



---

Developer Advocate

## Coding Challenge

The Marvel Comics API allows developers to access information about Marvel's vast library of comics.

We want you to use the Marvel API (see <https://developer.marvel.com/>) to retrieve data.

### Part One

1. Write a command line program that accepts the argument “characters” and prints out on the console a JSON object with the following data:
  - a. The top 10 Marvel character IDs in an array of integers. A top character is the one with the most number of appearances on comics and stories;
  - b. Some data statistics including: the total number of characters; the total number of characters appearing in comics; and the total number of characters appearing in stories;
  - c. Because the Marvel API only returns max 100 records per request, you need to execute all the necessary requests in your application and return a single response;
  - d. Add also in your response object the Marvel API response metrics: total number of requests; the total and the average response time in milliseconds; and total responses byte size;
  - e. The command line execution and response should be something like this:

```
$ marvel characters

{
  "characters": {
    "ids": [ 1009718, 1017100, 1009144, 1010699, 1016823, 1009148, ... ],
    "total": 1491,
    "total_in_comics": ...,
    "total_in_stories": ...
  },
  "requests": {
    "total": 14,
    "total_response_time": 4352,
    "average_response_time": 732,
    "total_response_size": 825245,
    "average_response_size": 82314
  }
}
```

2. Extend your program to accept a character ID as argument and return only the following data of the character:
  - a. Character id, name, description and full thumbnail URL;
  - b. Include also the corresponding requests metrics;

```
$ marvel characters 1009718

{
  "character": {
    "id": 1009718,
    "name": "Wolverine",
    "description": "Born with super-human senses and the power to heal from almost any wound, Wolverine was captured by a secret Canadian organization and given an unbreakable skeleton and claws. Treated like an animal, it took years for him to control himself. Now, he's a premiere member of both the X-Men and the Avengers.",
    "thumbnail": "http://i.annihil.us/u/prod/marvel/i/mg/2/60/537bcaef0f6cf.jpg"
  },
  "requests": {
    "total": 1,
    "total_response_time": 352,
    "average_response_time": 352,
    "total_response_size": 10145,
    "average_response_size": 10145
  }
}
```

You'll need to sign up for Marvel developer API key at <http://developer.marvel.com> (free)

Once you have a key then the API documentation is at <http://developer.marvel.com/docs>

## Part Two

Lets enrich the characters' responses with more value added from other external sources.

3. Enable your program to include the character's powers:  
**\$ marvel powers {characterId}**
  - a. The Marvel API does not give back the character's powers directly. You will need to discover and use the character's Wiki URL and from there extract the powers' text out of the HTML content.
  - b. You decide the response format.
4. Extend the powers' response to return a translated version to another language. Accept an additional argument with the language ISO-639-1 code:  
**\$ marvel powers {characterId} {languageCode}**
  - a. Use the Google Translate API to translate in real-time the original Powers text;

## Constraints

- We're looking for a program that can be executable from the command line (Windows, Mac or Linux)
- Use of third party libraries is ok; in the coding interview we'll be asking you about your choices
- The API keys / secrets should not be stored with the code.  
See here: <https://support.google.com/cloud/answer/6310037>

## On Completion

Write a markdown README.md with instructions on how to:

- Install any dependencies, files or environment variables that your code requires;
- How to build and run your program.

Send your source code in a compressed zip file to: [interview@yapily.com](mailto:interview@yapily.com)