, but was essent
ial to our next major step in space, the constr
uction of the International Space Station.
Keywords: NASA, National Aeronautics and Space
Administration, NASA, National Aeronautics and
Space Administration, Human spaceflight, Human
spaceflight


dabbledev / exploring-gpt3-python
Run
Upgrade
Share
Files
main.py
chapter05
chapter06
chapter07
chapter08
esrb-r...
keyw...
langu...
revie...
.replit
.replit
1 run = "python
chapter08/keywords-classifier.py"
Console
Shell
d on human and robotic spaceflight. NASA's
t high profile program was Project Mercury, an
effort to learn if humans could survive in spac
e. This was followed by Project Gemini, which u
sed spacecraft built for two astronauts to perf
ect the capabilities needed for the national ob
jective of a human trip to the Moon by the end
of the 1960s. Project Apollo achieved that obje
ctive in July 1969 with the Apollo 11 mission a
nd expanded on it with five more successful lun
ar landing missions through 1972. After the Sky
lab and Apollo-Soyuz Test Projects of the mid-1
970s, NASA's human spaceflight efforts again re
sumed in 1981, with the Space Shuttle program t
hat continued for 30 years. The Shuttle was not
only a breakthrough technology, but was essent
ial to our next major step in space, the constr
uction of the International Space Station.
Keywords: NASA, National Aeronautics and Space
Administration, NASA, National Aeronautics and
Space Administration

Chapter 9: Building a GPT-3-Powered Question-Answering App





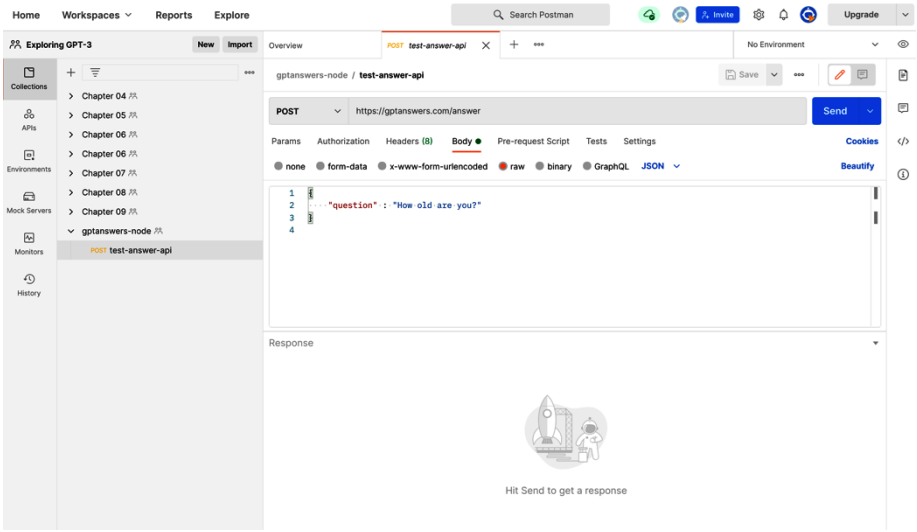


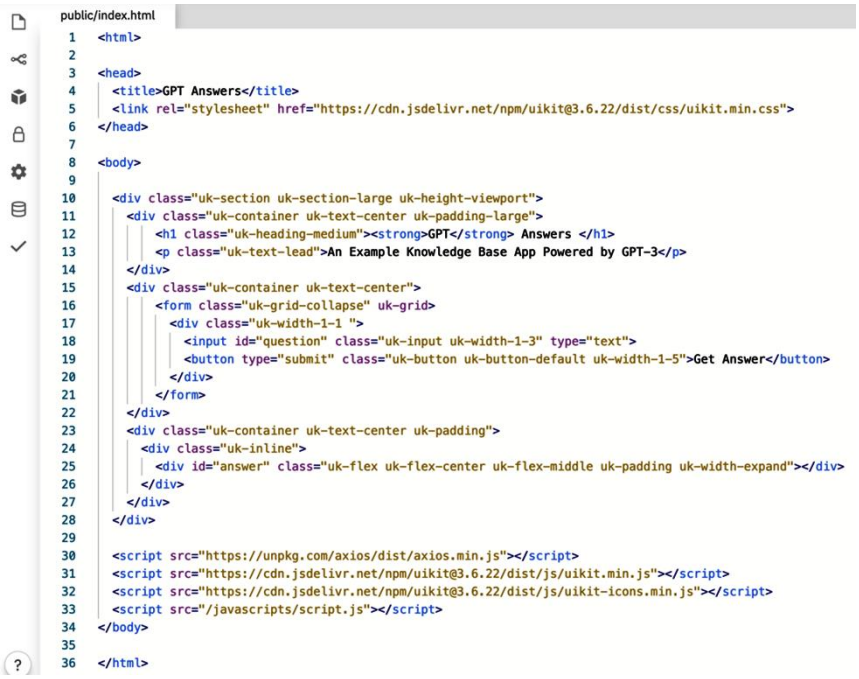
app.js

```
var express = require('express');
var path = require('path');
var cookieParser = require('cookie-parser');
var logger = require('morgan');

var indexRouter = require('./routes/index');
var usersRouter = require('./routes/users');

var app = express();

app.use(logger('dev'));
app.use(express.json());
app.use(express.urlencoded({ extended: false }));
app.use(cookieParser());
app.use(express.static(path.join(__dirname, 'public')));

app.use('/', indexRouter);
app.use('/users', usersRouter);

module.exports = app;
```

app.js

```
var express = require('express');
var path = require('path');
var cookieParser = require('cookie-parser');
var logger = require('morgan');

var indexRouter = require('./routes/index');
var answerRouter = require('./routes/answer');

var app = express();

app.use(logger('dev'));
app.use(express.json());
app.use(express.urlencoded({ extended: false }));
app.use(cookieParser());
app.use(express.static(path.join(__dirname, 'public')));

app.use('/', indexRouter);
app.use('/answer', answerRouter);

module.exports = app;
```


Home
Workspaces
Reports
Explore
Search Postman
Invite
Upgrade
Exploring GPT-3
New
Import
Overview
POST test-answer-api
No Environment
Collections
APIs
Environments
Mock Servers
Monitors
History
Chapter 04
Chapter 05
Chapter 06
Chapter 06
Chapter 07
Chapter 08
Chapter 09
gptanswers-node
POST test-answer-api
gptanswers-node / test-answer-api
Save
POST
https://gptanswers.com/answer
Send
Params
Authorization
Headers (8)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"question" : "How old are you?"
}
Response
Hit Send to get a response


public/index.html
<html>
<head>
<title>GPT Answers</title>
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/uikit@3.6.22/dist/css/uikit.min.css">
</head>
<body>

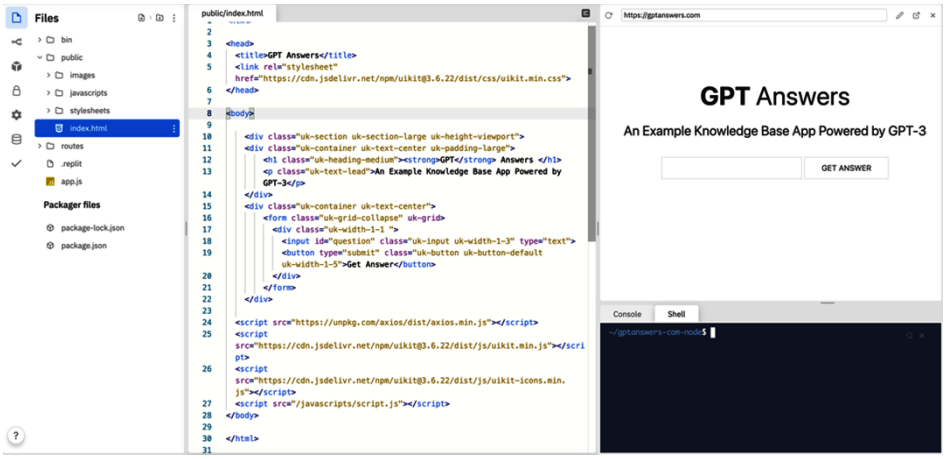
<h1 class="uk-heading-medium">GPT Answers </h1>
An Example Knowledge Base App Powered by GPT-3

<form class="uk-grid-collapse" uk-grid>

<input id="question" class="uk-input uk-width-1-3" type="text">
<button type="submit" class="uk-button uk-button-default uk-width-1-5">Get Answer</button>

</form>

<script src="https://unpkg.com/axios/dist/axios.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/uikit@3.6.22/dist/js/uikit.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/uikit@3.6.22/dist/js/uikit-icons.min.js"></script>
<script src="/javascripts/script.js"></script>
</body>
</html>


Files
bin
public
images
javascripts
stylesheets
index.html
routes
.replit
app.js
Packager files
package-lock.json
package.json
public/index.html
https://gptanswers.com
GPT Answers
An Example Knowledge Base App Powered by GPT-3
GET ANSWER
Console
Shell
~/gptanswers-com-node$

GPT Answers

An Example Knowledge Base App Powered by GPT-3

What is your name? GET ANSWER

placeholder for the answer

routes/answer.js

```
const axios = require('axios');
const express = require('express');
const router = express.Router();

const apiKey = process.env.OPENAI_API_KEY;
const client = axios.create({
  headers: { 'Authorization': 'Bearer ' + apiKey }
});

const documents = [
  "I am a day older than I was yesterday.<|endoftext|>",
  "I build applications that use GPT-3.<|endoftext|>",
  "My preferred programming is Python.<|endoftext|>"
]

const endpoint = 'https://api.openai.com/v1/answers';

router.post('/', (req, res) => {
    const data = {
    "file": process.env.ANSWERS_FILE,
    "question": req.body.question,
    "search_model": "ada",
    "model": "curie",
    "examples_context": "My favorite programming language is Python.",
    "examples": [["How old are you?", "I'm a day older than I was yesterday."], ["What languages do you know?", "I speak English and write code in Python."]],
    "max_tokens": 15,
    "temperature": 0,
    "return_prompt": false,
    "expand": ["completion"],
    "stop": ["\n", "<|endoftext|>"],
  }
  client.post(endpoint, data)
    .then(result => {
      res.send({"answer" : result.data.answers[0]})
    }).catch(result => {
      res.send({"answer" : "Sorry, I don't have an answer."})
    });
});

module.exports = router;
```

Secrets

System environment variables

key

OPENAI_API_KEY

value

sk-vxxXxxXxxBQpi2HG3KAxQ8TBNb2XXxxxXxxX3N

Add new secret

routes/answer.js

```
const axios = require('axios');
const express = require('express');
const router = express.Router();

const apiKey = process.env.OPENAI_API_KEY;
const client = axios.create({
  headers: { 'Authorization': 'Bearer ' + apiKey }
});

const answers = [
  "I am old enough to know not to answer that question.<|endoftext|>",
  "I write books and create video courses to help people learn about GTP-3.<|endoftext|>",
  "My prefered programming languages are C++, C#, JavaScript, Go, and Python.<|endoftext|>"
]

const endpoint = 'https://api.openai.com/v1/answers';

router.post('/', (req, res) => {
  const data = {
    "documents": answers,
    "question": req.body.question || "what is this?",
    "search_model": "ada",
    "model": "curie",
    "examples_context": "My favorite programming language is Python.",
    "examples": [["How old are you?", "I'm a day older than I was yesterday."],
```


GPT Answers

An Example Knowledge Base App Powered by GPT-3

What is your favorite food? GET ANSWER

I like pizza.

files-upload.js

```
const axios = require('axios');
const FormData = require('form-data');

const data = new FormData();
data.append('purpose', 'answers');
data.append('file', fs.createReadStream('answers.jsonl'));

const params = {
  method: 'post',
  url: 'https://api.openai.com/v1/files',
  headers: {
    'Authorization': 'Bearer ' + process.env.OPENAI_API_KEY,
    ...data.getHeaders()
  },
  data: data
}

axios(params)
  .then(function(response) {
    console.log(response.data);
  })
  .catch(function(error) {
    console.log(error.message);
  });

```

Console Shell

```
~/gptanswers-com-node$ node files-upload.js
{
  id: 'file-gGfXHNMv3Fm4J33ebWIobhkm',
  object: 'file',
  bytes: 132,
  created_at: 1621699209,
  filename: 'answers.jsonl',
  purpose: 'answers',
  status: 'uploaded',
  status_details: null
}
~/gptanswers-com-node$
```

```
router.post('/', (req, res) => {
    const data = {
    "file": process.env.ANSWERS_FILE,
    "question": req.body.question,
    "search_model": "ada",
    "model": "curie",
    "examples_context": "My favorite programming language is Python.",
    "examples": [["How old are you?", "I'm a day older than I was yesterday."],
    ["What languages do you know?", "I speak English and write code in Python."]]
    ,
    "max_tokens": 15,
    "temperature": 0,
    "return_prompt": false,
    "expand": ["completion"],
    "stop": ["\n", "<|endoftext|>"],
  }
```

GPT Answers

An Example Knowledge Base App Powered by GPT-3

What is your favorite food? GET ANSWER

Carrot cake.

GPT Answers

An Example Knowledge Base App Powered by GPT-3

Do you sell this one in red?

GET ANSWER

Sorry, I don't have an answer.

GPT Answers

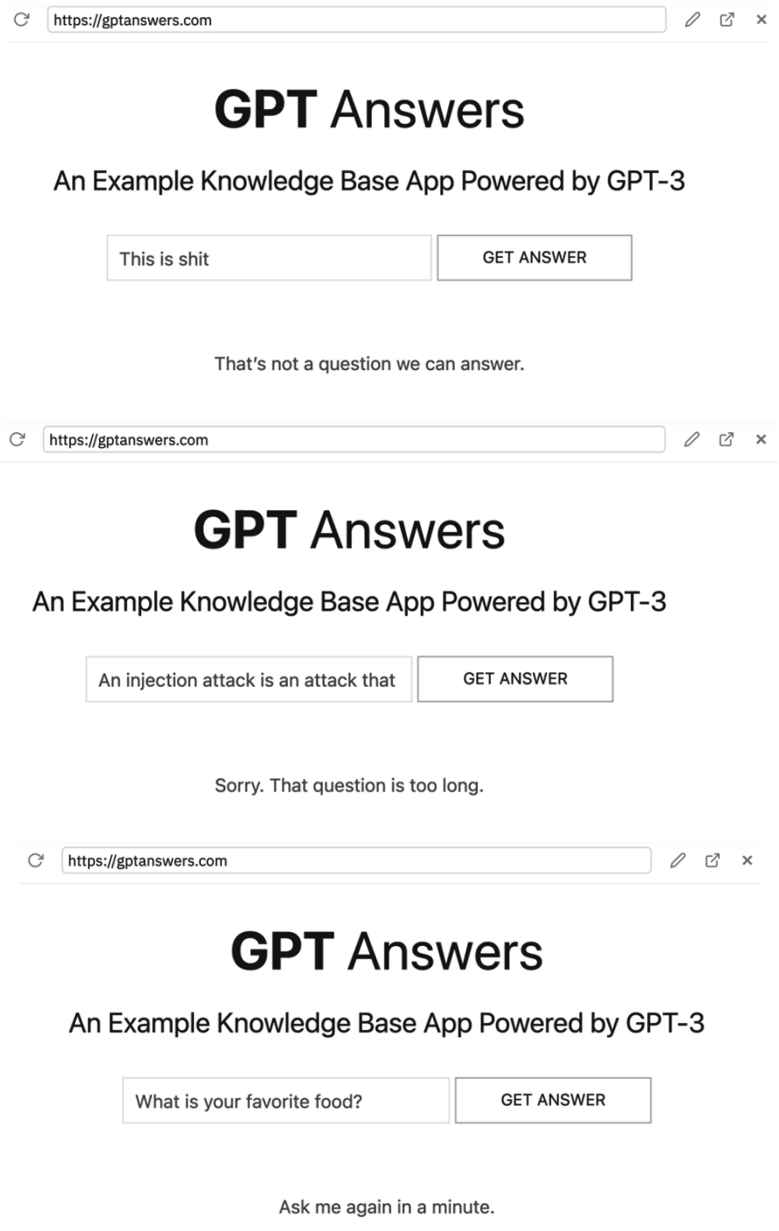
An Example Knowledge Base App Powered by GPT-3

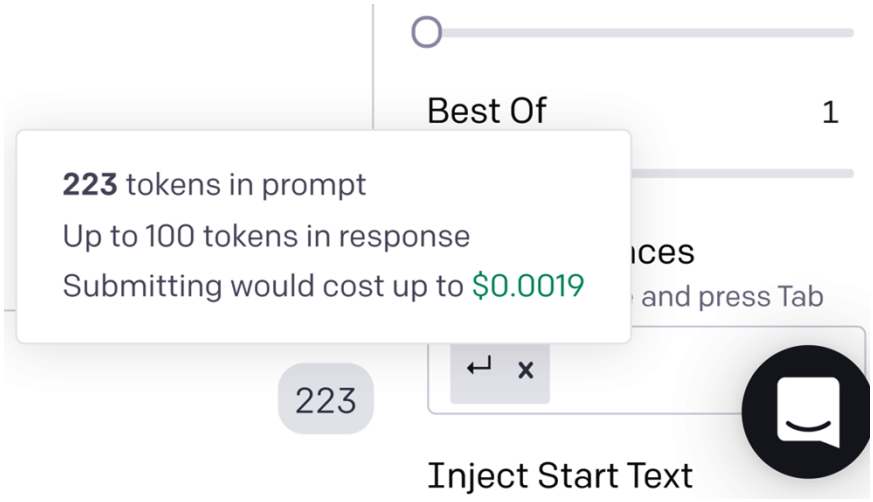
What is your favorite vacation spot?

GET ANSWER

I like to go to the beach.

Chapter 10: Going Live with OpenAI-Powered Apps




Best Of
1
223 tokens in prompt
Up to 100 tokens in response
Submitting would cost up to $0.0019
223
and press Tab
Inject Start Text