

# Creating a Simple JSON API with PHP

In this tutorial, you will create a simple API using PHP.

The API will return JSON data when visited in a browser.

This is the same type of data format used by many real-world APIs.

---

## Project Structure

Create a folder called:

```
api-demo
```

Inside the folder place:

```
api-demo
├── api.php
└── smart_security_data.json
```

## Create the API File

Create:

```
api.php
```

Add:

```
<?php

header(
```

```
        "Content-Type: application/json"
    );

    echo file_get_contents(
        "smart_security_data.json"
    );

?>
```

Save the file.

---

## Understanding the Code

This line tells the browser that JSON data is being returned:

```
header(
    "Content-Type: application/json"
);
```

This line reads the JSON file:

```
file_get_contents(
    "smart_security_data.json"
);
```

This line sends the JSON data to the browser:

```
echo file_get_contents(
    "smart_security_data.json"
);
```

---

## Test the API

Open:

```
http://localhost/api-demo/api.php
```

You should see:

```
{  
  "homeId": "GC-HOME-014",  
  "location": "Gold Coast"  
}
```

along with the rest of the JSON data.

## Test the API in a New Browser Tab

Notice that the URL now behaves like a data source.

Instead of displaying a webpage, it returns structured JSON data.

This is the same concept used by many modern APIs.

Example:

```
https://example.com/api/users  
https://example.com/api/products  
https://example.com/api/weather
```

---

## Complete API File

```
<?php  
  
header(  
    "Content-Type: application/json"  
);  
  
echo file_get_contents(  
    "smart_security_data.json"  
);  
  
?>
```

You have successfully created a JSON API using PHP.

---

Revision #2

Created 2026-06-10 02:01:29 UTC by Mr Napper

Updated 2026-08-19 23:05:16 UTC by Mr Napper