

Flask-2

Assignment Questions

Q1. Explain GET and POST methods.

Ans: - When a client communicates with a server over the web, this process is enabled by the Hypertext Transfer Protocol (HTTP). HTTP is a request-response protocol between a client and a server.

GET and POST Method is used commonly in HTTP request method,

1. GET Method:

GET is used to request data from a specified resource. It can retrieve any visible data to a client, such as HTML documents, images, and videos. To send a GET request, a client needs to specify the URL of the resource it wants to retrieve. The request is then sent to the server, which processes the request and sends the requested data back to the client.

2. POST Method:

The POST sends data to a server to create or update a resource. For example, it is often used to submit an HTML form to a server. To send a POST request, a client needs to specify the URL of the resource to which it wants to send data and the data itself. The request is then sent to the server, which processes the request and sends a response back to the client. The POST method is often used to submit forms or upload files to a server.

Differences between GET and POST

GET	POST
1. Data parameters are included in the URL and visible to everyone.	1. Data is not displayed in the URL but in the HTTP message body.
2. It is less secure because the URL contains part of the data sent.	2. It is safer because the parameters are not stored in web server logs or the browser history.
3. GET requests can be cached and remain in the browser history, This means GET requests can be bookmarked, shared, and revisited,	3. while POST requests cannot

GET requests are intended to retrieve data from a server and do not modify the server's state.	POST requests are used to send data to the server for processing and may modify the server's state.
The GET method is limited to a maximum number of characters, This is because the GET method sends data through the resource URL, which is limited in length,	While the POST method has no such limitation. While the POST method sends data through the HTTP message body, which has no such limitation.
The GET method supports only string data types,	while the POST method supports different data types such as string, numeric, binary, and so on.

Q2. Why is request used in Flask?

Ans: - To access the incoming data in Flask, you have to use the request object. The request object holds all incoming data from the request.

Which includes the mime type, referrer, IP address, raw data, HTTP method, and headers, among other things.

The request context keeps track of the request-level data during a request. Rather than passing the request object to each function that runs during a request, the request and session proxies are accessed instead.

Example: - In a Flask App, we have our own Webpage (Client) and a Server. The Server should process the data. The Request, in Flask, is an object that contains all the data sent from the Client to Server. This data can be recovered using the GET/POST Methods. POST is used when your application expects user input to be received by command or an HTTP request, while GET gets all the information before it even has a chance for submission. Using both methods at once gives perfect freedom but still requires complex UI patterns like AJAX calls etc, with most common applications being frameworks where multiple forms are necessary such as Slack Webhooks, MailgunMailserver, and eCommerce Commerce Framework. With Flask-Request class instead, we don't have any need anymore since this API allows us flexible handling of many other situations.

Q3. Why is redirect() used in Flask?

Ans: - Flask class has a redirect () function. When called, it returns a response object and redirects the user to another target location with specified status code. location parameter is the URL where response should be redirected. statuscode sent to browser's header, defaults to 302.

Example: - Google Chrome

When we access a website, our browser sends a request to the server, and the server replies with what is known as the HTTP status code, which is a three-digit number, The different reasons for errors are:

`flask.redirect(location,code=302)`

We'll discuss redirects and errors with Python Flask in this article. A redirect is used in the Unauthorized access or poor request.

Unsupported media file types.

Overload of the backend server.

Internal hardware/connection error.

Parameters:

`location(str)`: the location which URL directs to.

`code(int)`: The status code for Redirect.

Code: The default code is 302 which means that the move is only temporary.

Return: The response object and redirects the user to another target location with the specified code.

Q4. What are templates in Flask? Why is the `render_template()` function used?

Ans: - Templates are files that contain static data as well as placeholders for dynamic data. A template is rendered with specific data to produce a final document. Flask uses the Jinja template library to render templates. In your application, you will use templates to render HTML which will display in the user's browser.

Example: - `render_template` is a Flask function from the flask. templating package. `render_template` is used to generate output from a template file based on the Jinja2 engine that is found in the application's templates folder. Note that `render_template` is typically imported directly from the flask package instead of from flask.

Q5. Create a simple API. Use Postman to test it. Attach the screenshot of the output in the Jupyter Notebook.

Ans:-

Note: Create your assignment in Jupyter notebook and upload it to GitHub & share that GitHub repository link through your dashboard. Make sure the repository is public. Data Science Masters