Resources / Labs (/COMP9321/22T1/resources/72107) / Week 6 (/COMP9321/22T1/resources/72112) / Flask-RESTX (cont.)

Flask-RESTX (cont.)

Prerequisites:

It is assumed that you will install and take a look at the following packages in python before heading to activities:

- flask-restx (https://flask-restx.readthedocs.io/en/latest/)
- (http://flask.pocoo.org/) pandas (https://pandas.pydata.org/)

You do not need to learn how to create a flask web application (e.g. Jinja2 is not necessary).
For testing your RESTful APIs, you will need to be familiar with

- Swagger doc (https://swagger.io/)

All of the following activities will be based on the *Book dataset* (Download the CSV file (https://github.com/mysilver/COMP9321-Data-Services/blob/master/Week4_Visualization/Books.csv))

Activity-1: Get a List of Books

Description : Create a new Resource class to return all books.

Steps :

1. Make a copy of your code from the previous week
2. Create a new resource class
3. Add a new method called get to your resource class
   The method must return all books in the dataset which can be ordered (ascending or descending order) by any of columns
   You can use query parameters to pass the column name and order type (https://flask-restx.readthedocs.io/en/latest/parsing.html) .
4. You need to use a decorator called expect() (https://flask-restx.readthedocs.io/en/stable/parsing.html#file-upload) which accepts a request parser as input; this will automatically help flask-restx to generate appropriate swagger doc which includes inputs for query parameters
5. Run the application and browse http://127.0.0.1:5000/ (http://127.0.0.1:5000/) to test the endpoint. Test your API to see if it works properly.

</> (https://github.com/mysilver/COMP9321-Data-Services/blob/master/Week6_Flask2/activity_1.py)

Activity-2: Add a new Book

Description : Add another endpoint to the resource class which allow users to add new books.

Steps :

1. Make a copy of your code from the previous activity
2. Create a new method named "post"
   Details of a new book should be posted via request payload.
   The endpoint must return appropriate error message if the given id (book identifier) already exists in the dataset
3. Iterate over the key-values of the payload and update the dataframe
   Return error messages if the input includes unexpected keys
4. Run the application and browse http://127.0.0.1:5000/ (http://127.0.0.1:5000/) to test the endpoint. Test your API to see if it works properly.

</> (https://github.com/mysilver/COMP9321-Data-Services/blob/master/Week6_Flask2/activity_2.py)

Activity-3: Improving Swagger Documentation

Description : So far we did not manually created the swagger documentation; however, still the documentation can be improved by manually adding some information (e.g. API description, parameter description)

Steps :

1. Make a copy of your code from the previous activity
2. When creating an instance of API class, specify the title, description, default namespace, and version of the API. (https://flask-restx.readthedocs.io/en/stable/swagger.html#swagger-ui)
3. Use @api.response (http://flask-restx.readthedocs.io/en/stable/swagger.html#documenting-with-the-api-response-decorator) to indicate which HTTP response code does each method returns and what they mean
4. Use @api.param (http://flask-restx.readthedocs.io/en/stable/swagger.html#method-parameters) to add descriptions for the parameters
5. Use @api.doc (http://flask-restx.readthedocs.io/en/stable/swagger.html#documenting-with-the-api-doc-decorator) to add descriptions for API methods

</> (https://github.com/mysilver/COMP9321-Data-Services/blob/master/Week6_Flask2/activity_3.py)

Resource created 30 days ago (Monday 28 March 2022, 11:42:36 AM).

Comments

(/COMP9321/22T1/forums/search?forum_choice=resource/74266)

(/COMP9321/22T1/forums/resource/74266)

Add a comment

There are no comments yet.