What is an API?

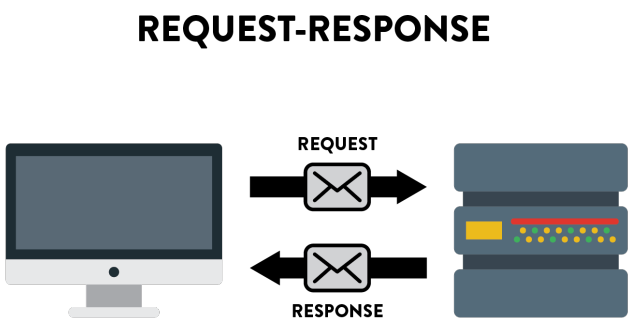
API stands for Application Programming Interface. It has different meanings in different contexts, but here it is used as: a system for computer programs to communicate over a network, such as the internet, usually to exchange information. These are also called **Web Services**.

There are many services on the internet that provide data via APIs. Some are free and some are paid. When you're working at a company, they may have their own internal APIs to provide data from one system to another. You may be the one writing the code for the API!

API Basics

Communicating with an API involves two players–the client and the server–and two messages–the request and the response:

1. The **client** starts the communication by sending a **request** to the server.
2. Then the **server** sends back a **response**.



The request

The request message involves several pieces of information:

1. **URL**: This identifies which server and service to talk to and what resources are of interest. e.g. `https://api.example.com/v2/inventory?q=shoes&pageSize=20`. The URL itself contains several elements:
    a. The host/domain identifies the server. e.g. api.example.com
    b. The path identifies the resources. e.g. /v2/inventory
    c. Sometimes query string parameters provide extra options. e.g. ?q=shoes&pageSize=20
2. **HTTP method**: This identifies the action to perform. There are several options including these five very common ones:
    a. GET: Get information from the server.
    b. POST: Add new information to the server.
    c. PUT: Update information by replacing the previous values.
    d. PATCH: Update information by modifying the previous values.
    e. DELETE: Delete information from the server.
3. **Request body**: Some requests require structured data to be sent to the server. This is typically required for a POST, PUT, or PATCH. This data is commonly formatted as JSON (see below).
4. **Headers**: These are extra variables sent with the request as needed. This may include authentication keys.

The response

The response message also contains several pieces of information:

1. **Status Code**: Was the server able to do what the client wanted? Did anything go wrong? There is a predefined set of status code numbers to communicate what happened. Here are some common ones:
    a. 200 Okay/success
    b. 400 Error: bad request
    c. 401 Error: requires authentication
    d. 404 Error: not found
    e. 500 Server-side error
2. **Response Body**: Most responses do send back structured data. JSON is the most common format.
3. **Headers**: The response can also include its own set of extra variables.

JSON Format

JSON stands for JavaScript Object Notation. It is a way to put structured data into a text file. JSON is popular because it is simple, easy for humans and computers to read, and it translates well into data structures in most languages.

The format is very similar to how objects, arrays, strings, numbers, and booleans are written in JavaScript. Here's an example:

```
[
  {
    "firstName": "Blaise",
    "lastName": "Pascal",
    "innovation": "invented the mechanical calculator",
    "year": 1645,
    "english": false
  },
  {
    "firstName": "Ada",
    "lastName": "Lovelace",
    "innovation": "published the first computer programs",
    "year": 1843,
    "english": true
  },
  {
    "firstName": "Charles",
    "lastName": "Babbage",
    "innovation": "designed first mechanical computer",
    "year": 1837,
    "english": true
  }
]
```

Testing an API

Before writing code to access an API, it's a good idea to test it out manually and get a look at the different URLs and data that is available.

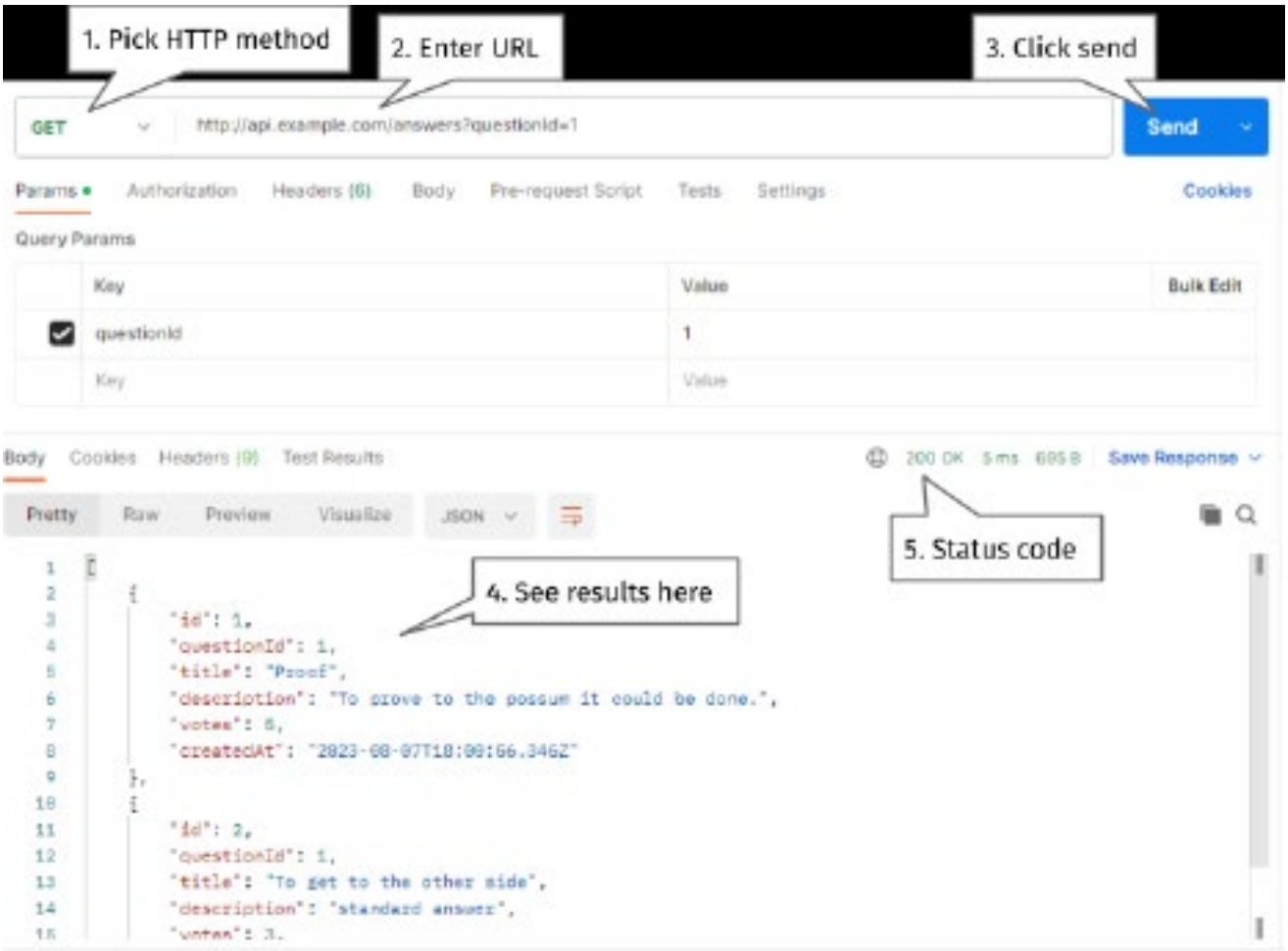
To do this you'll want to use an API client. There are many options out there. We recommend Postman

If you're only making simple GET requests, you can also just put the URL into your web browser. Although, the response data may be hard to read unless you install a JSON formatter plugin such as JSON Formatter for Chrome.

Many APIs will require you to sign up for an account. They will then typically provide you with some kind of API key. Follow the directions they give you to include it as a URL parameter or request header.

Here are a couple screenshots to demonstrate how to use Postman.

Example GET Request



Example POST request