

API DASHBOARD

- [Your Account](#)
- [Your API Metrics](#)
- [Your Searches](#)
- [Your Playground](#)
- [API DOCUMENTATION](#)
- [Google Search API](#)
- [Google Light Search API](#)
- [Google Ads Transparency API](#)
- [Google AI Mode API](#)
- [Google AI Overview API](#)
- [Google Autocomplete API](#)
- [Google Events API](#)
- [Google Forums API](#)
- [Google Finance API](#)
- [Google Flights API](#)
- [Google Hotels API](#)
- [Google Images API](#)
- [Google Images Light API](#)
- [Google Immersive Product API](#)
- [Google Jobs API](#)
- [Google Lens API](#)
- [Google Local API](#)
- [Google Local Services API](#)
- [Google Maps API](#)
- [Google Maps Reviews API](#)
- [Google News API](#)
- [Google News Light API](#)
- [Google Patents API](#)
- [Google Play Store API](#)
- [Google Related Questions API](#)
- [Google Reverse Image API](#)
- [Google Scholar API](#)
- [Google Shopping API](#)
- [Google Shopping Light API](#)
- [Google Light Fast API](#)
- [Google Short Videos API](#)
- [Google Travel Explore API](#)
- [Google Trends API](#)
- [Google Videos API](#)
- [Google Videos Light API](#)
- [Amazon Search API](#)
- [Apple App Store API](#)
- [Baidu Search API](#)
- [Bing Images API](#)
- [Bing Copilot API](#)
- [Bing Search API](#)
- [DuckDuckGo Search API](#)
- [DuckDuckGo Light API](#)
- [eBay Search API](#)
- [Facebook Profile API](#)
- [Naver Search API](#)
- [The Home Depot Search API](#)
- [Tripadvisor Search API](#)
- [Walmart Search API](#)
- [Yahoo! Search API](#)
- [Yandex Search API](#)
- [Yelp Search API](#)
- [YouTube Search API](#)
- [Extra APIs](#)
- [Status and Error Codes](#)

Google AI Mode API

API uptime 99.716%

Our Google AI Mode API allows you to scrape results from the Google AI Mode page. The API is accessed through the following endpoint: [/search?](#)
engine=google_ai_mode. A user may query the following:
https://serpapi.com/search?engine=google_ai_mode utilizing a GET request. Head to the [playground](#) for a live and interactive demo.

Note on Query Language Support

Google AI Mode supports multiple languages, including **English**.

Some languages are not yet supported and may not return valid or expected results.

Search Query

Search

AI Mode [All](#) [Images](#) [Videos](#) [News](#) [More](#) ▾

Sign in Tools

Coffee

Coffee is a popular brewed beverage made from the roasted and ground seeds of the coffee plant. It is one of the most widely consumed drinks in the world, prized for its invigorating, caffeinated effect. While its exact origins are debated, coffee likely originated in Ethiopia, with credible evidence of it being consumed in Yemen in the mid-15th century.

Types of beans and roasts

The flavor profile of coffee varies depending on the bean species, growing region, and roasting process.

Bean types:

- **Arabica:** Accounts for roughly 60% of global production. It is a delicate bean that grows at higher altitudes, resulting in a more flavorful and aromatic cup.
- **Robusta:** A heartier, more disease-resistant bean that can grow at lower altitudes. It has a bolder, more bitter flavor and contains 50–60% more caffeine than Arabica beans.

Roast levels:

11 sites

Coffee | Origin, Types, Uses, History, & Facts - Britannica
coffee * What is coffee? Coffee is a beverage brewed from the roasted and ground seeds of the tropical evergreen coffee plant. Coffee is one of the three...

Types of Coffee Drinks: Latte, Cappuccino, Macchiato and More
A Guide to Coffee Drinks: Cappuccino, Latte, Macchiato, Flat White and Beyond. It's all about the espresso-to-milk ratio... Coffee: We drink it, we love it, a...

Coffee - Wikipedia
Table_title: Coffee Table_content: header: | Espresso latte and black filtered coffee || row: | Espresso latte and black filtered coffee: Type | Usually hot,

Ask anything

HTTP Method: GET

```
1 | https://serpapi.com/search.json?engine=google_ai_mode&q=Coffee
```

Code to integrate:

```
1 | require "serpapi"
2 |
3 | client = SerpApi::Client.new(
4 |   engine: "google_ai_mode",
5 |   q: "Coffee",
6 |   api_key: "a85853e3401522bd0dbb7b0d6b4901dc8a5849baae4354be6cea1dd229296958"
7 |
8 |
9 | results = client.search
10 | text_blocks = results[:text_blocks]
```

[CURL](#) [Ruby](#) [Python](#) [JavaScript](#) [Go](#) [PHP](#) [Java](#) [Rust](#) [.NET](#) [Google Sheets](#)

API Parameters

Search Query



Required

Parameter defines the query you want to search. You can use anything that you would use in a regular Google AI Mode search.

Google AI Mode may support multiple languages (including **English**), though not all are available yet.

API DASHBOARD

- [Your Account](#)
- [Your API Metrics](#)
- [Your Searches](#)
- [Your Playground](#)

API DOCUMENTATION

- [Google Search API](#)
- [Google Light Search API](#)
- [Google Ads Transparency API](#)
- [Google AI Mode API](#)
- [Google AI Overview API](#)
- [Google Autocomplete API](#)
- [Google Events API](#)
- [Google Forums API](#)
- [Google Finance API](#)
- [Google Flights API](#)
- [Google Hotels API](#)
- [Google Images API](#)
- [Google Images Light API](#)
- [Google Immersive Product API](#)
- [Google Jobs API](#)
- [Google Lens API](#)
- [Google Local API](#)
- [Google Local Services API](#)
- [Google Maps API](#)
- [Google Maps Reviews API](#)
- [Google News API](#)
- [Google News Light API](#)
- [Google Patents API](#)
- [Google Play Store API](#)
- [Google Related Questions API](#)
- [Google Reverse Image API](#)
- [Google Scholar API](#)
- [Google Shopping API](#)
- [Google Shopping Light API](#)
- [Google Light Fast API](#)
- [Google Short Videos API](#)
- [Google Travel Explore API](#)
- [Google Trends API](#)
- [Google Videos API](#)
- [Google Videos Light API](#)
- [Amazon Search API](#)
- [Apple App Store API](#)
- [Baidu Search API](#)
- [Bing Images API](#)
- [Bing Copilot API](#)
- [Bing Search API](#)
- [DuckDuckGo Search API](#)
- [DuckDuckGo Light API](#)
- [eBay Search API](#)
- [Facebook Profile API](#)
- [Naver Search API](#)
- [The Home Depot Search API](#)
- [Tripadvisor Search API](#)
- [Walmart Search API](#)
- [Yahoo! Search API](#)
- [Yandex Search API](#)
- [Yelp Search API](#)
- [YouTube Search API](#)
- [Extra APIs](#)
- [Status and Error Codes](#)

[location](#)

Optional

Parameter defines from where you want the search to originate. If several locations match the location requested, we'll pick the most popular one. Head to the [/locations.json API](#) if you need more precise control. The [location](#) and [uule](#) parameters can't be used together. It is recommended to specify [location](#) at the city level in order to simulate a real user's search. If [location](#) is omitted, the search may take on the location of the proxy.

[uule](#)

Optional

Parameter is the Google encoded location you want to use for the search.

[uule](#) and [location](#) parameters can't be used together.

Localization

[gl](#)

Optional

Parameter defines the country to use for the Google search. It's a two-letter country code. (e.g., [us](#) for the United States, [uk](#) for United Kingdom, or [fr](#) for France). Head to the [Google countries page](#) for a full list of supported Google countries.

[hi](#)

Optional

Parameter defines the language to use for the Google search. It's a two-letter language code. (e.g., [en](#) for English, [es](#) for Spanish, or [fr](#) for French). Head to the [Google languages page](#) for a full list of supported Google languages.

Serpapi Parameters

[engine](#)

Required

Set parameter to [google_ai_mode](#) to use the Google Ai Mode API engine.

[device](#)

Optional

Parameter defines the device to use to get the results. It can be set to [desktop](#) (default) to use a regular browser, [tablet](#) to use a tablet browser (currently using iPads), or [mobile](#) to use a mobile browser.

[no_cache](#)

Optional

Parameter will force SerpApi to fetch the Google Ai Mode results even if a cached version is already present. A cache is served only if the query and all parameters are exactly the same. Cache expires after 1h. Cached searches are free, and are not counted towards your searches per month. It can be set to [false](#) (default) to allow results from the cache, or [true](#) to disallow results from the cache. [no_cache](#) and [async](#) parameters should not be used together.

[async](#)

Optional

Parameter defines the way you want to submit your search to SerpApi. It can be set to [false](#) (default) to open an HTTP connection and keep it open until you got your search results, or [true](#) to just submit your search to SerpApi and retrieve them later. In this case, you'll need to use our [Searches Archive API](#) to retrieve your results. [async](#) and [no_cache](#) parameters should not be used together. [async](#) should not be used on accounts with [Ludicrous Speed](#) enabled.

[zero_trace](#)

Optional

Enterprise only. Parameter enables [ZeroTrace mode](#). It can be set to [false](#) (default) or [true](#). Enable this mode to skip storing search parameters, search files, and search metadata on our servers. This may make debugging more difficult.

[api_key](#)

Required

Parameter defines the SerpApi private key to use.

[output](#)

Optional

Parameter defines the final output you want. It can be set to [json](#) (default) to get a structured [JSON](#) of the results, or [html](#) to get the raw HTML retrieved.

[json_restrictror](#)

Optional

Parameter defines the fields you want to restrict in the outputs for smaller, faster responses. See [JSON Restrictor](#) for more details.

API Results

JSON Results

JSON output includes structured data for [text_blocks](#), [references](#), [inline_images](#), [inline_products](#), and more.

A search status is accessible through [search_metadata.status](#). It flows this way: [Processing](#) -> [Success](#) || [Error](#). If a search has failed, [error](#) will contain an error message. [search_metadata.id](#) is the search ID inside SerpApi.

HTML Results

HTML output is useful to debug JSON results or support features not supported yet by SerpApi. HTML output gives you the raw HTML results from Google AI Mode.

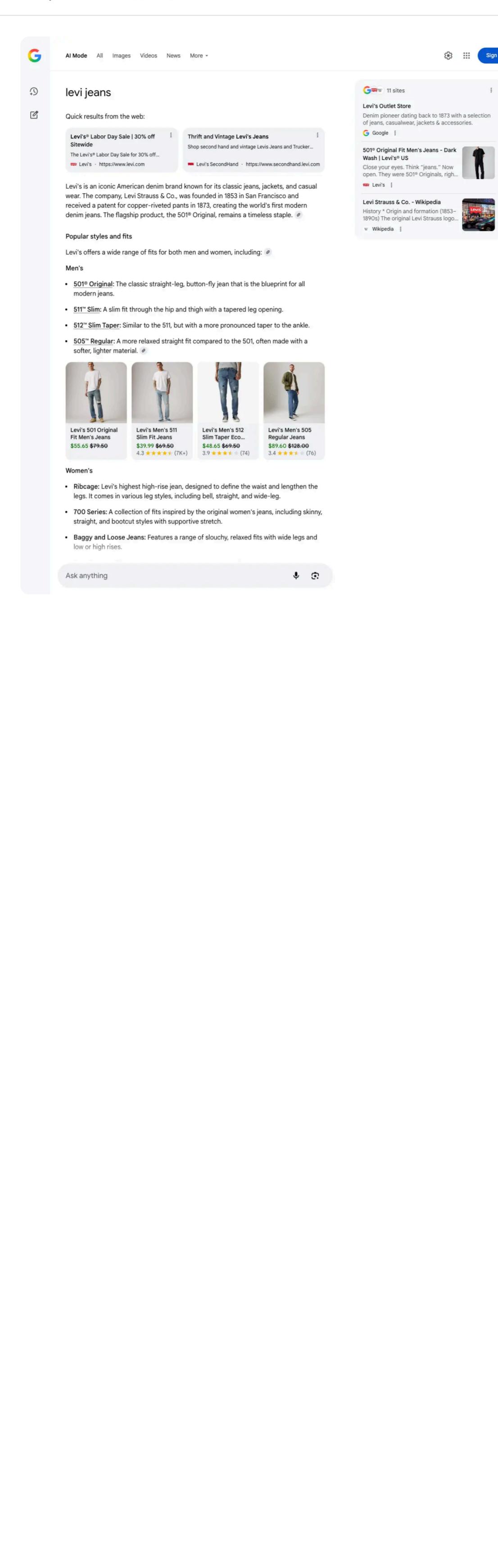
API DASHBOARD

- [Your Account](#)
- [Your API Metrics](#)
- [Your Searches](#)
- [Your Playground](#)

API DOCUMENTATION

- [Google Search API](#)
- [Google Light Search API](#)
- [Google Ads Transparency API](#)
- [Google AI Mode API](#)
- [Google AI Overview API](#)
- [Google Autocomplete API](#)
- [Google Events API](#)
- [Google Forums API](#)
- [Google Finance API](#)
- [Google Flights API](#)
- [Google Hotels API](#)
- [Google Images API](#)
- [Google Images Light API](#)
- [Google Immersive Product API](#)
- [Google Jobs API](#)
- [Google Lens API](#)
- [Google Local API](#)
- [Google Local Services API](#)
- [Google Maps API](#)
- [Google Maps Reviews API](#)
- [Google News API](#)
- [Google News Light API](#)
- [Google Patents API](#)
- [Google Play Store API](#)
- [Google Related Questions API](#)
- [Google Reverse Image API](#)
- [Google Scholar API](#)
- [Google Shopping API](#)
- [Google Shopping Light API](#)
- [Google Light Fast API](#)
- [Google Short Videos API](#)
- [Google Travel Explore API](#)
- [Google Trends API](#)
- [Google Videos API](#)
- [Google Videos Light API](#)
- [Amazon Search API](#)
- [Apple App Store API](#)
- [Baidu Search API](#)
- [Bing Images API](#)
- [Bing Copilot API](#)
- [Bing Search API](#)
- [DuckDuckGo Search API](#)
- [DuckDuckGo Light API](#)
- [eBay Search API](#)
- [Facebook Profile API](#)
- [Naver Search API](#)
- [The Home Depot Search API](#)
- [Tripadvisor Search API](#)
- [Walmart Search API](#)
- [Yahoo! Search API](#)
- [Yandex Search API](#)
- [Yelp Search API](#)
- [YouTube Search API](#)
- [Extra APIs](#)
- [Status and Error Codes](#)

Example with `shopping_results` and `quick_results`



The screenshot shows a Google search results page for the query "levi jeans". The results are divided into two main sections: "Quick results from the web:" and "Levi's Outlet Store".

Quick results from the web:

- Levi's® Labor Day Sale | 30% off Sitemide**: The Levi's Labor Day Sale for 30% off...
- Thrift and Vintage Levi's Jeans**: Shop second hand and vintage Levi's Jeans and Trucker...
- Levi's Secondhand**: Levi's Secondhand - https://www.secondhand.levi.com

Levi's Outlet Store: Levi's pioneer dating back to 1873 with a selection of jeans, casualwear, jackets & accessories.

Popular styles and fits:

- 501® Original: The classic straight-leg, button-fly jean that is the blueprint for all modern jeans.
- 511® Slim: A slim fit through the hip and thigh with a tapered leg opening.
- 512® Slim Taper: Similar to the 511, but with a more pronounced taper to the ankle.
- 505® Regular: A more relaxed straight fit compared to the 501, often made with a softer, lighter material.

Men's and **Women's** sections show product cards for various Levi's jeans models, including 501 Original Fit Men's Jeans, 511 Slim Fit Jeans, 512 Slim Taper Eco Wash Men's Jeans, and 505 Regular Jeans. Each card includes a thumbnail image, product name, price (\$55.65-\$79.90 or \$39.99-\$69.90), and a rating (4.3 or 3.9 stars).

An "Ask anything" input field is at the bottom.

`</> GET`

```
1 | https://serpapi.com/search.json?
2 | engine=google_ai_mode&q=Levi+jeans
```

`</> Code to integrate`

```
1 | require "serpapi"
2 |
3 | client = SerpApi::Client.new(
4 |   engine: "google_ai_mode",
5 |   q: "Levi jeans",
6 |   api_key:
7 | "a85853e3401522bd0dbb7b0d6b4901dc8a5849baae4354be6cea1dd229296958"
8 |
9 |
10| results = client.search
| shopping_results = results[:shopping_results]
```

`</> JSON Example`

API DASHBOARD

- [Your Account](#)
- [Your API Metrics](#)
- [Your Searches](#)
- [Your Playground](#)

API DOCUMENTATION

- [Google Search API](#)
- [Google Light Search API](#)
- [Google Ads Transparency API](#)
- [Google AI Mode API](#)
- [Google AI Overview API](#)
- [Google Autocomplete API](#)
- [Google Events API](#)
- [Google Forums API](#)
- [Google Finance API](#)
- [Google Flights API](#)
- [Google Hotels API](#)
- [Google Images API](#)
- [Google Images Light API](#)
- [Google Immersive Product API](#)
- [Google Jobs API](#)
- [Google Lens API](#)
- [Google Local API](#)
- [Google Local Services API](#)
- [Google Maps API](#)
- [Google Maps Reviews API](#)
- [Google News API](#)
- [Google News Light API](#)
- [Google Patents API](#)
- [Google Play Store API](#)
- [Google Related Questions API](#)
- [Google Reverse Image API](#)
- [Google Scholar API](#)
- [Google Shopping API](#)
- [Google Shopping Light API](#)
- [Google Light Fast API](#)
- [Google Short Videos API](#)
- [Google Travel Explore API](#)
- [Google Trends API](#)
- [Google Videos API](#)
- [Google Videos Light API](#)
- [Amazon Search API](#)
- [Apple App Store API](#)
- [Baidu Search API](#)
- [Bing Images API](#)
- [Bing Copilot API](#)
- [Bing Search API](#)
- [DuckDuckGo Search API](#)
- [DuckDuckGo Light API](#)
- [eBay Search API](#)
- [Facebook Profile API](#)
- [Naver Search API](#)
- [The Home Depot Search API](#)
- [Tripadvisor Search API](#)
- [Walmart Search API](#)
- [Yahoo! Search API](#)
- [Yandex Search API](#)
- [Yelp Search API](#)
- [YouTube Search API](#)
- [Extra APIs](#)
- [Status and Error Codes](#)

```
{
  "title": "Levi's® Labor Day Sale | 30% off Sitewide",
  "link": "https://www.levi.com/US/en_US/",
  "snippet": "The Levi's® Labor Day Sale for 30% off sitewide, shop and s",
  "source": "Levi's",
  "displayed_link": "https://www.levi.com",
  "favicon": "https://serpapi.com/searches/68af016fdbbb466318f53099/image"
},
{
  "title": "Thrift and Vintage Levi's Jeans",
  "link": "https://www.secondhand.levi.com/shop/jeans",
  "snippet": "Shop second hand and vintage Levis Jeans and Trucker Jacket",
  "source": "Levi's SecondHand",
  "displayed_link": "https://www.secondhand.levi.com",
  "favicon": "https://serpapi.com/searches/68af016fdbbb466318f53099/image"
},
],
"shopping_results": [
{
  "title": "Levi's 501 Original Fit Men's Jeans",
  "product_link": "https://www.google.com/search?prds=pvt:hg,pvo:29,mid:5",
  "immersive_product_page_token": "eyJlaSI6bnVsbCwichJvZHjdGlkIjpuWxsLC",
  "serpapi_impressive_product_api": "https://serpapi.com/search.json?engin",
  "thumbnail": "https://encrypted-tbn0.gstatic.com/shopping?q=tbn:ANd9Gcs",
  "price": "$55.65",
  "extracted_price": 55.65,
  "old_price": "$79.50",
  "extracted_old_price": 79.5,
  "index": 0
},
{
  "title": "Levi's Men's 511 Slim Fit Jeans",
  "product_link": "https://www.google.com/search?prds=pvt:hg,pvo:29,mid:5",
  "immersive_product_page_token": "eyJjYXRhbG9naWQlOiIxNTk5OTUwNTg0NjA4MC",
  "serpapi_impressive_product_api": "https://serpapi.com/search.json?engin",
  "thumbnail": "https://encrypted-tbn0.gstatic.com/shopping?q=tbn:ANd9Gcs",
  "price": "$39.99",
  "extracted_price": 39.99,
  "old_price": "$69.50",
  "extracted_old_price": 69.5,
  "rating": 4.3,
  "reviews": 7000,
  "index": 1
},
{
  "title": "Levi's Men's 512 Slim Taper Eco Performance Jeans",
  "product_link": "https://www.google.com/search?prds=pvt:hg,pvo:29,mid:5",
  "immersive_product_page_token": "eyJjYXRhbG9naWQlOiIxNjQ5MjgyMzE5NzQ3Mz",
  "serpapi_impressive_product_api": "https://serpapi.com/search.json?engin",
  "thumbnail": "https://encrypted-tbn3.gstatic.com/shopping?q=tbn:ANd9Gcs",
  "price": "$48.65",
  "extracted_price": 48.65,
  "old_price": "$69.50",
  "extracted_old_price": 69.5,
  "rating": 3.9,
  "reviews": 74,
  "index": 2
},
...
],
"text_blocks": [
{
  "type": "paragraph",
  "snippet": "Levi's is an iconic American denim brand known for its clas",
  "reference_indexes": [
    4,
    0,
    1,
    2
  ]
},
...
{
  "type": "paragraph",
  "snippet": "Men's",
  "reference_indexes": [
    "Men's"
  ]
},
{
  "type": "list",
  "list": [
    {
      "snippet": "501® Original: The classic straight-leg, button-fly jac",
      "snippet_links": [
        {
          "text": "501® Original",
          "link": "https://www.google.com/search?prds=pvt:hg,pvo:29,mid:5",
          "shopping_results_reference_index": 0
        }
      ]
    },
    ...
    {
      "snippet": "511™ Slim: A slim fit through the hip and thigh with a",
      "snippet_links": [
        {
          "text": "511™ Slim",
          "link": "https://www.google.com/search?prds=pvt:hg,pvo:29,mid:5",
          "shopping_results_reference_index": 1
        }
      ]
    },
    ...
    {
      "snippet": "512™ Slim Taper: Similar to the 511, but with a more pr",
      "snippet_links": [
        {
          "text": "512™ Slim Taper",
          "link": "https://www.google.com/search?prds=pvt:hg,pvo:29,mid:5",
          "shopping_results_reference_index": 2
        }
      ]
    },
    ...
  ],
  "reference_indexes": [
    1,
    6
  ]
},
...
]
```

API DASHBOARD

[Your Account](#)[Your API Metrics](#)[Your Searches](#)[Your Playground](#)

API DOCUMENTATION

[G Google Search API](#)[G Google Light Search API](#)[G Google Ads Transparency API](#)[G Google AI Mode API](#)[G Google AI Overview API](#)[G Google Autocomplete API](#)[G Google Events API](#)[G Google Forums API](#)[G Google Finance API](#)[G Google Flights API](#)[G Google Hotels API](#)[G Google Images API](#)[G Google Images Light API](#)[G Google Immersive Product API](#)[G Google Jobs API](#)[G Google Lens API](#)[G Google Local API](#)[G Google Local Services API](#)[G Google Maps API](#)[G Google Maps Reviews API](#)[G Google News API](#)[G Google News Light API](#)[G Google Patents API](#)[G Google Play Store API](#)[G Google Related Questions API](#)[G Google Reverse Image API](#)[G Google Scholar API](#)[G Google Shopping API](#)[G Google Shopping Light API](#)[G Google Light Fast API](#)[G Google Short Videos API](#)[G Google Travel Explore API](#)[B Google Trends API](#)[B Google Videos API](#)[B Google Videos Light API](#)[A Amazon Search API](#)[A Apple App Store API](#)[Baidu Search API](#)[B Bing Images API](#)[B Bing Copilot API](#)[B Bing Search API](#)[D DuckDuckGo Search API](#)[D DuckDuckGo Light API](#)[e eBay Search API](#)[F Facebook Profile API](#)[N Naver Search API](#)[T The Home Depot Search API](#)[T TripAdvisor Search API](#)[W Walmart Search API](#)[Y Yahoo! Search API](#)[Y Yandex Search API](#)[Y Yelp Search API](#)[Y YouTube Search API](#)[E Extra APIs](#)[S Status and Error Codes](#)

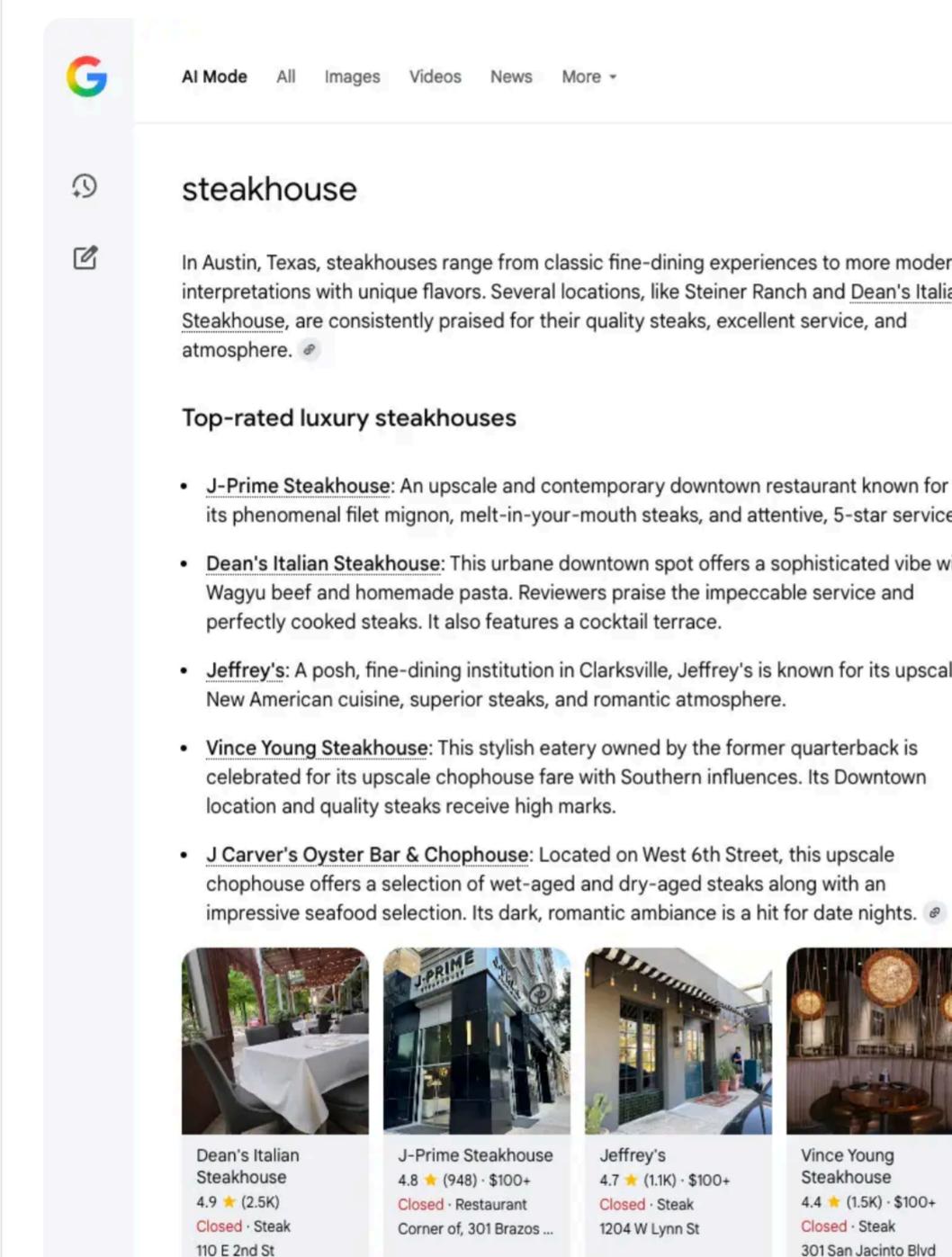
/

```

    "title": "Levi's® Outlet Store",
    "link": "https://www.google.com/viewer/place?mid=g/1vr1wdxw",
    "snippet": "Denim pioneer dating back to 1873 with a selection of jeans",
    "source": "Google",
    "index": 0
},
{
    "title": "501® Original Fit Men's Jeans - Dark Wash | Levi's® US",
    "link": "https://www.levi.com/US/en_US/clothing/men/jeans/straight/501-",
    "snippet": "Close your eyes. Think \"jeans.\" Now open. They were 501® Or",
    "source": "Levi's",
    "index": 1
},
{
    "title": "Levi Strauss & Co. - Wikipedia",
    "link": "https://en.wikipedia.org/wiki/Levi_Strauss_%26_Co.",
    "snippet": "History * Origin and formation (1853–1890s) The original Le",
    "source": "Wikipedia",
    "index": 2
},
...
],
...
}

```

Example with `local_results`

A screenshot of a Google search results page for the query "steakhouse". The interface shows the "AI Mode" button is active. The results are categorized under "Top-rated luxury steakhouses". Below this, there is a section titled "Steakhouses with a view or unique atmosphere". At the bottom, there is a "See more" button and a "Ask anything" input field.

Restaurant	Rating	Address
Dean's Italian Steakhouse	4.9 ★ (2.5K)	110 E 2nd St
J-Prime Steakhouse	4.8 ★ (948)	Corner of, 301 Brazos ...
Jeffrey's	4.7 ★ (1.1K)	1204 W Lynn St
Vince Young Steakhouse	4.4 ★ (1.5K)	301 San Jacinto Blvd

↔ GET

```
1 | https://serpapi.com/search.json?
engine=google_ai_mode&q=Steakhouse&location=Austin,+Texas,+United+States
```

↔ Code to integrate

Ruby

```

1 | require "serpapi"
2 |
3 | client = SerpApi::Client.new(
4 |   engine: "google_ai_mode",
5 |   q: "Steakhouse",
6 |   location: "Austin, Texas, United States",
7 |   api_key:
8 | "a85853e3401522bd0dbb7b0d6b4901dc8a5849baae4354be6cea1dd229296958"
9 |
10| results = client.search
11| local_results = results[:local_results]
```

{ JSON Example



API DASHBOARD

[Your Account](#)[Your API Metrics](#)[Your Searches](#)[Your Playground](#)

API DOCUMENTATION

[Google Search API](#)[Google Light Search API](#)[Google Ads Transparency API](#)[Google AI Mode API](#)[Google AI Overview API](#)[Google Autocomplete API](#)[Google Events API](#)[Google Forums API](#)[Google Finance API](#)[Google Flights API](#)[Google Hotels API](#)[Google Images API](#)[Google Images Light API](#)[Google Immersive Product API](#)[Google Jobs API](#)[Google Lens API](#)[Google Local API](#)[Google Local Services API](#)[Google Maps API](#)[Google Maps Reviews API](#)[Google News API](#)[Google News Light API](#)[Google Patents API](#)[Google Play Store API](#)[Google Related Questions API](#)[Google Reverse Image API](#)[Google Scholar API](#)[Google Shopping API](#)[Google Shopping Light API](#)[Google Light Fast API](#)[Google Short Videos API](#)[Google Travel Explore API](#)[Google Trends API](#)[Google Videos API](#)[Google Videos Light API](#)[Amazon Search API](#)[Apple App Store API](#)[Baidu Search API](#)[Bing Images API](#)[Bing Copilot API](#)[Bing Search API](#)[DuckDuckGo Search API](#)[DuckDuckGo Light API](#)[eBay Search API](#)[Facebook Profile API](#)[Naver Search API](#)[The Home Depot Search API](#)[Tripadvisor Search API](#)[Walmart Search API](#)[Yahoo! Search API](#)[Yandex Search API](#)[Yelp Search API](#)[YouTube Search API](#)[Extra APIs](#)[Status and Error Codes](#)

```
{
    "title": "Dean's Italian Steakhouse",
    "link": "https://www.google.com/viewer/place?mid=%2Fg%2F11pyc12_7j",
    "thumbnail": "https://lh3.googleusercontent.com/gps-cs-s/AC9h4nq9PWzVet",
    "rating": 4.9,
    "reviews": 2500,
    "hours": "Closed",
    "type": "Steak",
    "address": "110 E 2nd St",
    "index": 0
},
{
    "title": "J-Prime Steakhouse",
    "link": "https://www.google.com/viewer/place?mid=%2Fg%2F11rv6jvz8n",
    "thumbnail": "https://lh3.googleusercontent.com/gps-cs-s/AC9h4noBVrXsb",
    "rating": 4.8,
    "reviews": 948,
    "price": "$100+",
    "hours": "Closed",
    "type": "Restaurant",
    "address": "Corner of, 301 Brazos St ste 150, San Jacinto Blvd and 3rd",
    "index": 1
},
{
    "title": "Jeffrey's",
    "link": "https://www.google.com/viewer/place?mid=%2Fm%2F020y__p",
    "thumbnail": "https://lh3.googleusercontent.com/gps-cs-s/AC9h4nozMCmu6E",
    "rating": 4.7,
    "reviews": 1100,
    "price": "$100+",
    "hours": "Closed",
    "type": "Steak",
    "address": "1204 W Lynn St",
    "index": 2
},
...
],
"text_blocks": [
{
    "type": "paragraph",
    "snippet": "In Austin, Texas, steakhouses range from classic fine-dinir",
    "snippet_links": [
        {
            "text": "Dean's Italian Steakhouse",
            "link": "https://www.google.com/viewer/place?mid=/g/11pyc12_7j&sa=x",
            "local_results_reference_index": 0
        }
    ],
    "reference_indexes": [
        1,
        0,
        2
    ]
},
{
    "type": "heading",
    "snippet": "Top-rated luxury steakhouses"
},
{
    "type": "list",
    "list": [
        {
            "snippet": "J-Prime Steakhouse: An upscale and contemporary downtown",
            "snippet_links": [
                {
                    "text": "J-Prime Steakhouse",
                    "link": "https://www.google.com/viewer/place?mid=/g/11rv6jvz8n&sa=x",
                    "local_results_reference_index": 1
                }
            ]
        },
        {
            "snippet": "Dean's Italian Steakhouse: This urbane downtown spot of",
            "snippet_links": [
                {
                    "text": "Dean's Italian Steakhouse",
                    "link": "https://www.google.com/viewer/place?mid=/g/11pyc12_7j&sa=x",
                    "local_results_reference_index": 0
                }
            ]
        }
    ],
    "reference_indexes": [
        0
    ]
},
{
    "snippet": "Jeffrey's: A posh, fine-dining institution in Clarksvil",
    "snippet_links": [
        {
            "text": "Jeffrey's",
            "link": "https://www.google.com/viewer/place?mid=/m/020y__p&sa=x",
            "local_results_reference_index": 2
        }
    ]
},
...
],
"reference_indexes": [
    0
],
...
],
"references": [
{
    "title": "TOP 10 BEST Old School Steak near Austin, TX 78701 - Yelp",
    "link": "https://www.yelp.com/search?find_desc=Old+School+Steak&find_loc=",
    "snippet": "* 1. Seven Grand. 4.5 (135 reviews) 0.3 mi. 405 E 7th St, A",
    "source": "Yelp",
    "index": 0
},
{
    "title": "J-Prime Steakhouse",
    "link": "https://www.google.com/viewer/place?mid=/g/11rv6jvz8n",
    "snippet": "Upscale restaurant with a contemporary vibe serving filet m",
    "source": "Google",
    "index": 1
},
{
    "title": "Steiner Ranch Steakhouse",
    "link": "https://www.google.com/viewer/place?mid=/g/1tx8tmmd",
    "snippet": "Sophisticated steakhouse serving prime steaks & wines, with",
    "source": "Google",
    "index": 2
}
]
```

API DASHBOARD

- [Your Account](#)
- [Your API Metrics](#)
- [Your Searches](#)
- [Your Playground](#)

API DOCUMENTATION

- [Google Search API](#)
- [Google Light Search API](#)
- [Google Ads Transparency API](#)
- [Google AI Mode API](#)
- [Google AI Overview API](#)
- [Google Autocomplete API](#)
- [Google Events API](#)
- [Google Forums API](#)
- [Google Finance API](#)
- [Google Flights API](#)
- [Google Hotels API](#)
- [Google Images API](#)
- [Google Images Light API](#)
- [Google Immersive Product API](#)
- [Google Jobs API](#)
- [Google Lens API](#)
- [Google Local API](#)
- [Google Local Services API](#)
- [Google Maps API](#)
- [Google Maps Reviews API](#)
- [Google News API](#)
- [Google News Light API](#)
- [Google Patents API](#)
- [Google Play Store API](#)
- [Google Related Questions API](#)
- [Google Reverse Image API](#)
- [Google Scholar API](#)
- [Google Shopping API](#)
- [Google Shopping Light API](#)
- [Google Light Fast API](#)
- [Google Short Videos API](#)
- [Google Travel Explore API](#)
- [Google Trends API](#)
- [Google Videos API](#)
- [Google Videos Light API](#)
- [Amazon Search API](#)
- [Apple App Store API](#)
- [Baidu Search API](#)
- [Bing Images API](#)
- [Bing Copilot API](#)
- [Bing Search API](#)
- [DuckDuckGo Search API](#)
- [DuckDuckGo Light API](#)
- [eBay Search API](#)
- [Facebook Profile API](#)
- [Naver Search API](#)
- [The Home Depot Search API](#)
- [Tripadvisor Search API](#)
- [Walmart Search API](#)
- [Yahoo! Search API](#)
- [Yandex Search API](#)
- [Yelp Search API](#)
- [YouTube Search API](#)
- [Extra APIs](#)
- [Status and Error Codes](#)

Example with **table**

The screenshot shows a search results page for "lebron james" on a web browser. The results include a snippet from LeBron James' Wikipedia page, a table of his career achievements, and a section about his latest season statistics.

Team	Seasons Played
Cleveland Cavaliers	2003–2010, 2014–2018
Miami Heat	2010–2014
Los Angeles Lakers	2018–present

↔/ GET

```
1 | https://serpapi.com/search.json?
engine=google_ai_mode&q=Lebron+James
```

↔/ Code to integrate

```
1 | require "serpapi"
2 |
3 | client = SerpApi::Client.new(
4 |   engine: "google_ai_mode",
5 |   q: "Lebron James",
6 |   api_key:
7 | "a85853e3401522bd0dbb7b0d6b4901dc8a5849baae4354be6cea1dd229296958"
8 | )
9 |
10 | results = client.search
|   text_blocks = results[:text_blocks]
```

Ruby

{ } JSON Example

API DASHBOARD

- [Your Account](#)
- [Your API Metrics](#)
- [Your Searches](#)
- [Your Playground](#)

- [Google Search API](#)
- [Google Light Search API](#)
- [Google Ads Transparency API](#)
- [Google AI Mode API](#)
- [Google AI Overview API](#)
- [Google Autocomplete API](#)
- [Google Events API](#)
- [Google Forums API](#)
- [Google Finance API](#)
- [Google Flights API](#)
- [Google Hotels API](#)
- [Google Images API](#)
- [Google Images Light API](#)
- [Google Immersive Product API](#)
- [Google Jobs API](#)
- [Google Lens API](#)
- [Google Local API](#)
- [Google Local Services API](#)
- [Google Maps API](#)
- [Google Maps Reviews API](#)
- [Google News API](#)
- [Google News Light API](#)
- [Google Patents API](#)
- [Google Play Store API](#)
- [Google Related Questions API](#)
- [Google Reverse Image API](#)
- [Google Scholar API](#)
- [Google Shopping API](#)
- [Google Shopping Light API](#)
- [Google Light Fast API](#)
- [Google Short Videos API](#)
- [Google Travel Explore API](#)
- [Google Trends API](#)
- [Google Videos API](#)
- [Google Videos Light API](#)
- [Amazon Search API](#)
- [Apple App Store API](#)
- [Baidu Search API](#)
- [Bing Images API](#)
- [Bing Copilot API](#)
- [Bing Search API](#)
- [DuckDuckGo Search API](#)
- [DuckDuckGo Light API](#)
- [eBay Search API](#)
- [Facebook Profile API](#)
- [Naver Search API](#)
- [The Home Depot Search API](#)
- [Tripadvisor Search API](#)
- [Walmart Search API](#)
- [Yahoo! Search API](#)
- [Yandex Search API](#)
- [Yelp Search API](#)
- [YouTube Search API](#)
- [Extra APIs](#)
- [Status and Error Codes](#)



/

[AI Mode](#) [All](#) [Images](#) [Videos](#) [News](#) [More](#)

• **Relationship with the Supreme Court:** Trump's policies have faced legal review by the Supreme Court. A July 2025 ruling limited the power of lower-court judges to issue nationwide injunctions, while other cases involving his policies on immigration and federal agency officials are ongoing.

• **Foreign affairs:** On August 27, 2025, European leaders expressed concerns about the stability of the economy after Trump's actions related to the Federal Reserve. The administration also faces challenges with EU leaders over potential tariffs and with European allies over peace efforts in Ukraine. [🔗](#)

Status of legal cases

Following his election, several criminal cases against Trump were either dropped or put aside, in line with the Justice Department policy against prosecuting a sitting president. However, civil lawsuits against him continue, some of which are still under appeal. [🔗](#)

- **New York hush money case:** Trump was found guilty in May 2024 of falsifying business records. In January 2025, the judge sentenced him to an unconditional discharge, sparing him any jail time, and he is appealing the conviction.
- **Federal election and classified documents cases:** Special Counsel Jack Smith moved to drop these federal cases after Trump's election, citing the policy against prosecuting a sitting president.
- **Georgia election interference case:** The criminal case brought by Fulton County District Attorney Fani Willis stalled over revelations about her relationship with a prosecutor on the case. The case was removed from Willis's purview by a state appeals court in December 2024, and the future of the prosecution is uncertain.
- **New York civil fraud case:** In August 2025, a New York appeals court overturned the massive financial penalty against Trump, ruling it was an "excessive fine" under the Constitution. However, other restrictions on his business activities were left in place pending further appeal.
- **E. Jean Carroll defamation lawsuits:** Trump was found liable for sexual abuse and defamation against E. Jean Carroll in 2023 and ordered to pay damages in a second trial in January 2024. He is currently appealing the verdicts. [🔗](#)

AI responses may include mistakes. [Learn more](#)

Ask anything

🔗 GET

```
1 | https://serpapi.com/search.json?
  engine=google_ai_mode&q=Donald+Trump
```

🔗 Code to integrate

Ruby

```
1 | require "serpapi"
2 |
3 | client = SerpApi::Client.new(
4 |   engine: "google_ai_mode",
5 |   q: "Donald Trump",
6 |   api_key:
7 | "a85853e3401522bd0dbb7b0d6b4901dc8a5849baae4354be6cea1dd229296958"
8 | )
9 |
10 results = client.search
  inline_videos = results[:inline_videos]
```

🔗 JSON Example

```
{
  ...
  "inline_videos": [
    {
      "title": "How many wars has Donald Trump really stopped? | BBC ...",
      "link": "https://www.youtube.com/watch?v=l1gtEK0ot1U",
      "thumbnail": "https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTGe...",
      "channel": "BBC News",
      "duration": "32:48",
      "platform": "YouTube"
    },
    {
      "title": "US Breaking News LIVE: Trump Issues Urgent Call For ...",
      "link": "https://www.youtube.com/watch?v=FRs0jtFxA14",
      "thumbnail": "https://encrypted-tbn1.gstatic.com/images?q=tbn:ANd9GcTq...",
      "channel": "ET NOW",
      "duration": "00:00",
      "platform": "YouTube"
    },
    {
      "title": "Donald Trump: 'A lot of people are saying maybe we'd like a ...",
      "link": "https://www.youtube.com/watch?v=koruWF1cfyc",
      "thumbnail": "https://encrypted-tbn1.gstatic.com/images?q=tbn:ANd9GcSp7...",
      "channel": "FOX 9 Minneapolis-St. Paul",
      "duration": "01:54",
      "platform": "YouTube"
    },
    ...
  ],
  ...
}
```

API DASHBOARD

[Your Account](#)[Your API Metrics](#)[Your Searches](#)[Your Playground](#)

API DOCUMENTATION

[Google Search API](#)[Google Light Search API](#)[Google Ads Transparency API](#)[Google AI Mode API](#)[Google AI Overview API](#)[Google Autocomplete API](#)[Google Events API](#)[Google Forums API](#)[Google Finance API](#)[Google Flights API](#)[Google Hotels API](#)[Google Images API](#)[Google Images Light API](#)[Google Immersive Product API](#)[Google Jobs API](#)[Google Lens API](#)[Google Local API](#)[Google Local Services API](#)[Google Maps API](#)[Google Maps Reviews API](#)[Google News API](#)[Google News Light API](#)[Google Patents API](#)[Google Play Store API](#)[Google Related Questions API](#)[Google Reverse Image API](#)[Google Scholar API](#)[Google Shopping API](#)[Google Shopping Light API](#)[Google Light Fast API](#)[Google Short Videos API](#)[Google Travel Explore API](#)[Google Trends API](#)[Google Videos API](#)[Google Videos Light API](#)[Amazon Search API](#)[Apple App Store API](#)[Baidu Search API](#)[Bing Images API](#)[Bing Copilot API](#)[Bing Search API](#)[DuckDuckGo Search API](#)[DuckDuckGo Light API](#)[eBay Search API](#)[Facebook Profile API](#)[Naver Search API](#)[The Home Depot Search API](#)[Tripadvisor Search API](#)[Walmart Search API](#)[Yahoo! Search API](#)[Yandex Search API](#)[Yelp Search API](#)[YouTube Search API](#)[Extra APIs](#)[Status and Error Codes](#)

AI Mode All Images Videos News More

ruby for each loop

Quick results from the web:

"for" vs "each" in Ruby - Stack Overflow
With the for loop, the iterator variable still lives...

[Stack Overflow · https://stackoverflow.com](#)

How to Use the Each Method in Ruby | Mix & Go
each is just another method on an object. That...

[mixandgo.com](#)

In Ruby, the "for each loop" is typically implemented using the `each` method, which is the preferred and most common way to iterate over collections like arrays, hashes, and ranges. The `each` method handles the iteration for you, allowing you to focus on the code block that executes for every item.

Syntax and examples

Iterating over an array

The `each` method is called on the array, and a block of code is defined to handle each element. The temporary variable within the pipes `|element|` represents the current item in each iteration.

ruby

```
fruits = ["apple", "banana", "orange"]

fruits.each do |fruit|
  puts "A tasty #{fruit}!"
end
```

Output:
A tasty apple!
A tasty banana!
A tasty orange!

Use code with caution.

</> GET

1 | [https://serpapi.com/search.json?
engine=google_ai_mode&q=ruby+for+each+loop](https://serpapi.com/search.json?engine=google_ai_mode&q=ruby+for+each+loop)

</> Code to integrate

```
1 | require "serpapi"
2 |
3 | client = SerpApi::Client.new(
4 |   engine: "google_ai_mode",
5 |   q: "ruby for each loop",
6 |   api_key:
7 | "a85853e3401522bd0dbb7b0d6b4901dc8a5849baae4354be6cea1dd229296958"
8 |
9 | results = client.search
10 | code_block = results[:code_block]
```

Ruby

{ } JSON Example

Ask anything





API DASHBOARD

[Your Account](#)[Your API Metrics](#)[Your Searches](#)[Your Playground](#)

API DOCUMENTATION

 [Google Search API](#) [Google Light Search API](#) [Google Ads Transparency API](#) [Google AI Mode API](#) [Google AI Overview API](#) [Google Autocomplete API](#) [Google Events API](#) [Google Forums API](#) [Google Finance API](#) [Google Flights API](#) [Google Hotels API](#) [Google Images API](#) [Google Images Light API](#) [Google Immersive Product API](#) [Google Jobs API](#) [Google Lens API](#) [Google Local API](#)[Google Local Services API](#) [Google Maps API](#) [Google Maps Reviews API](#) [Google News API](#) [Google News Light API](#) [Google Patents API](#) [Google Play Store API](#) [Google Related Questions API](#) [Google Reverse Image API](#) [Google Scholar API](#) [Google Shopping API](#) [Google Shopping Light API](#) [Google Light Fast API](#) [Google Short Videos API](#)[Google Travel Explore API](#) [Google Trends API](#) [Google Videos API](#) [Google Videos Light API](#) [Amazon Search API](#) [Apple App Store API](#)[Baidu Search API](#) [Bing Images API](#) [Bing Copilot API](#)[Bing Search API](#) [DuckDuckGo Search API](#) [DuckDuckGo Light API](#)[eBay Search API](#) [Facebook Profile API](#)[Naver Search API](#)[The Home Depot Search API](#)[Tripadvisor Search API](#)[Walmart Search API](#) [Yahoo! Search API](#) [Yandex Search API](#) [Yelp Search API](#) [YouTube Search API](#) [Extra APIs](#) [Status and Error Codes](#)

/

},
...
],
...
}

API DASHBOARD

- [Your Account](#)
- [Your API Metrics](#)
- [Your Searches](#)
- [Your Playground](#)

- [Google Search API](#)
- [Google Light Search API](#)
- [Google Ads Transparency API](#)
- [Google AI Mode API](#)
- [Google AI Overview API](#)
- [Google Autocomplete API](#)
- [Google Events API](#)
- [Google Forums API](#)
- [Google Finance API](#)
- [Google Flights API](#)
- [Google Hotels API](#)
- [Google Images API](#)
- [Google Images Light API](#)
- [Google Immersive Product API](#)
- [Google Jobs API](#)
- [Google Lens API](#)
- [Google Local API](#)
- [Google Local Services API](#)
- [Google Maps API](#)
- [Google Maps Reviews API](#)
- [Google News API](#)
- [Google News Light API](#)
- [Google Patents API](#)
- [Google Play Store API](#)
- [Google Related Questions API](#)
- [Google Reverse Image API](#)
- [Google Scholar API](#)
- [Google Shopping API](#)
- [Google Shopping Light API](#)
- [Google Light Fast API](#)
- [Google Short Videos API](#)
- [Google Travel Explore API](#)
- [Google Trends API](#)
- [Google Videos API](#)
- [Google Videos Light API](#)
- [Amazon Search API](#)
- [Apple App Store API](#)
- [Baidu Search API](#)
- [Bing Images API](#)
- [Bing Copilot API](#)
- [Bing Search API](#)
- [DuckDuckGo Search API](#)
- [DuckDuckGo Light API](#)
- [eBay Search API](#)
- [Facebook Profile API](#)
- [Naver Search API](#)
- [The Home Depot Search API](#)
- [Tripadvisor Search API](#)
- [Walmart Search API](#)
- [Yahoo! Search API](#)
- [Yandex Search API](#)
- [Yelp Search API](#)
- [YouTube Search API](#)
- [Extra APIs](#)
- [Status and Error Codes](#)

```
{
  ...
  "inline_images": "Array of strings - List of inline image URLs",
  "quick_results": [
    {
      "title": "String - Title of the quick result",
      "link": "String - URL to the quick result",
      "snippet": "String - Snippet of the quick result",
      "source": "String - Source of the quick result",
      "displayed_link": "String - Displayed link for the quick result",
      "favicon": "String - URL to the favicon of the source"
    },
    ...
  ],
  "text_blocks": [
    {
      "type": "String - Type of the text block. Can be 'heading', 'paragraph', 'list', 'expandable', 'comparison', 'code_block',
      "snippet": "String - Snippet of the text block",
      "snippet_highlighted_words": "Array of strings - Highlighted words in the snippet",
      "snippet_inline_code": "Array of strings - Inline code within the snippet",
      "snippet_links": "Array of objects - Links within the snippet",
      "reference_indexes": "Array of integers - Indexes of the references in the root 'references' field",
      // Only for 'list' type
      "list": [
        {
          "title": "String - Title of the list item",
          "snippet": "String - Snippet of the list item",
          "reference_indexes": "Array of integers - Indexes of the references in the root 'references' field",
          "thumbnail": "String - URL to the thumbnail image",
          // Nested lists
          "list": [
            {
              "snippet": "String - Snippet of the nested list item",
              "code_block": {
                "language": "String - Programming language of the code block",
                "code": "String - Actual code content"
              },
              "reference_indexes": "Array of integers - Indexes of the references in the root 'references' field",
            },
            ...
          ],
          ...
        ],
        ...
      ],
      // Nested text blocks
      "text_blocks": [
        // The same structure as the parent 'text_blocks' field
      ],
      ...
    ],
    ...
  ],
  // Only for `table` type
  "table": [
    [
      "String - Table cell",
      ...
    ],
    ...
  ],
  ...
  "formatted": "Array or Object, depending on the structure of the table - Formatted table data",
  // Only for `code_block` type
  "language": "String - Programming language of the code block",
  "code": "String - Actual code content"
},
  ...
],
  "references": [
    {
      "title": "String - Title of the reference",
      "link": "String - URL to the reference",
      "snippet": "String - Snippet of the reference",
      "source": "String - Source of the reference",
      "index": "Integer - Index of the reference"
    },
    ...
  ],
  "shopping_results": [
    {
      "title": "String - Title of the product",
      "product_link": "String - URL to the product",
      "immersive_product_page_token": "String - Product token used with Google Immersive Product API",
      "serpapi_impressive_product_api": "String - SerpApi Google Immersive Product URL for the selected product",
      "thumbnail": "String - URL to the thumbnail image",
      "price": "String - Price of the product",
      "extracted_price": "Numeric - Product price as a float or integer",
      "old_price": "String - Product's price before discount (if applicable)",
      "extracted_old_price": "Numeric - Product old price as a float or integer",
      "rating": "Numeric - Product rating",
      "reviews": "Integer - Product review count"
    },
    ...
  ],
  "local_results": [
    {
      "title": "String - Title of the place",
      "link": "String - URL to the place",
      "thumbnail": "String - URL to the thumbnail image",
      "rating": "Numeric - Place rating",
      "reviews": "Integer - Place review count",
      "price": "String - Price range of the place",
      "hours": "String - Opening hours of the place",
      "type": "String - Type of the place (e.g., restaurant, park, museum)",
      "address": "String - Address of the place"
    },
    ...
  ],
  "inline_videos": [
    {
      "title": "String - Title of the video",
      "link": "String - URL to the video",
      "thumbnail": "String - URL to the thumbnail image",
      "duration": "String - Duration of the video",
      "channel": "String - Channel name",
      "platform": "String - Platform (e.g., YouTube, Vimeo)"
    },
    ...
  ],
  ...
],
  "error": "String - Error message if the AI Mode results are not available",
}
```



/

API DASHBOARD

[Your Account](#)[Your API Metrics](#)[Your Searches](#)[Your Playground](#)[Api Status](#) · [GitHub](#) · [Legal](#) · [Security](#) · [Libraries](#) · [Release Notes](#) · [Public Roadmap](#) · [Support](#)

© 2016-2025 SerpApi, LLC.

API DOCUMENTATION

 [Google Search API](#) [Google Light Search API](#) [Google Ads Transparency API](#) [Google AI Mode API](#) [Google AI Overview API](#) [Google Autocomplete API](#) [Google Events API](#) [Google Forums API](#) [Google Finance API](#) [Google Flights API](#) [Google Hotels API](#) [Google Images API](#) [Google Images Light API](#) [Google Immersive Product API](#) [Google Jobs API](#) [Google Lens API](#) [Google Local API](#)

Google Local Services API

 [Google Maps API](#) [Google Maps Reviews API](#) [Google News API](#) [Google News Light API](#) [Google Patents API](#) [Google Play Store API](#) [Google Related Questions API](#) [Google Reverse Image API](#) [Google Scholar API](#) [Google Shopping API](#) [Google Shopping Light API](#) [Google Light Fast API](#) [Google Short Videos API](#)

Google Travel Explore API

 [Google Trends API](#) [Google Videos API](#) [Google Videos Light API](#) [Amazon Search API](#) [Apple App Store API](#)

Baidu Search API

 [Bing Images API](#) [Bing Copilot API](#)

Bing Search API

 [DuckDuckGo Search API](#) [DuckDuckGo Light API](#)

eBay Search API

 [Facebook Profile API](#)

Naver Search API

The Home Depot Search API

Tripadvisor Search API

Walmart Search API

 [Yahoo! Search API](#) [Yandex Search API](#) [Yelp Search API](#) [YouTube Search API](#) [Extra APIs](#) [Status and Error Codes](#)