Streamline with references

In the previous topic, you completed your simple OpenAPI Specification. While this file only had one operation and one response, most OpenAPI Specifications contain more, and it is helpful to define the data structures for parameters and responses separately, so they can be used by more than one parameter or response.

Summary

Until now, you have written the object definitions for parameters and responses directly in the parameter or response object, by using the **schema** object. This means if you wanted to use the same object definition in several responses, you would need to copy and paste that definition into each response object.

References allow you to write the definition once in a special section of the OpenAPI Specification, and refer to it with a reference link every time you need it. It helps keep your OpenAPI Specification easy to read and maintain.

References and definitions in the OpenAPI Specification

Definitions in a OpenAPI Specification are stored in a special object that is usually at the end of the file, after the **paths** object. Each definition must have its own unique name within the OpenAPI Specification, and is referred to using the special **$ref** key, with the path to the definition as the value.

- Definitions for responses are stored in a **definitions** object.
- Definitions for parameters are stored in a **parameters** object.

  Note: This is not the parameters array that exists within each HTTP method object, but a separate parameters object that is placed after the **paths** object.

Create a definition and reference

To create a definition and reference, you declare:

- The name of the definition
- The properties it contains

Then add the reference path to the schema of the objects that should use the definition.

Step 1: Create the definitions object

After the **paths** object, create the **definitions** object to hold the response definitions.

Copy

```
"definitions": {

}
```

Creating a parameters object to hold parameter definitions

Step 2: Create a definition object

Inside the **definitions** object, create an object to define the airport information received from the Airport-Data API. The key of the object is the name you give the definition. It must be unique within your definitions object.

Copy

```
"definitions": {
  "airportInfo": {

  }
}
```

Creating more than one definition

Step 3: Define the object

Inside the definition object, add the data **type** of **object**, and a **properties** object to describe the information it contains. You can copy this directly from the original **schema** object in the response object.

Copy

```
"definitions": {
  "airportInfo": {
    "type": "object",
    "properties": {
      "name": {
        "type": "string"
      },
      "location": {
        "type": "string"
      },
      "country": {
        "type": "string"
      }
    }
  }
}
```

Step 4: Replace the original object with a reference

In the response object, replace the **schema** object **properties** with a key of **$ref**, and a value that is the path to the definition object.

Copy

```
"responses": {
  "200": {
    "description": "OK",
    "schema": {
      "$ref": "#/definitions/airportInfo"
    }
  }
}
```

Referencing a parameter definition

The OpenAPI Specification

The OpenAPI Specification is complete. You can import it for use in your workflows.

Copy

```
{
  "swagger": "2.0",
  "info": {
    "version": "1.0.0",
    "title": "Airport Data"
  },
  "host": "www.airport-data.com",
  "basePath": "/api",
  "schemes": [
    "http"
  ],
  "produces": [
    "application/json"
  ],
  "paths": {
    "/ap_info.json": {
      "get": {
        "summary": "Get airport info",
        "description": "Get airport info",
        "operationId": "getAirportInfo",
        "produces": [
          "application/json"
        ],
        "parameters": [
          {
            "name": "iata",
            "description": "Airport code",
            "in": "query",
            "type": "string"
          }
        ],
        "responses": {
          "200": {
            "description": "OK",
            "schema": {
              "$ref": "#/definitions/airportInfo"
            }
          }
        }
      }
    }
  },
  "definitions": {
    "airportInfo": {
      "type": "object",
      "properties": {
        "name": {
          "type": "string"
        },
        "location": {
          "type": "string"
        },
        "country": {
          "type": "string"
        }
      }
    }
  }
}
```

Next steps

Import the OpenAPI Specification