Asset Generation • Part I

API useapi.net/docs/articles/asset-generation-part-1

Automating Asset Generation with the Midjourney API • Part I

3 min read • November 13, 2023 (August 15, 2025)

Table of contents

Introduction

If you are a designer, a game developer, or simply experimenting with [Midjourney](#) and want to create a large number of [/imagine](#) generations based on your ideas, it may be very tedious and time-consuming to do so manually.

In this article, we will explore methods to efficiently execute such tasks with minimal effort.

We will also attempt to measure the time it takes to generate 50 Midjourney /imagine generations and compare it with the time reported by Midjourney for their Fast GPU.

Please check a [YouTube video](#) covering this article.

Setup

We will use the Midjourney API provided by [useapi.net](#) to interact with Midjourney.

To get started, you'll need a Discord account with an active [Midjourney subscription](#). The $10 Basic Plan will suffice.

Follow these [simple steps](#) to obtain the following:

- Discord Direct Messages channel id number, referred to in this article as `channel`.

- Discord token, referred to in this article as `discord`. [Verify Discord access](#).
- Create Midjourney configuration using retrieved above `discord` token via [POST account/midjourney](#).
- [useapi.net](#) API token, referred to in this article as `token`. [Verify useapi.net API access](#).

useapi.net provides an easy way to experiment with all API endpoints without writing any code. Check the `Try It` section at the end of each document page, such as [jobs/imagine](#).

For your convenience, we have published all the [code](#) used in this article. You can choose between JavaScript and Python examples. Clone this repository locally and use it as a starting point for your experiments.

Preparing Midjourney Prompts

Let's use [ChatGPT](#) to create a list of prompts. For this article, we will be generating fairy tale characters. Here's our ChatGPT prompt:

> Create a JSON array of strings with 50 Midjourney prompts. Each prompt should describe a popular fairy tale protagonist or antagonist placed in the modern world, solving contemporary problems using fairy tale methods. Be creative!

The output will look something like this:

```
[
    "Cinderella, now a well-known fashion designer, uses her knack for transforming
rags into fabulous outfits to help those in poverty dress for success.",
    "Jack, of Jack and the Beanstalk fame, is a successful botanist who tackles the
issue of world hunger by growing gigantic, genetically modified crops.",
    "Rapunzel, a professional rock climber, uses her long-braided hair for safety
ropes during rescue missions in difficult terrains.",
    ...
    "The Big Bad Wolf, a top-notch security consultant, fortifies homes against
break-ins, teaching the importance of strong construction materials."
]
```

An array of prompts generated by ChatGPT needs to be saved to a locally cloned [prompts.json](#) file.

Executing Prompts Using the Midjourney API

Create a file locally in the same folder named `example.sh` with the following content:

[JavaScript](#)

```
USEAPI_TOKEN="useapi API token" USEAPI_CHANNEL="Discord Direct Messages channel" node
./example.js
```

[Python](#)

```
USEAPI_TOKEN="useapi API token" USEAPI_CHANNEL="Discord Direct Messages channel"
python3 ./example.py
```

Execute it from the command line like this: `./example.sh` and observe the magic of the API. We have created a [YouTube video](#) covering this entire process.

The generated images will be saved locally. You can continue the generation process in a Discord Direct Messages channel to refine your desired creations, or you can use the [jobs/button](#) to automate it, following the same process as demonstrated above.

Conclusion

As shown in the [YouTube video](#), you can observe the following:

- Time to generate 50 Midjourney /imagine creations using the Midjourney API: **~19** minutes
- Utilization of the Midjourney Fast GPU: **~26** minutes

Visit our [Discord Server](#) for any support questions and concerns.

We regularly post guides and tutorials on the [YouTube Channel](#).

Cross posted

- [hashnode.dev](#)
- [dev.to](#)

- medium.com