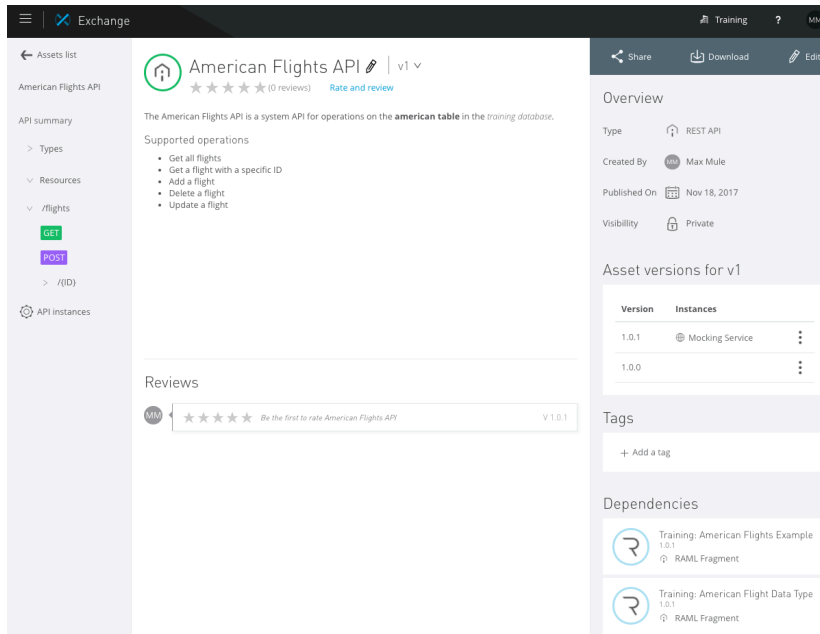


## Walkthrough 3-4: Add an API to Anypoint Exchange

In this walkthrough, you make an API discoverable by adding it to Anypoint Exchange. You will:

- Publish an API to Exchange from API designer.
- Review an auto-generated API portal in Exchange and test the API.
- Add information about an API to its API portal.
- Create and publish a new API version to Exchange.



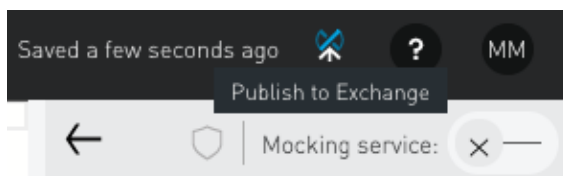
### Remove the baseUri by turning off the mocking service

1. Return to API designer.
2. Click the slider to turn off the mocking service; the RAML code should no longer have a baseUri.

```
1  #%RAML 1.0
2  title: American Flights API
3
```

### Publish the API to Anypoint Exchange from API designer

3. Click the Publish to Exchange button.



4. In the Publish API specification to Exchange dialog box, leave the default values:
  - Name: American Flights API
  - Main file: american-flights-api.raml
  - Asset version: 1.0.0
  - API version: v1
5. Click the Show advanced link.
6. Look at the ID values.
7. Click Publish.

### Publish API specification to Exchange

Name (required)	Asset version ⓘ (required)
<input type="text" value="American Flights API"/>	<input type="text" value="1.0.0"/>
Main file (required)	API version ⓘ (required)
<input type="text" value="american-flights-api.raml"/> ▼	<input type="text" value="v1"/>

Valid RAML

[Hide advanced ▼](#)

Group id  
800b1585-b6be-4c62-82de-02d8c2adf413

Asset id (required)

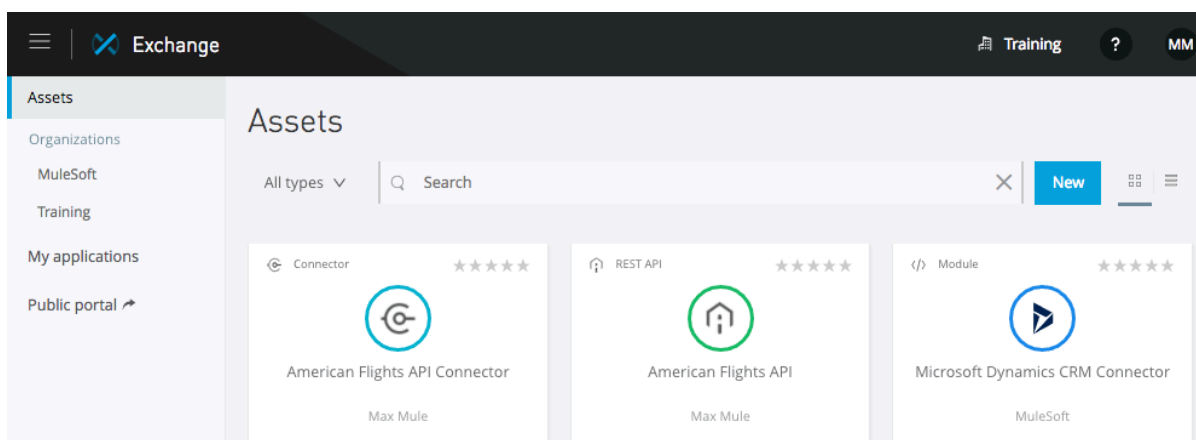
8. Click the Publish button.
9. In the Publish API specification to Exchange dialog box, click Done.
10. Look at the RAML file; you should see a version has been added.

```
1  #%RAML 1.0
2  version: v1
3  title: American Flights API
```

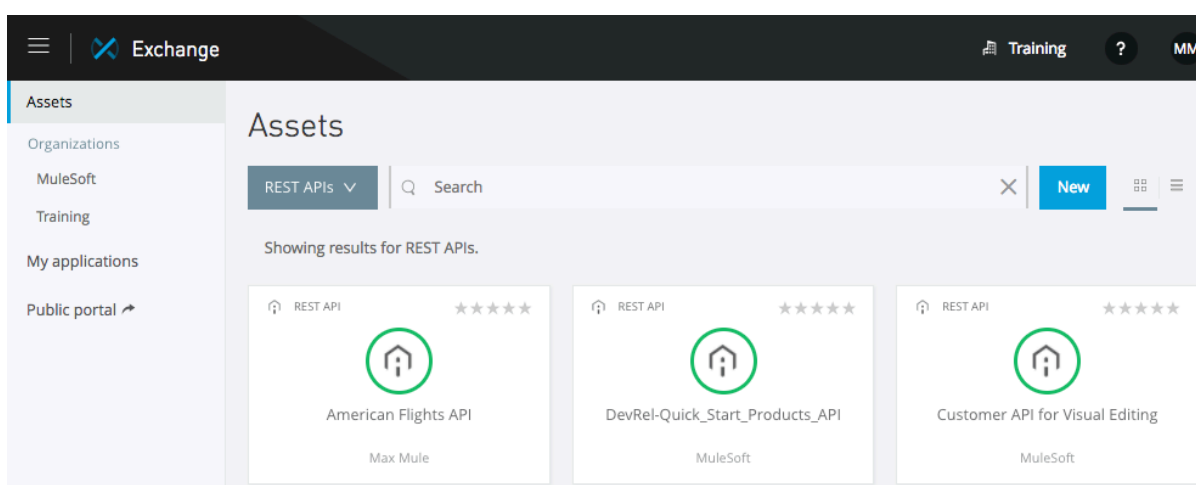
## Locate your API in Anypoint Exchange

11. Return to Design Center.

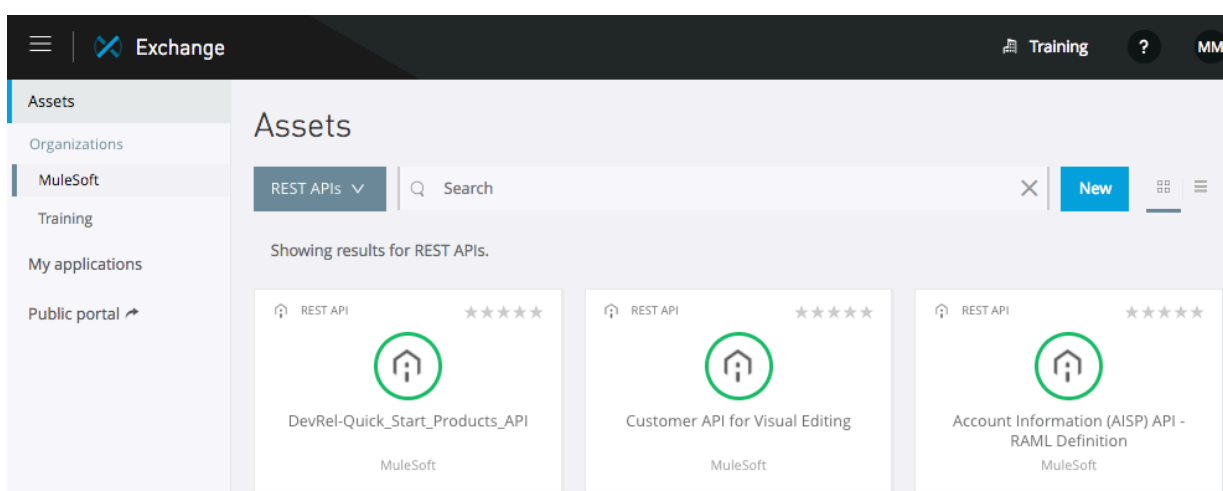
12. In the main menu, select Exchange; you should see your American Flights API.



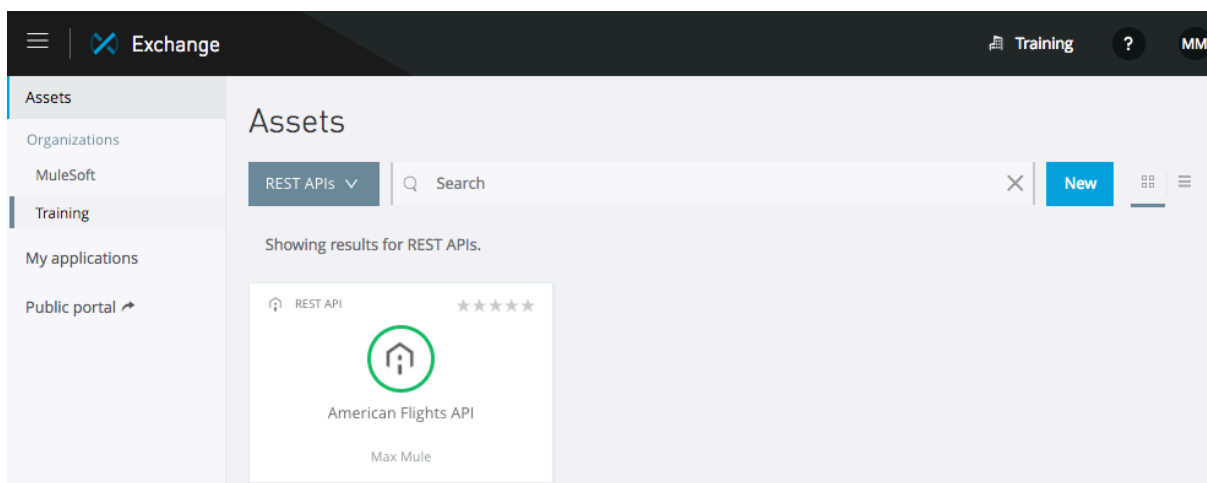
13. In the types drop-down menu, select REST APIs; you should still see your API.



14. In the left-side navigation, select MuleSoft; you should not see your American Flights API in the public Exchange (just the Training: American Flights API).

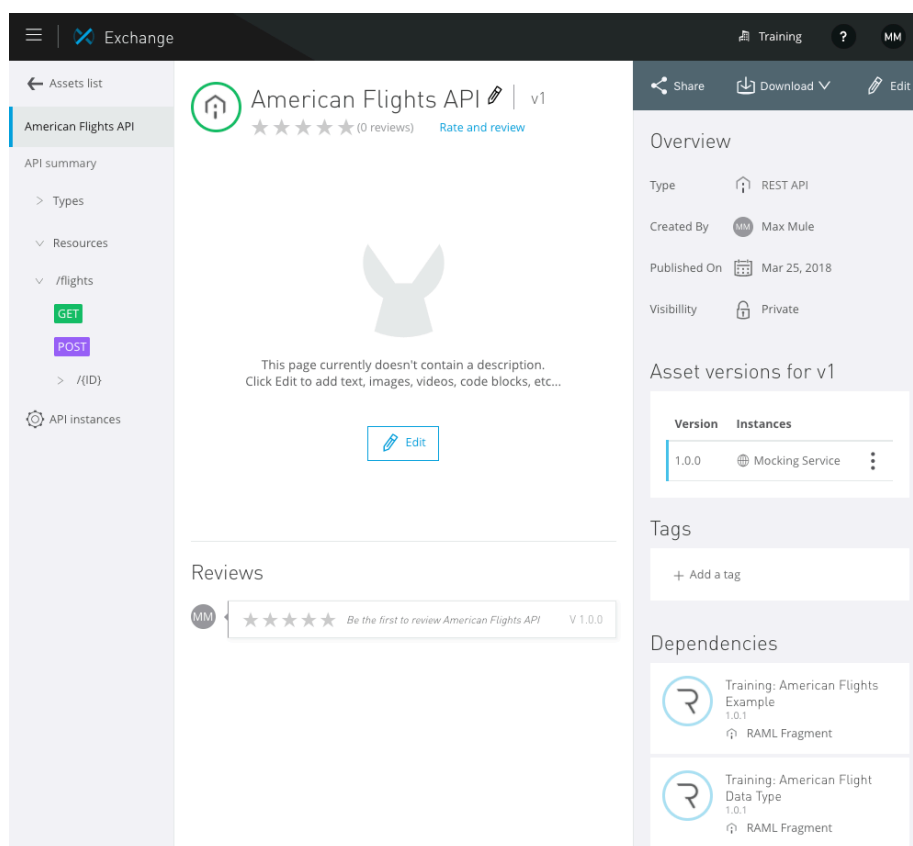


15. In the left-side navigation, select the name of your organization (Training in the screenshots); you should see your American Flights API in your private Exchange.

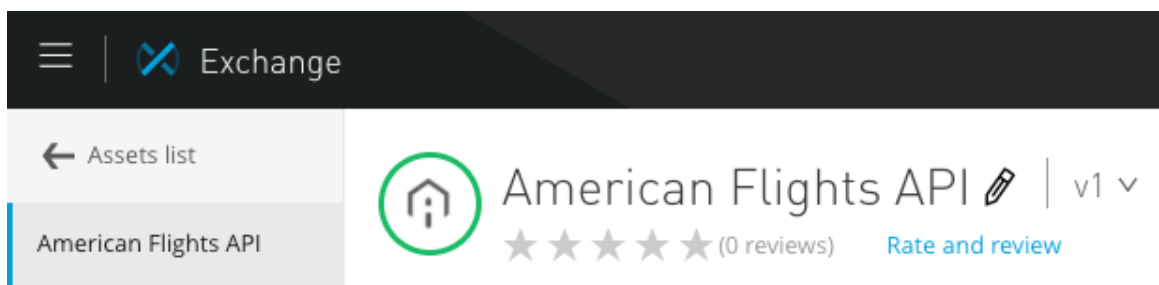


## Review the auto-generated API portal

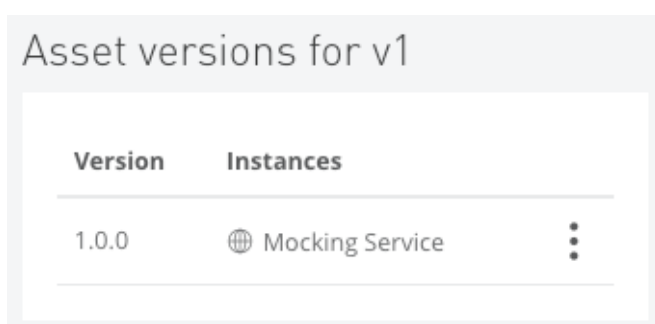
16. Click the American Flights API.
17. Review the page; you should see that as the creator of this API, you can edit, review, share, download, and add tags to this version.



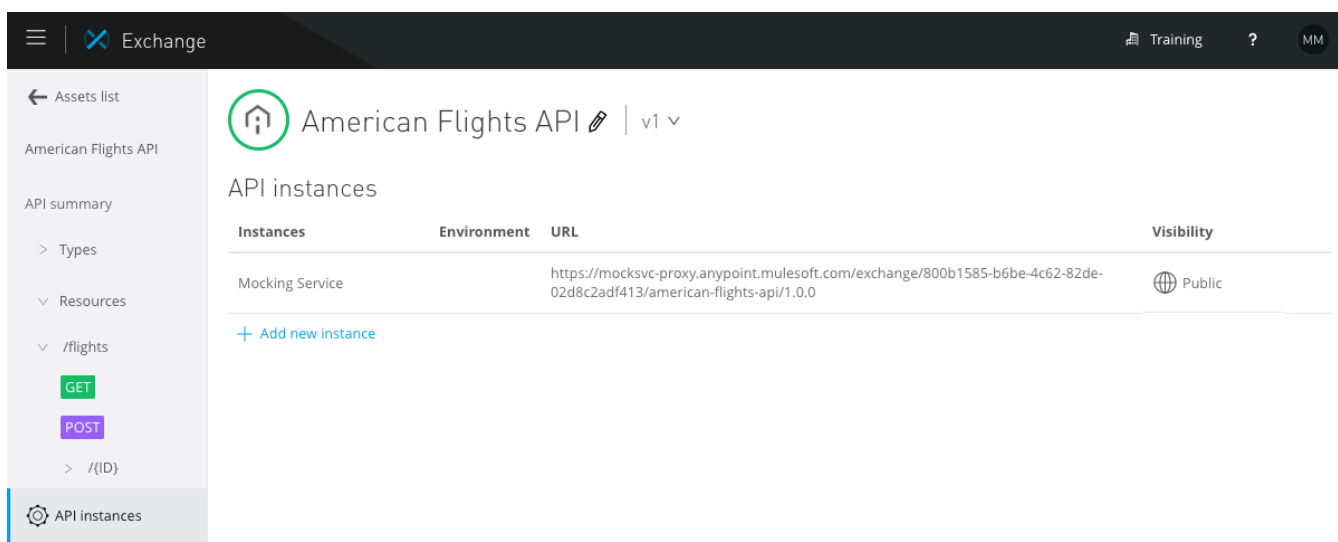
18. Locate the API dependencies in the lower-right corner.
19. Locate the API version (v1) next to the name of the API at the top of the page.



20. Locate the asset versions for v1; you should see one version (1.0.0) of this API specification (v1) has been published and there is one API instance for it and that uses the mocking service.



21. In the left-side navigation, select API instances; you should see information for the API instance generated from the API specification using the mocking service.



22. In the left-side navigation, expand Types.

23. Select AmericanFlight and review the information.

The screenshot shows the MuleSoft Exchange interface. On the left is a navigation sidebar with 'Assets list', 'American Flights API', 'API summary', 'Types', 'AmericanFlight', 'Resources', '/flights', and 'API instances'. The main area displays the 'American Flights API' with a version dropdown set to 'v1'. Below the API name is a JSON schema for the 'AmericanFlight' type. A table below the schema lists parameters: 'ID' (integer) and 'code' (string, required).

```
{
  "ID": "integer",
  "code": "string",
  "price": "number",
  "departureDate": "string",
  "origin": "string",
  "destination": "string",
  "emptySeats": "integer",
  "plane": {
    "type": "string",
    "totalSeats": "integer"
  }
}
```

Parameter	Type	Description
ID	integer	
code (required)	string	

## Test the API in its API portal in Exchange

24. In the left-side navigation, select the /flights GET method; you should now see the API console on the right side of the page.
25. In the API console, select to show optional query parameters.
26. Click the destination drop-down and select a value.

The screenshot shows the API console for the 'American Flights API' at version 'v1'. The left sidebar is the same as in the previous screenshot. The main area shows the endpoint '/flights : get' with a 'Request' section containing 'GET /flights'. Below this is a 'Parameters' table. On the right, a 'Mocking Service' panel is open, showing the URL 'https://mocksvc-proxy.anypoint.mulesoft.com/exc'. It has tabs for 'Parameters' and 'Headers'. Under 'Parameters', there is a 'Query parameters' section with a checkbox 'Show optional parameters' checked. Below this is a dropdown menu for 'LAX' and a 'Send' button.

Parameter	Type	Description
Query parameters		
destination	string (enum)	Possible values: SFO, LAX, CLE

27. Click Send; you should get a 200 response and the example data displayed – just as you did in API console in API designer.

The screenshot shows the API console interface. At the top, there's a green 'GET' button. Below it, a dropdown menu shows 'Mocking Service'. The URL is 'https://mocksvc-proxy.anypoint.mulesoft.com/exc'. There are tabs for 'Parameters' and 'Headers'. Under 'Parameters', there's a 'Query parameters' section with a dropdown set to 'LAX' and a 'Show optional parameters' checkbox. A blue 'Send' button is to the right. Below the 'Send' button, the response status is '200 OK' with a response time of '582.08 ms' and a 'Details' link. At the bottom, there are icons for clipboard, download, code editor, and table view. The response body is displayed in a code editor, showing a JSON array of two flight objects.

```
[Array[2]
  -0: {
    "ID": 1,
    "code": "ER38sd",
    "price": 400,
    "departureDate": "2017/07/26",
    "origin": "CLE",
    "destination": "SFO",
    "emptySeats": 0,
    -"plane": {
      "type": "Boeing 737",
      "totalSeats": 150
    }
  },
  -1: {
    "ID": 2,
    "code": "ER45if".
```

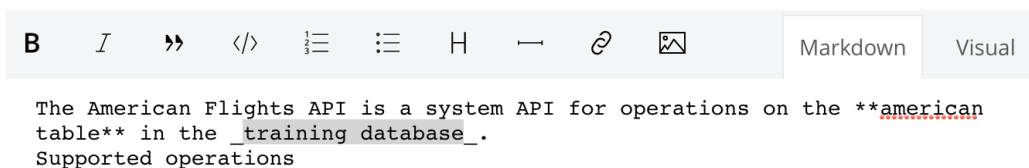
## Add information about the API

28. In the left-side navigation, select the name of the API: American Flights API.
29. Click one of the Edit buttons for the API.
30. Return to the course snippets.txt file and copy the text for the American Flights API description text.
31. Return to the editor in Anypoint Exchange and paste the content.

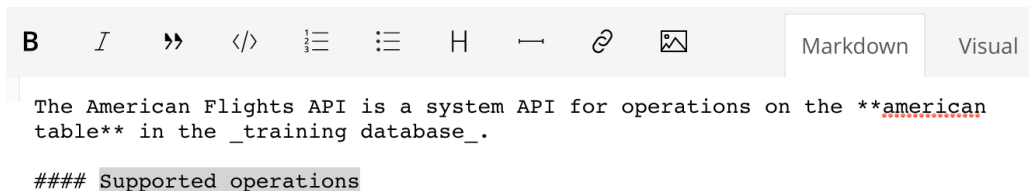
32. Select the words american table and click the strong button (the B).



33. Select the words training database and click the emphasis button (the I).

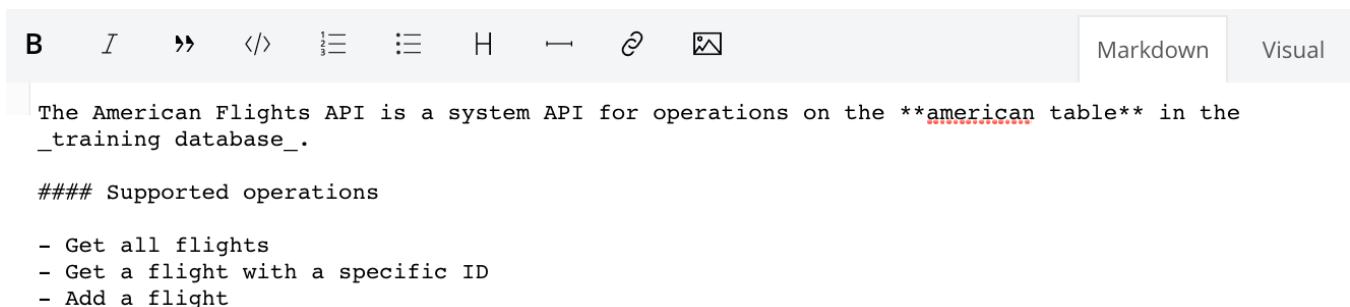


34. Select the words Supported operations and click the heading button four times (the H).



35. Select Get all flights and click the bulleted list button.

36. Repeat for the other operations.





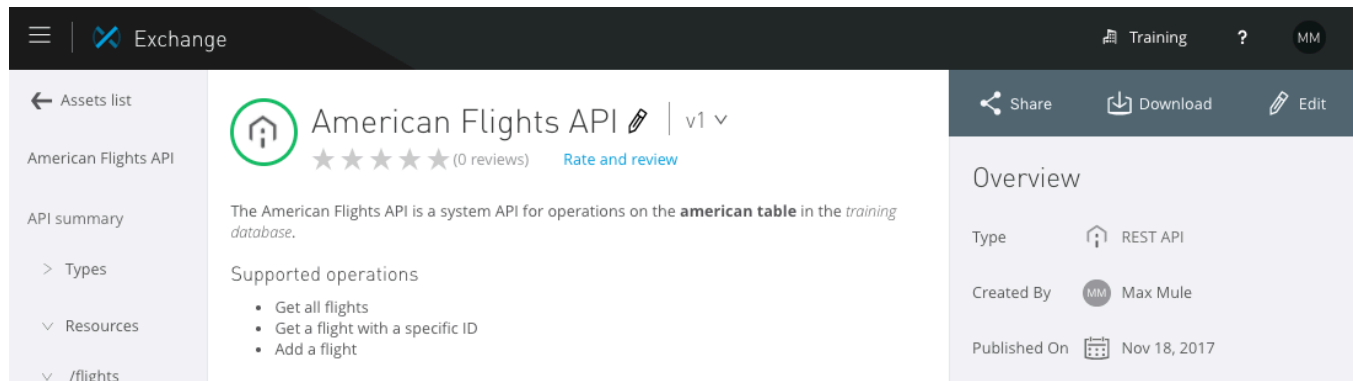
37. Select the Visual tab in the editor toolbar and view the rendered markdown.

The screenshot shows the MuleSoft Exchange interface for editing an API. The top navigation bar includes a menu icon and the word "Exchange". On the left, a sidebar contains links for "Assets list", "American Flights API", "+ Add new page", "API summary", "+ Add terms and conditions", and "API instances". The main content area displays the API title "American Flights API" with a version dropdown set to "v1". Below the title is a rich text editor toolbar with icons for bold, italic, link, code, list, and other formatting options. The editor shows the following text: "The American Flights API is a system API for operations on the **american table** in the *training database*." followed by "Supported operations" and a bulleted list: "Get all flights", "Get a flight with a specific ID", and "Add a flight". At the bottom right, there are two buttons: "Discard changes" and "Save as draft".

38. Click the Save as draft button; you should now see buttons to exit the draft or publish it.

This screenshot shows the same API editor after clicking "Save as draft". The top right of the interface now includes "MuleSoft", a help icon, and a user profile icon labeled "MM". A dark banner across the top of the editor area states "This is a draft that has not been published yet." with an "Exit Draft" button. The left sidebar remains the same. The main content area shows the API title "American Flights API" with a version dropdown "v1" and a star rating "(0 reviews)". The description text is the same as in the previous screenshot. On the right, an "Overview" panel displays the API type as "REST API" and the creator as "Max Mule" (with the "MM" profile icon). At the top right of the editor area, there are now "Publish" and "Edit" buttons.

39. Click the Publish button; you should now see the new information about the API in the API portal.



## Modify the API and publish the new version to Exchange

40. Return to your American Flights API in API designer.
41. Return to the course snippets.txt file and copy the American Flights API - /{ID} DELETE and PUT methods.
42. Return to american-flights-api.raml and paste the code after the {ID}/get method.
43. Fix the indentation.
44. Review the code.

```
/{ID}:
  get:
    responses:
      200:
        body:
          application/json:
            type: AmericanFlight
            example: !include examples/AmericanFlightExample.raml
    delete:
      responses:
        200:
          body:
            application/json:
              example:
                message: Flight deleted (but not really)
    put:
      body:
        application/json:
          type: AmericanFlight
          example: !include examples/AmericanFlightNoIDExample.raml
      responses:
        200:
          body:
            application/json:
              example:
                message: Flight updated (but not really)
```

45. Click the Publish to Exchange button.

46. In the Publish API specification to Exchange dialog box, look at the asset version.

### Publish API specification to Exchange

Name (required)	Asset version ⓘ (required)
American Flights API	1.0.1
	Current version: 1.0.0
Main file (required)	API version ⓘ (required)
american-flights-api.raml	v1
Valid RAML	

[Show advanced >](#)

CancelPublish

47. Click Publish.

48. In the Publish API specification to Exchange dialog box, click Exchange; Exchange should open in a new browser tab.

49. Locate the asset versions listed for the API; you should see both asset versions of the API listed with an associated API instance using the mocking service for the latest version.

#### Asset versions for v1

Version	Instances
1.0.1	Mocking Service
1.0.0	

50. Click the Edit button.

51. Add two new operations that delete a flight and update a flight.

**B** *I* **»** `</>` `≡` `≡` **H** `←` `↻` `🖼`

MarkdownVisual

The American Flights API is a system API for operations on the **american table** in the `_training database_`.

#### Supported operations

- Get all flights
- Get a flight with a specific ID
- Add a flight
- Delete a flight
- Update a flight

52. Click Save as draft and then Publish.

The screenshot displays the MuleSoft Exchange interface for the 'American Flights API' asset. The interface is divided into a left sidebar, a main content area, and a right sidebar.

**Left Sidebar:**

- Assets list
- American Flights API
- API summary
  - > Types
  - > Resources
    - > /flights
      - GET
      - POST
    - > /(ID)
  - API instances

**Main Content Area:**

- Header: American Flights API | v1
- Rating: 5 stars (0 reviews) | [Rate and review](#)
- Description: The American Flights API is a system API for operations on the **american** table in the *training* database.
- Supported operations:
  - Get all flights
  - Get a flight with a specific ID
  - Add a flight
  - Delete a flight
  - Update a flight
- Reviews section: A review from user 'MM' with 5 stars and the text 'Be the first to rate American Flights API'.

**Right Sidebar:**

- Share, Download, Edit buttons
- Overview
  - Type: REST API
  - Created By: Max Mule
  - Published On: Nov 18, 2017
  - Visibility: Private
- Asset versions for v1


Version	Instances
1.0.1	Mocking Service
1.0.0	

- Tags
- + Add a tag
- Dependencies
- Training: American Flights Example 1.0.1 (RAML Fragment)
- Training: American Flight Data Type 1.0.1 (RAML Fragment)