Used: Java 17 and Spring Boot

Once I understood the problem I realized a few things that needed tob e done, so I created a list:

Split input into multiple Requests

Split Object into Strings,

Remove duplicates,

Sort,

Return

In order to split the input into multiple requests, I used the requestparam which is provided by Spring. The request param was named ‘u’ so that the input gets split whenever u? appears. Once the input was split into multiple requests, I iterate over them and send the requests. Since Spring Boot has built in Multithreading, I was able to skip creating my own multithreaded requests.

After sending the requests I split the json object into strings by using regex. I add the strings to a set to avoid duplicates before I sort them. Once the sorting was finished I was able to create my own json object and return it.

Things that are in question:

I was not sure what the output needs to look like. Currently the output does not 100% match the input because parsing a string into a json object can cause problems, because of the quotations. If I just return the Strings (which are currently outputted to the console) I have achieved the goal of this task. I’m not really familiar with Unit and Integration-tests, which is why I only wrote one Integration test. I did not want to portray myself as a expert by googling unit tests just so I can pass the task.

I also made the assumption that the requests will only be ‘http’ and not ‘https’ because testing on a local server would not have been possible. The code can include ‘https’ request by adding ‘https’ to the schemes array. I would usually create a Service folder and a Controller folder but since this was a small task, I did not see the need to do that.

Overall, a very fun experience.

Best Regards,

Yafet

Verwendet wurden: Java 17 und Spring Boot und dependencies (siehe maven datei)

Nachdem ich das Problem verstanden hatte, erkannte ich einige Dinge, die getan werden mussten, also erstellte ich eine Liste:

Aufteilung der Eingabe in mehrere Requests

Objekt in Strings aufteilen,

Duplikate entfernen,

Sortieren,

Return statement

Um den Eingabestring in mehrere Anfragen aufzuteilen, habe ich den Requestparam verwendet, der von Spring bereitgestellt wird. Der requestparam wurde 'u' genannt, so dass die Eingabe aufgeteilt wird, wenn u? erscheint. Sobald die Eingabe in mehrere Anfragen aufgeteilt wurde, habe ich die Anfragen gesendet. Da Spring Boot über integriertes Multithreading verfügt, konnte ich das Erstellen eigener Multithreading-Anfragen übersprungen werden.

Nach dem Senden der Anfragen zerlege ich das json-Objekt mit Hilfe von Regex in Strings. Daraufhin füge Ich die Strings zu einem Set hinzu, um Duplikate zu vermeiden, bevor ich sie sortiere. Sobald die Sortierung abgeschlossen war, konnte ich mein eigenes json-Objekt erstellen und es zurückgeben.

Dinge, die in Frage gestellt werden:

Ich war mir nicht sicher, wie die Ausgabe aussehen muss. Derzeit stimmt die Ausgabe nicht zu 100% mit der Eingabe überein, da das Parsen eines Strings in ein json-Objekt wegen der Anführungszeichen Probleme verursachen kann. Wenn ich nur die Strings zurückgebe (die derzeit auf der Konsole ausgegeben werden), habe ich das Ziel dieser Aufgabe erreicht. Mit Unit- und Integrationstests bin ich nicht wirklich vertraut, weshalb ich nur einen Integrationstest geschrieben habe. Ich wollte mich nicht als Experte darstellen, indem ich Unit-Tests google, nur damit ich die Aufgabe bestehen kann.

Ich bin auch davon ausgegangen, dass die Anfragen nur "http" und nicht "https" sein werden, da ein Test auf einem lokalen Server nicht möglich gewesen wäre. Der Code kann "https"-Anfragen einschließen, indem man "https" zum Schema-Array hinzufügt. Normalerweise würde ich einen Service-Ordner und einen Controller-Ordner erstellen, aber da es sich um eine kleine Aufgabe handelte, sah ich keine Notwendigkeit dafür.

Insgesamt eine sehr unterhaltsame Erfahrung.

Mit freundlichen Grüßen,

Yafet

Übersetzt mit www.DeepL.com/Translator (kostenlose Version)