Java Cat Fact Ninja Web API HashMap Tutorial

Contents

Time required: 60 minutes

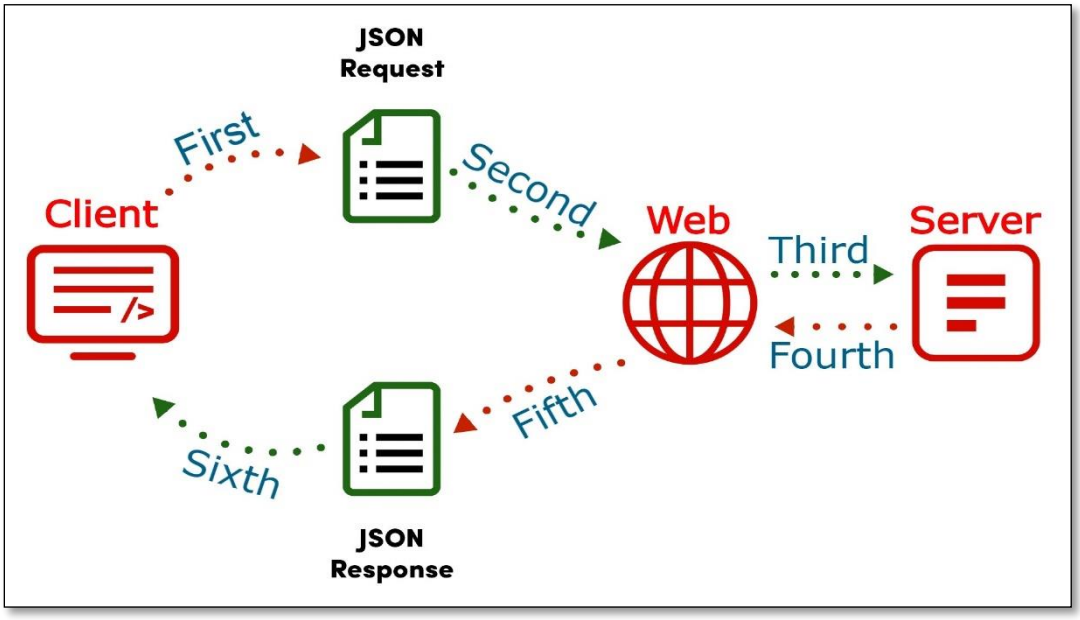
- Comment each line of code as shown in the tutorials and other code examples.
- Follow all directions carefully and accurately.
- Think of the directions as minimum requirements.
- Please read all the directions before beginning the assignment.

What is an API?

What is an API? This is what Wikipedia says.

"An **application programming interface** ('API') is a computing interface that defines interactions between multiple software intermediaries. It defines the kinds of calls or requests that can be made, how to make them, the data formats that should be used, the conventions to follow, etc. It can also provide extension mechanisms so that users can extend existing functionality in various ways and to varying degrees. An API can be entirely custom, specific to a component, or designed based on an industry-standard to ensure interoperability. Through information hiding, APIs enable modular programming, allowing users to use the interface independently of the implementation."

An API is a software intermediary that allows two applications to talk to each other. In other words, an API is the messenger that delivers your request to the provider that you're requesting it from and then delivers the response back to you.



Web API's are hosted on webservers. Instead of a web browser asking for a webpage (like how most interactions with the internet is done today) your program asks for data. This data is usually returned to your program in a JSON format. JSON stands for JavaScript Object Notation and is a lightweight text-based data-interchange format.

To get data to our program we make a request to a web server. The webserver then replies with our desired data and a status code. Sometimes an API key is needed to be sent along with the request to successfully return data.

There are many ways to call information from an API. One of the most common are **get** requests. You use a get request each time you browse to a web site.

For example, to request data in a Python script using the extra functionality gained from the request library the statement below would be used.

```
response = requests.get('http://api_example.com/example.json')
```

This data object can be named anything, response was used as the identifier for this example. This API doesn't exist. This is the format they tend to take.

Data from the web can tell us the exact weather situation of any place on the globe, the location of faces on an image we provide or the number of astronauts currently floating far,

far out in space. Any information freely accessible on the World Wide Web is literally within our grasp from within our programs.

Why Use a Web API?

You could use static data which you have downloaded or created yourself. Data used this way with Java would work perfectly fine. Here are some examples of when APIs are more useful than static data sets.

- When the data is changing quickly, and you need the data to be accurate and up to date. Planes are currently in the air or the location of the international space station, the data is constantly changing.
- When only a small piece of a much larger data set is desired.
- When you are taking advantage of already completed, repeated computations. Sources such Facebook, Spotify and Google already have a ton of data, which have undergone serious calculations, which you can access easily with an API request.

In cases like the ones above, a Web API is the right solution. Using an API will save time and effort over doing all the computation ourselves or constantly refreshing our desired data.

Public API's

There are hundreds of public API's available. This website contains a list of some of them.

https://github.com/public-apis/public-apis

Cat Facts Ninja API

Who doesn't want to know random facts about cats?

We do! Let's create a Java program to help us with our cat education.

Before we do that, we have a few new concepts to go through.

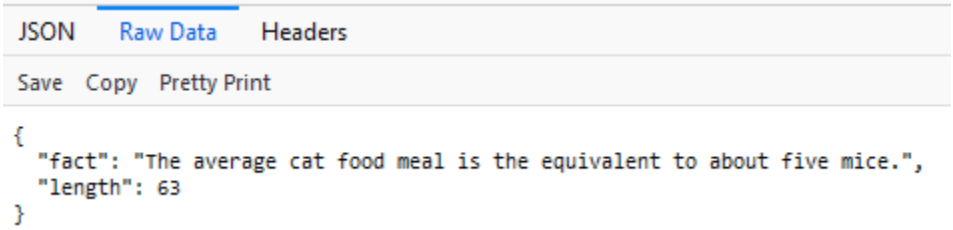
What is JSON?

The Cat Facts Ninja API is a free https-based Web API which provides a JSON response.

Raw API data is typically in a text file in JSON or XML format. JSON is the most popular format for web API's.

JavaScript Object Notation (JSON) is an open standard text-based file and data interchange format. It uses human-readable text to store and transmit data objects consisting of attribute–value pairs and array data types.

Go to https://catfact.ninja/fact to see some raw JSON data. This view is using Firefox.



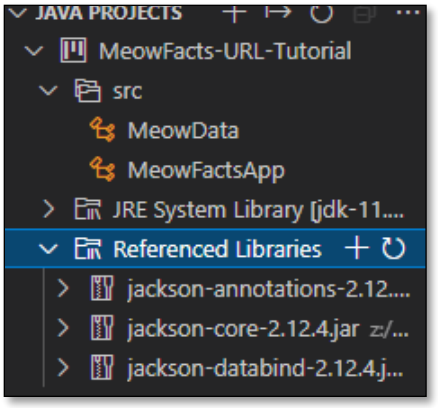
Jackson JSON Libraries

This will be a Visual Studio Code project in Java.

1. Create a folder to hold your project: **CatFactNinja**
2. Create two folders in your **CatFactNinja** folder.
    a. **lib**
    b. **src**

We will be retrieving data from the web in JSON format. Java doesn't have any native JSON parsing. We are going to add three libraries to parse out the JSON data into Java classes.

3. Download these three files.
    a. jackson-annotations-2.16.1.jar
    b. jackson-core-2.16.1.jar
    c. jackson-databind-2.16.1.jar
4. Copy and paste the three jar files into the **lib** folder.
5. In the **src** folder, create **CatFactNinja.java**
6. Double Click **CatFactNinja.java**
7. You should see the following at the bottom left if you open **Java Projects → Referenced Libraries**



Cat Fact Ninja Java App

Add the following code to the **CatFactNinja.java** file.

```java
/**
 * Filename: CatFactNinja.java
 * Written by: William A Loring
 * Written on: 08/02/21
 * Revised:
 * Description: Get Cat Fact Ninja Facts from API endpoint
 */

// Handle any Input Output exceptions
import java.io.IOException;
// Create a URL object
import java.net.URL;
import java.util.Scanner;
import java.util.HashMap;
import java.util.Map;

// Import Jackson JSON libraries
import com.fasterxml.jackson.core.JsonGenerationException;
import com.fasterxml.jackson.core.type.TypeReference;
import com.fasterxml.jackson.databind.JsonMappingException;
import com.fasterxml.jackson.databind.ObjectMapper;

public class CatFactNinja {
    private static Scanner keyboard = new Scanner(System.in);
    // API endpoint URL
    private final static String URL = "https://catfact.ninja/fact";
    // Create CatData HashMap object
    static HashMap<String, Object> dataHashMap = new HashMap<String, Object>();
    private static String menu = "y";

    public static void main(String[] args) {
        while (!menu.equals("n")) {
            getData();
            // Display a Fact at the console
            // For demo purposes, print out entire HashMap
            // System.out.println(dataHashMap.toString());
            // Display a single Cat Fact
            System.out.println(dataHashMap.get("fact"));
            System.out.print("Another Cat Fact (y, n): ");
            menu = keyboard.nextLine();
        }
        System.out.println("Thanks for using Cat Fact Ninja.");
    }

```

```
45      // Get data from the API endpoint
46      public static void getData() {
47          try {
48              // Create URL object
49              URL url = new URL(URL);
50
51              // Create Jackson JSON object to read the raw JSON data
52              // into POJO (Plain Old Java Objects)
53              ObjectMapper objectMapper = new ObjectMapper();
54
55              // Read POJO into MeowData HashMap Object
56              dataHashMap = (HashMap<String, Object>) objectMapper.readValue(
57                      url,
58                      new TypeReference<Map<String, Object>>() {
59                      });
60
61          } catch (JsonGenerationException | JsonMappingException e) {
62              // JSON Exception handling
63              // e.printStackTrace() will print the exception
64              e.printStackTrace();
65              // Input Output Exception handling
66          } catch (IOException e) {
67              // Handle any other exceptions
68              // e.printStackTrace() will print the exception
69              e.printStackTrace();
70          } catch (Exception e) {
71              // e.printStackTrace() will print the exception
72              e.printStackTrace();
73          }
74      }
75  }
```

The above code does 2 things.

1. Creates a Jackson object to read JSON into a HashMap.
2. Displays a Cat Fact.

Time for a Cat Ninja Fact

Time to run your new application.

Example run:

```
A cat's appetite is the barometer of its health. Any cat that does not eat or drink for more than
 two days should be taken to a vet.
Another Cat Fact (y, n): y
A female cat will be pregnant for approximately 9 weeks or between 62 and 65 days from conception
 to delivery.
Another Cat Fact (y, n): y
A cat cannot see directly under its nose.
Another Cat Fact (y, n):
```

Assignment Submission

1. Insert a screenshot of a successful run of the program.
2. Zip up the CatFactNinja folder.
3. Attach the zip file to the assignment in Blackboard.