Exercise 3: Schemas

Now you'll add to the YAML file: a POST, PUT, and some responses.

You'll probably want to refer to the OAS file I created for the photo album example and used in the lesson you just watched. You can find it at http://sdkbridge.com/swagger/schemas.yaml.

Add a new schema

Open the Swagger editor at: http://editor2.swagger.io You should see the YAML from the last exercise, but if not, then you can import it from the file you saved.

You are going to add information about the request body for a POST request to create a new playlist, with a name and a list of IDs for each of the songs in the playlist. Here's an example of JSON in the request body:

```
{
    "name": "Mellow jazz",
    "songIds": [183, 13, 435, 98, 689]
}
```

Add a **definitions** key at the bottom of the file. Under that, add a **newPlayList** key (indented) and then add a string property for the name and an array of integers, which are the song IDs in the playlist.

```
definitions:
  newPlaylist:
    properties:
      name:
        type: string
      songIds:
        type: array
        items:
          type: integer
```

Now add a **required** key and make the name required, but leave the **songIds** as optional.

At this stage, you should see no errors, but you should see a little warning saying that the definition is not used. Remember, if you are seeing errors, try a refresh on the page.

```
65 definitions:
66   newPlaylist:
67     properties:
68       name:
69         type: string
70       songIds:
71         type: array
72         items:
73           type: string
```

Add a POST Request

Add a POST Request to create a new playlist. Here's a sample request

```
POST https://api.muzicplayz.com/v3/playlist

{
    "name": "Mellow jazz",
    "songIds": [183, 13, 435, 98, 689]
}
```

The URL is:

```
https://api.muzicplayz.com/v3/playlist
```

This means that you don't need a new path. You can add the request to the existing /`playlist` path.

1. Below the **get** section, add a similar section called **post**. (Same indentation.)
2. Add a **parameters** key, just like in the **get** section.
3. For **name**, use **newPlaylist**
4. For **in**, use **body**
5. For **required**, use **true**
6. For **schema**, refer to the **newPlaylist** object you've created in the **definitions** section.
7. Copy the simple **responses** section from **get** into **post:**

```
responses:
  # Response code
  200:
    description: Successful response
```

Notice that the warning has gone away, and also that your new POST request appears in the documentation on the left. There is a Schema section that will show your schema once you expand it all: the name as a string, and songIds as an array of integers.

Add a GET Response

Let's add GET request to the **/playlist/{playlist-id}** path that returns a playlist.

Here's a sample request:

```
GET https://api.muzicplayz.com/v3/playlist/playlist333
```

Here's a sample response:

```
{
   "id": "playlist333",
   "name": "Mellow jazz",
   "songs":
     [
        {"id": 183, "title": "String of Pearls"},
        {"id": 13, "title": "Stella by Starlight"},
        ...
     ]
}
```

Unlike the **newPlaylist** schema, for the **get**, we don't just want the song IDs. We want information about the songs in the playlist to be returned that we can display it. So we'll create a different schema for returning playlist info.

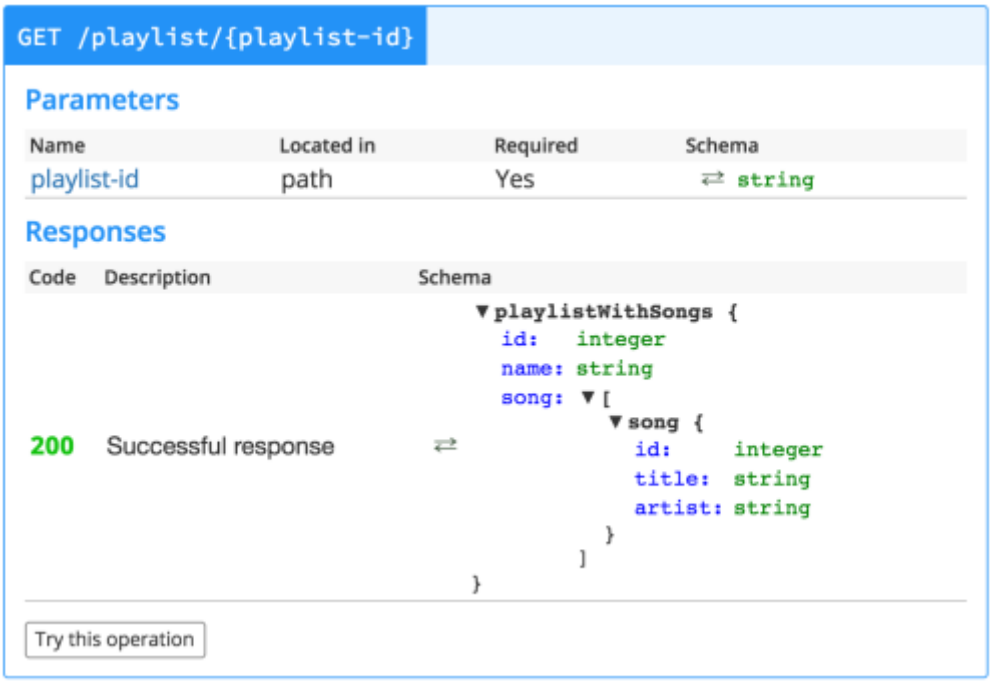
1. Copy the delete section and paste it under itself (including the **responses** section). It should have identical indentation.
2. Change **delete** to **get**
3. The path parameter is the same, so we don't need to modify that.
4. For the 200 response, add a reference to a schema called **playlistWithSongs** that we'll put in the **definitions** section.
5. At the bottom of the file, create a new section for **playlistWithSongs**. It should have the following properties (you won't use the description just yet):

Property	Type	Description
id	integer	ID of the playlist
name	string	Name of the playlist
songs	array	Array of type "song" object. Use the $ref key for this

6. You're going to need to add another section called **song**. It will have these properties:

Property	Type	Description
id	integer	ID of the song
title	string	Name of the song

On the right side, the documentation will look like this. You can see the response schema in the right column, and if you open it all out, you can see down to the song schema.



Save

It's a good idea to save your YAML file after each exercise. From the **File** menu, choose **Download YAML**. Save this somewhere where you can easily get to it for the next exercise.

Solution

If you get stuck, you can look at my version of the OAS file: http://sdkbridge.com/swagger/Exercise3Answer.yaml.