Identifying the API type

Easy Way:

1. Check the Data Format

- JSON → REST
- XML → SOAP

2. Look at HTTP Methods

- REST → Uses `GET`, `POST`, `PUT`, `DELETE`.
- SOAP → Always uses `POST`.

3. Search for WSDL Reference

- WSDL present → SOAP
- No WSDL → REST

4. Look for XML Handling Code

- XML parsing (e.g., `SimpleXMLElement`) → SOAP
- JSON encoding/decoding → REST

This way, you can quickly identify the API type just by looking at the code.

OR

Here’s a **simple way to tell** if it's REST or SOAP just by looking at the **code**:

1. **Check the Data Format**

- **REST**: The code will likely handle **JSON** data like this:

  php

  Copy code

  ```
  header('Content-Type: application/json');
  echo json_encode($data);
  ```

- **SOAP**: The code will generate or parse **XML** like this:

  ```
  $xml = "<soap:Envelope>...</soap:Envelope>";
  echo $xml;
  ```

2. **Look at HTTP Methods**

- **REST**: You will see different HTTP methods like `GET`, `POST`, `PUT`, or `DELETE` used:

  ```
  if ($_SERVER['REQUEST_METHOD'] === 'POST') {
      // Handle POST request
  }
  ```

- **SOAP**: SOAP only uses **POST** for all operations, and the request will be wrapped in an XML envelope.

3. **Look for WSDL Reference**

- **SOAP**: You’ll see **WSDL URLs** or something like this in the code:

  ```
  $client = new SoapClient('http://example.com/service.wsdl');
  ```

- **REST**: There won’t be any WSDL or XML-based structure. It focuses more on JSON and simpler URLs.

4. **Look for XML Handling**

- **SOAP**: You’ll see lots of XML parsing and building:

  ```
  $xml = new SimpleXMLElement($response);
  ```

- **REST**: You won’t find heavy XML handling—only JSON encoding/decoding:

  ```
  $data = json_decode($jsonString, true);
  ```

Summary

- **REST**: Uses `json_encode`, HTTP methods like `GET`/`POST`, no WSDL.
- **SOAP**: Uses `SoapClient`, WSDL references, and **XML** everywhere.

In your case, if the PHP code mainly uses **JSON**, handles requests with `$_SERVER['REQUEST_METHOD']`, and has no WSDL references, it’s a **REST API**.

Last way

You can determine whether the API you are working with is **REST** or **SOAP** by examining several key characteristics. Here's how you can tell the difference:

1. **Check the Protocol**

- **REST**: Typically uses **HTTP/HTTPS**.
  Example:

  ```
  GET http://localhost/my-api/api/index.php
  POST http://localhost/my-api/api/index.php
  ```

- **SOAP**: Uses **HTTP** but wraps the data in **XML** envelopes with specific headers. SOAP also supports other protocols like SMTP.

2. **Look at the Data Format**

- **REST**: Data is often exchanged in **JSON** or **XML** formats.
  Example (JSON response):

  json

  Copy code

  ```
  {
      "id": 1,
      "title": "Book One",
      "author": "Author One"
  }
  ```

- **SOAP**: Always sends requests and responses in **XML** format.
  Example (SOAP XML response):

  ```
  xml

  <soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/
  ">
    <soap:Body>
      <GetBookResponse>
        <Book>
          <ID>1</ID>
          <Title>Book One</Title>
          <Author>Author One</Author>
        </Book>
      </GetBookResponse>
    </soap:Body>
  </soap:Envelope>
  ```

3. **Check the Request and Response Structure**

- **REST**:
    - Uses standard HTTP methods like `GET`, `POST`, `PUT`, and `DELETE`.
    - Data sent as query parameters (for `GET`) or JSON body (for `POST`).
- **SOAP**:
    - Uses **XML envelopes** to wrap the entire request/response.
    - Requires a **WSDL (Web Services Description Language)** file that describes the service's structure.

4. **Check HTTP Headers**

- **REST**:
    - Headers typically contain `Content-Type: application/json` or `application/xml`.
- **SOAP**:
    - Headers contain `Content-Type: text/xml` or `application/soap+xml`.

---

5. **Check the URL for WSDL File**

- If the API has a **WSDL** endpoint, it is a SOAP service. Example:

  `http://localhost/my-api/service.wsdl`
- If there's no WSDL, it's likely a REST API.

Conclusion

Since your PHP example is working with **`index.php`** and using JSON to handle book data, it is highly likely a **REST API**. SOAP APIs are more complex and rely on XML-based messages and WSDL files, which are not part of your current setup.