SERP Data Api Guide

DataForSEO is a great API to retrieve SERP data for cheap.

Here I show you how to get the most out of DataForSEO in this very simple guide.

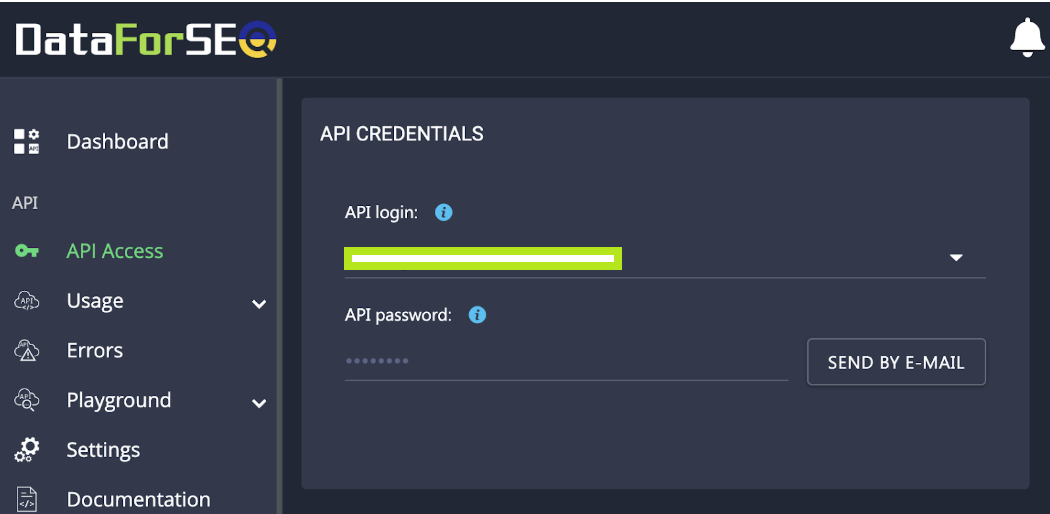
Setup and Requirements

1. Create a DataForSEO account at <https://dataforseo.com/>

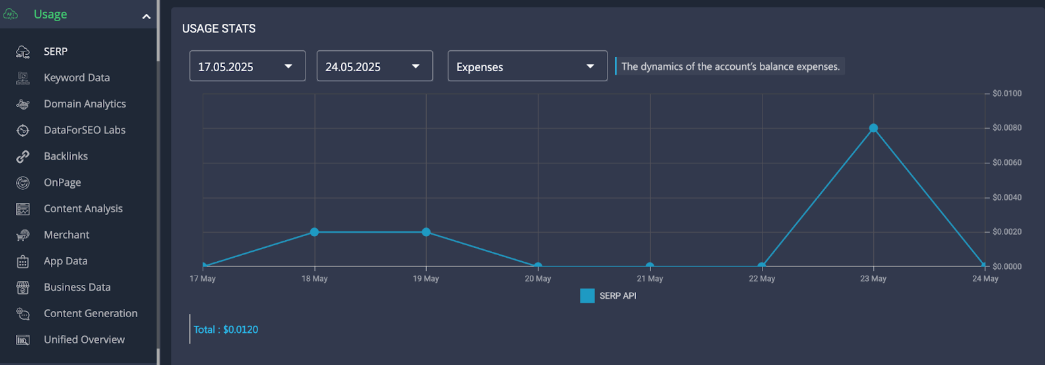
2. Navigate to "API Access" section

3. Request your API password via email (this is your key)

4. Store your credentials securely for API access

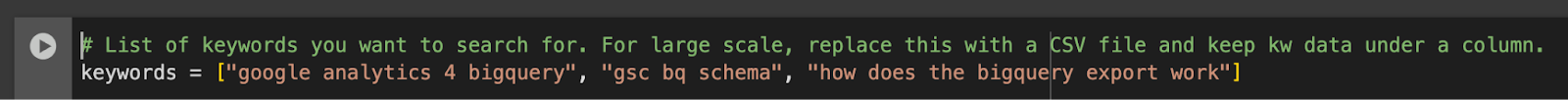


You can also go to "Usage" to see how your queries would look like and which parameters you can request.



If you struggle at any point, AI is your friend and does a great job at explaining you what’s going on!

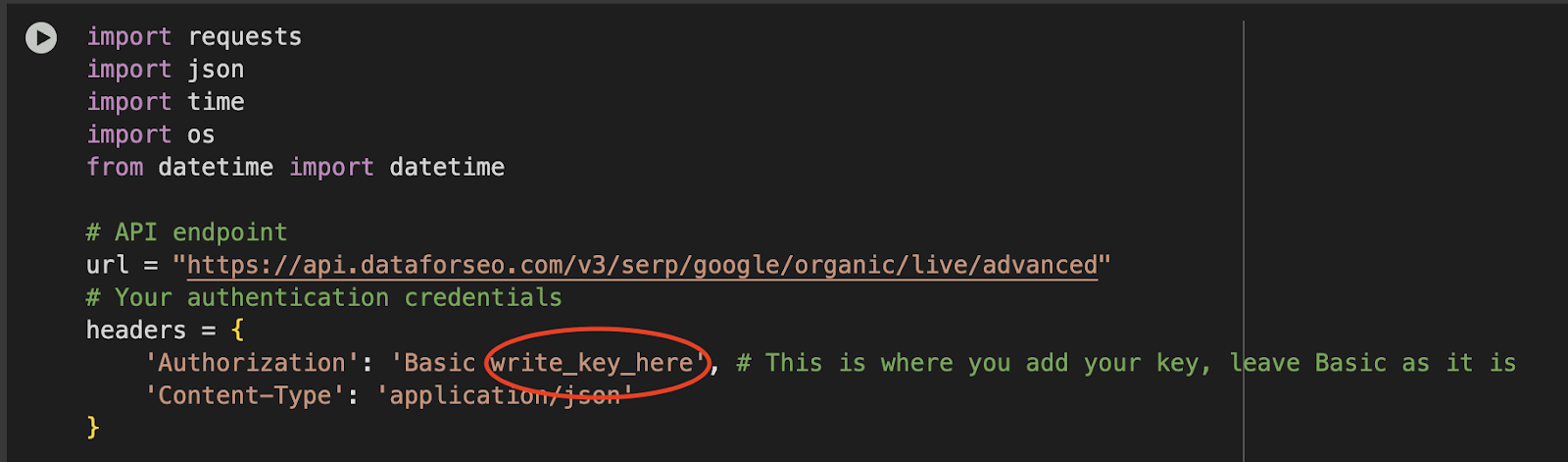
Since we are are scraping SERPs, the #1 thing to do is to specify which keywords you want, as I do below in this list:



The script makes a call for each keyword and then combines the results.

To make it work, the key we got before via email should be replaced in the red circle below (go to Code Example inside DataForSEO dashboard).

Remember, Basic should stay!



So add your key in the code above ^^^

*P.S. The code I shared saves the results inside a CSV file. In Google Colab, you can easily locate the file.*

N.B. You can also modify the **url** variable in the picture above to include a different endpoint (aka DataForSEO service).

The Playground option offers you the option to see the code needed for the task.

### **AI Overviews**

AI Overviews are identified in the API response through the `type` field. When analyzing results:

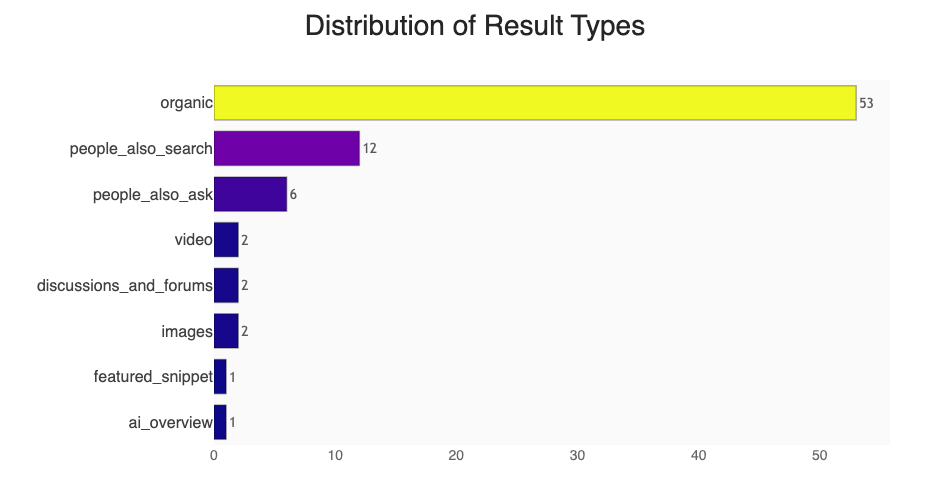
1. Each result is categorized by its "type"

2. "ai\_overview" type represents the AI-generated summary

3. Position and visibility can be analyzed relative to other result types

*P.S. In Europe, only logged-in users will get AI Overviews. This is quite different from the US where every user is eligible for AIO.*

In the code I show you, I showcase a simple visualization like the horizontal bar plot below:

**

You can use this to understand what are the most prominent search types in your niche and act accordingly.

### **Pixel Ranking**

Instead of using the average position provided in GSC or even checking the rank with DataForSEO, you could measure how many pixels a given URL is occupying.

This method is better for evaluating SERPs as you can tell what’s getting actually displayed.

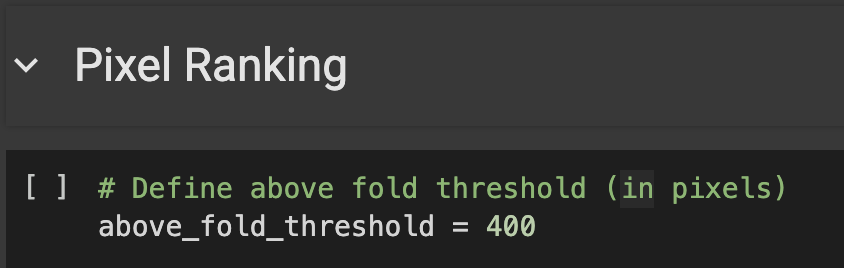
My advice is to have 2 groups of results:

* Above fold
* Below fold

**Desktop > 800 pixels**

**Mobile > 400 pixels**

In the code, I simply show how to do it, all you have to do is replace this bit:



Put whatever number of pixels you want.

Pixel ranking and AIO analysis can be integrated with some good ol’ CTR Analysis that is described in my FREE BigQuery Handbook.

🔗You can get it by subscribing to my newsletter in the welcome email: https://seotistics.com/best-seo-and-analytics-newsletter/

## MCP Connection and Examples

An MCP is an open standard to connect data sources/other tools and AI-powered tools, like Claude.

This means we can **connect Claude straight to DataForSEO without writing any Python code**.

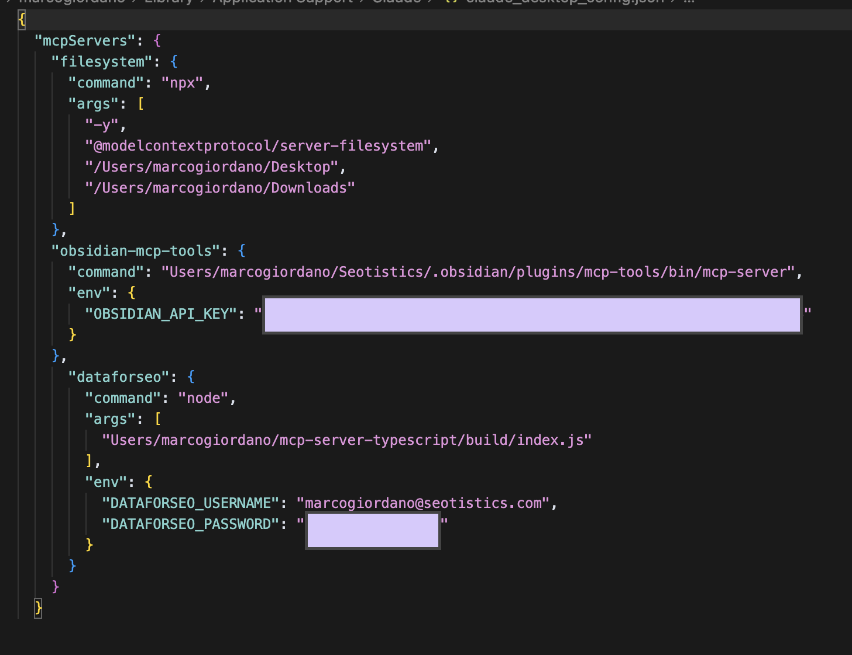
DataForSEO actually prepared a nice and clear guide for the topic: <https://dataforseo.com/help-center/setting-up-the-official-dataforseo-mcp-server-simple-guide>

—

If you already have other MCPs connected, then you may have trouble adding DataForSEO too.

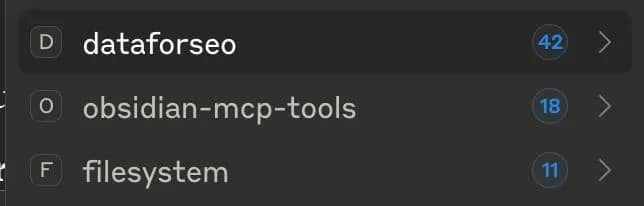
It’s a matter of syntax, be sure that the brackets are correct!

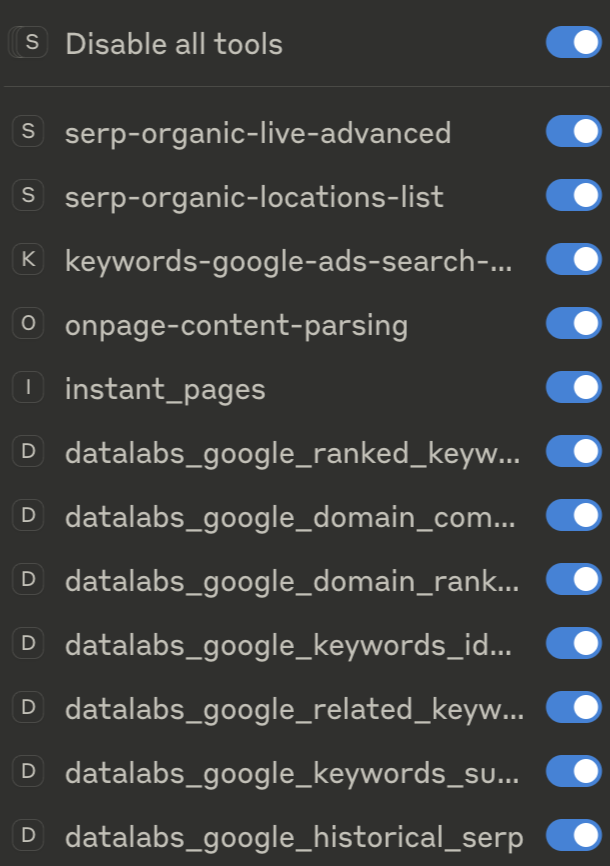
My configuration files is indeed confusing with all those brackets:



—

Once you manage to activate it, you should be able to see all of your DataForSEO options:





*P.S. Use the following commands as references and not as your final deliverable to the client, please.*

The process for using this integration is straightforward, you ask a request to Claude, for example one of the following questions:

### **SERP & Ranking Analysis**

* "What does the current SERP look like for 'best coffee makers 2025'?"
* "Show me the historical ranking changes for 'cryptocurrency news' over the past year"
* "Which domains dominate the first page for 'digital marketing services'?"
* "Analyze the SERP competitors for my target keyword list"

### **Competitor Analysis**

* "Who are the top SEO competitors for nike.com and what keywords do they rank for?"
* "Compare the backlink profiles of airbnb.com vs vrbo.com"
* "What keywords does my competitor rank for that I don't? (domain intersection analysis)"
* "Analyze the top pages and their SEO metrics for python.org"

### **Keyword Research & Analysis (Nice to have)**

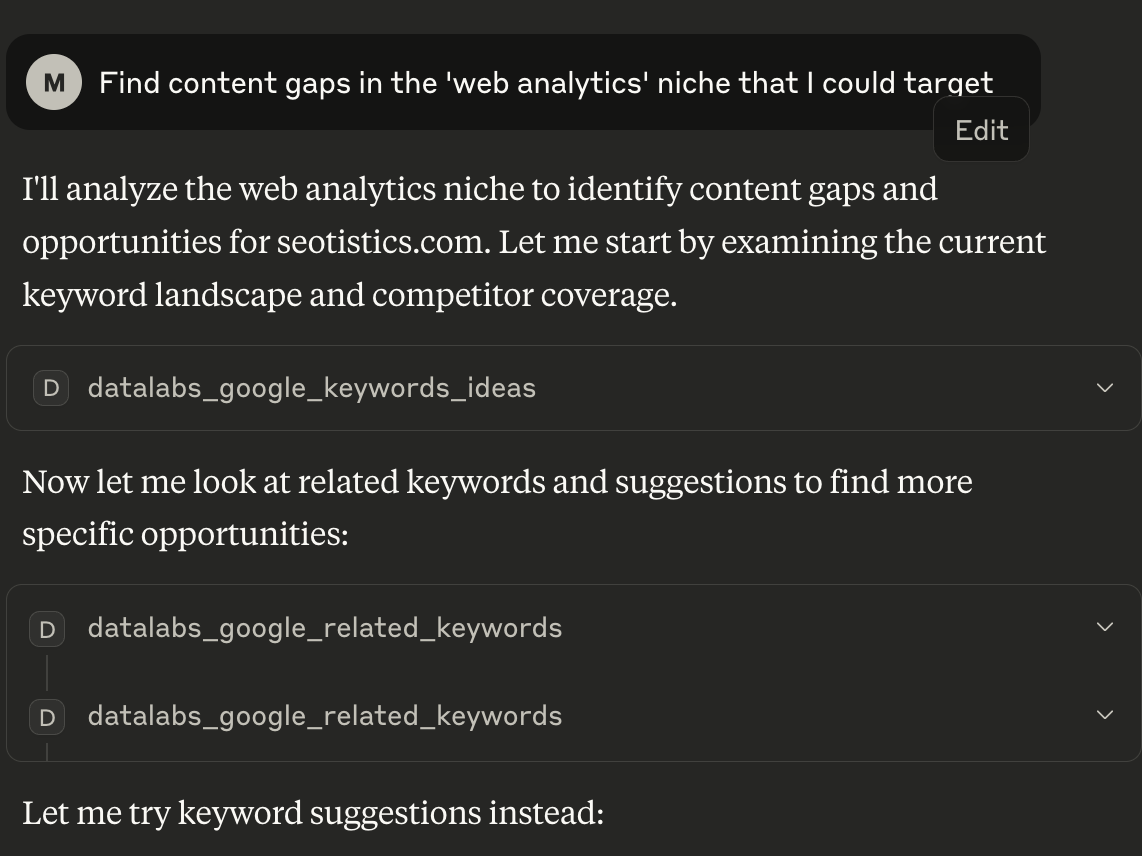
* "What are the top 50 keyword suggestions for 'sustainable fashion' with search volume data?"
* "Find me low-competition, high-volume keywords related to 'home workouts' that I could target"
* "Show me related keywords and search intent data for 'electric vehicles"

When using MCP, I recommend you actually check how much you can spend first (and avoid giant websites).

From what I tested, Claude can’t tell you the price per request and that’s exactly why you should trim down your output!

Here follows an example I just used after installing this MCP.

As you can see, it queries data for different offerings by DataForSEO (e.g. DataLabs) and only then elaborates an output.



## When To Choose Each

Both the API and the MCP approach have their pros and cons but luckily they don’t require extensive technical skills.

### Choose MCP Integration When:

* ✅ Need quick, ad-hoc analysis
* ✅ Non-technical team members involved
* ✅ Exploratory research and insights
* ✅ Client presentations and reports
* ✅ Limited technical resources

### Choose Python API When:

* ✅ Building automated systems
* ✅ Need custom data processing
* ✅ Large-scale, repeated operations
* ✅ Integration with existing systems
* ✅ Real-time applications
* ✅ Full control over costs and performance -> **important!!!**

**Optimal Method:**

1. **Use MCP for:** Initial research, ad-hoc analysis, client presentations
2. **Use Python API for:** Automated reporting, large-scale operations, system integration
3. **Transition path:** Start with MCP for exploration → Build Python automation for repeated tasks