**INSTRUCTIONS ON RUNNING THE CURRENCY EXCHANGE APPLICATION**

**Introduction**

1. For this programming assignment, I have developed a fully standalone web-based currency exchange application using Spring Boot.
2. Once the application is run, interactions with respective APIs would be done via HTTP.GET methods. Access to the APIs would be done via URL “http://localhost:8080”. The results would be presented as JSON messages, as elaborated in the subsequent sections.

**Overview**

1. Three APIs have been exposed in this application, namely –
   1. Given a date, get the exchange rate of all the currencies (wrt USD);
   2. Given a date and 2 currencies, find the exchange rate between them; and
   3. Given a date range and a given currency, find the exchange rate of that currency (wrt USD) for the entire date range.
2. Subsequent sections would elaborate on how these APIs would be executed, and expected results presented. For illustrative purposes, we would use the Postman Chrome app to present the results.

**Using API 1 – Finding the exchange rates of all currencies (wrt USD) for a given date**

1. To find the exchange rates of all currencies (wrt USD) for a given date, enter the URL into the browser in the format **localhost:8080/ByDate/{date}** where {date} is of the format “yyyy-MM-dd”. This is illustrated in the screenshot below, using “2017-01-01” as the specified date –

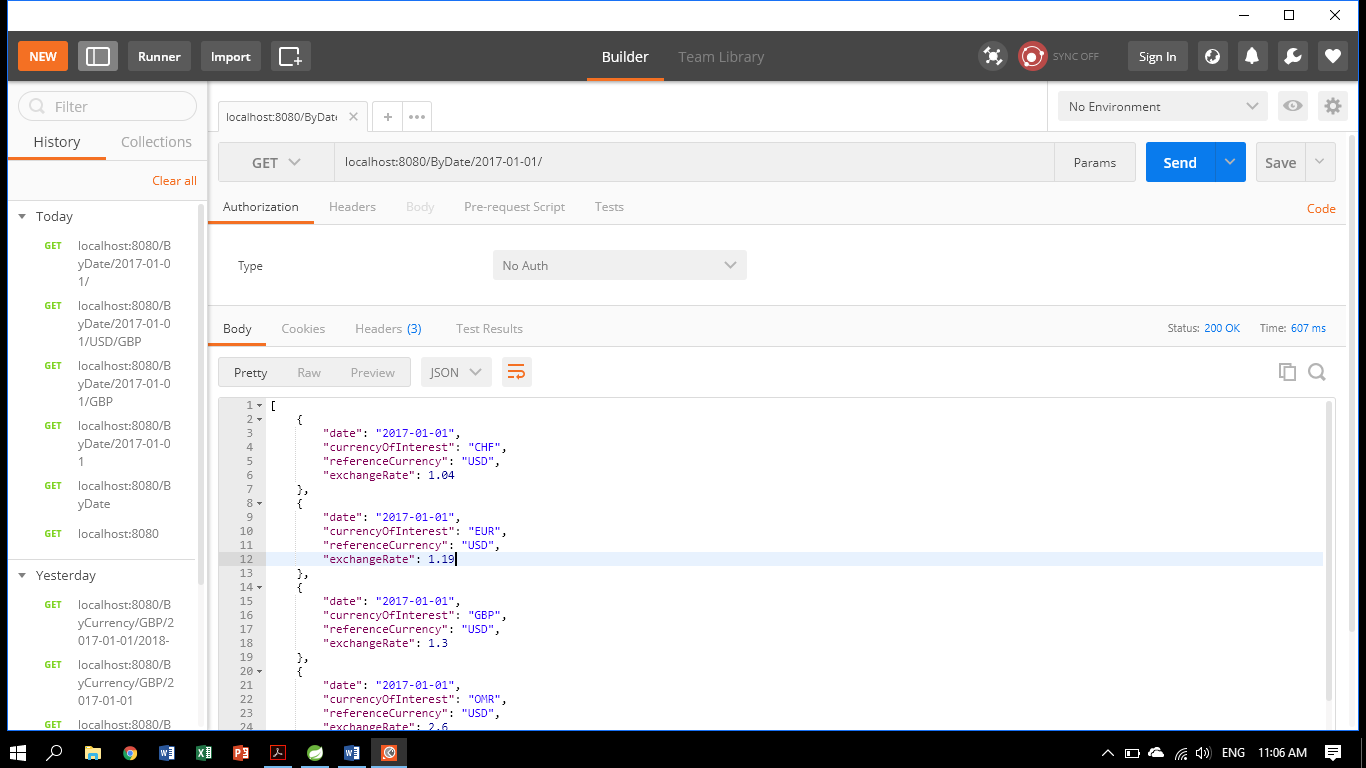


Figure 1 - Finding the exchange rates of all currencies (wrt USD) for a given date (**valid entry** **example**)

1. However, in cases where the **{date}** field is formatted wrongly, or the data is yet to be available in the repository, the following result would be provided instead to highlight that no such data is available for whatever reason –

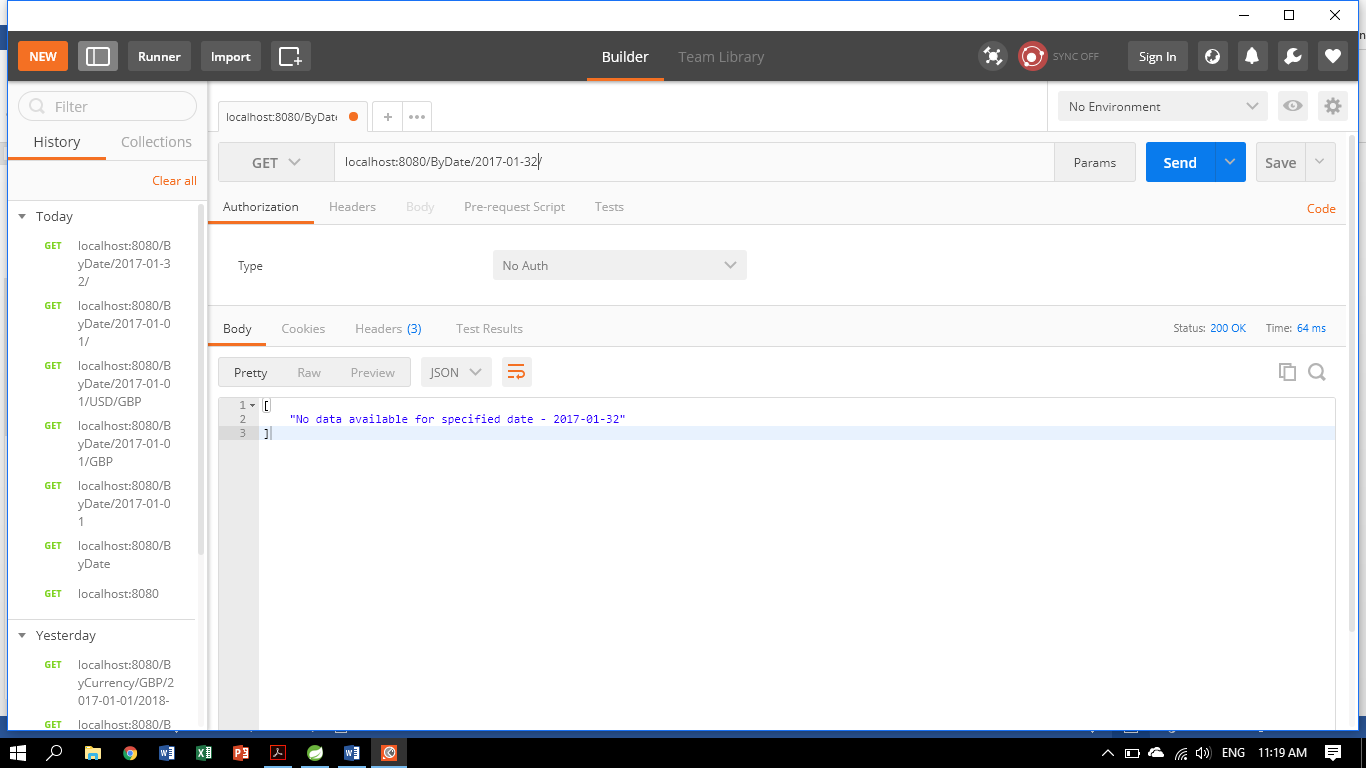


Figure 2 - Finding the exchange rates of all currencies (wrt USD) for a given date (**invalid entry example**)

**Using API 2 – Finding the exchange rate between two currencies given a certain date**

1. To find the exchange rate between two currencies given a certain date, use the URL **localhost:8080/ByDate/{date}/{currencyOfInterest}/{referenceCurrency}**. The {date} is of format “yyyy-MM-dd”, while {currencyOfInterest} and {referenceCurrency} use case-sensitive format “XXX”.
2. This is illustrated in the screenshot below, using “2017-01-01” as the specified date, and “SGD” and “EUR” as the currency of interest and reference currency respectively –

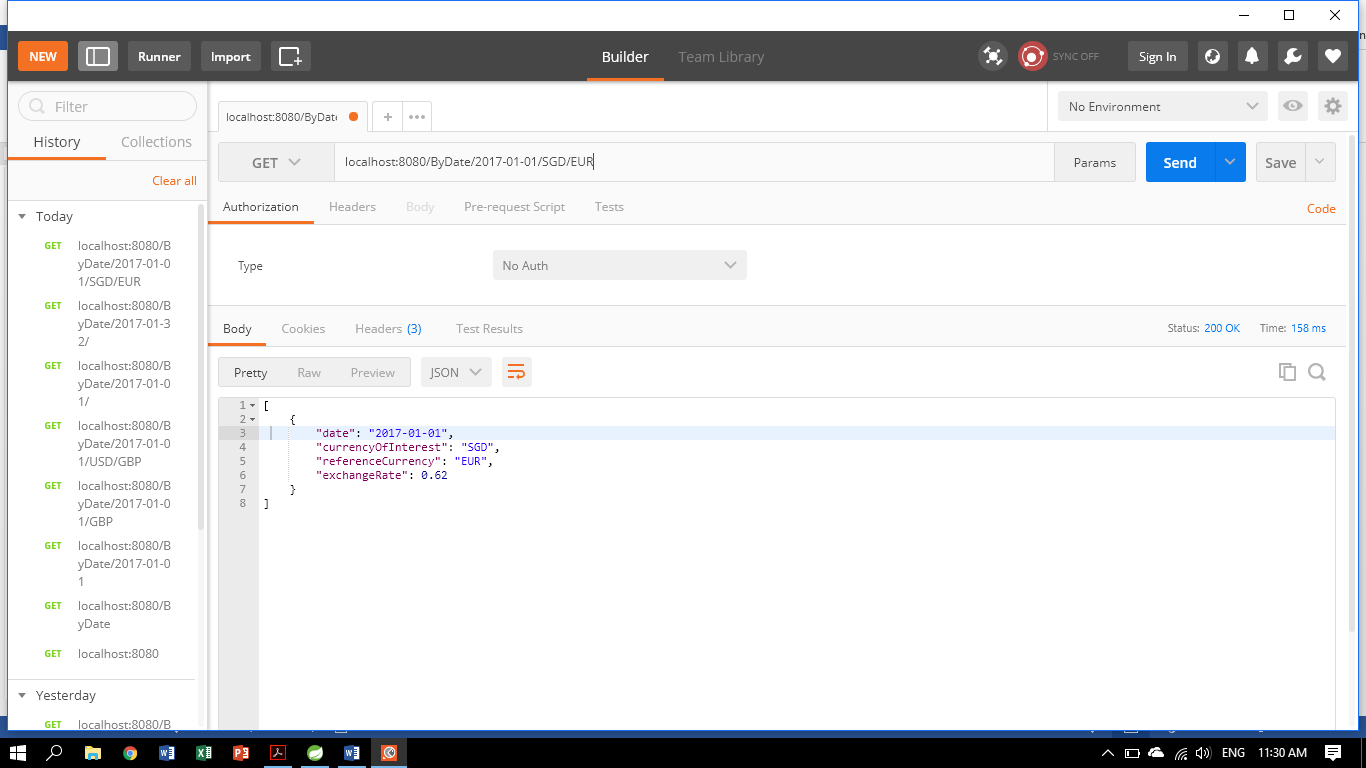


Figure 3 - Finding the exchange rates between two currencies for a given date (**valid entry** **example**)

1. Should any of the fields be formatted wrongly or data not being available, the API would return the following message instead to indicate that there is no data available for whatever reason –

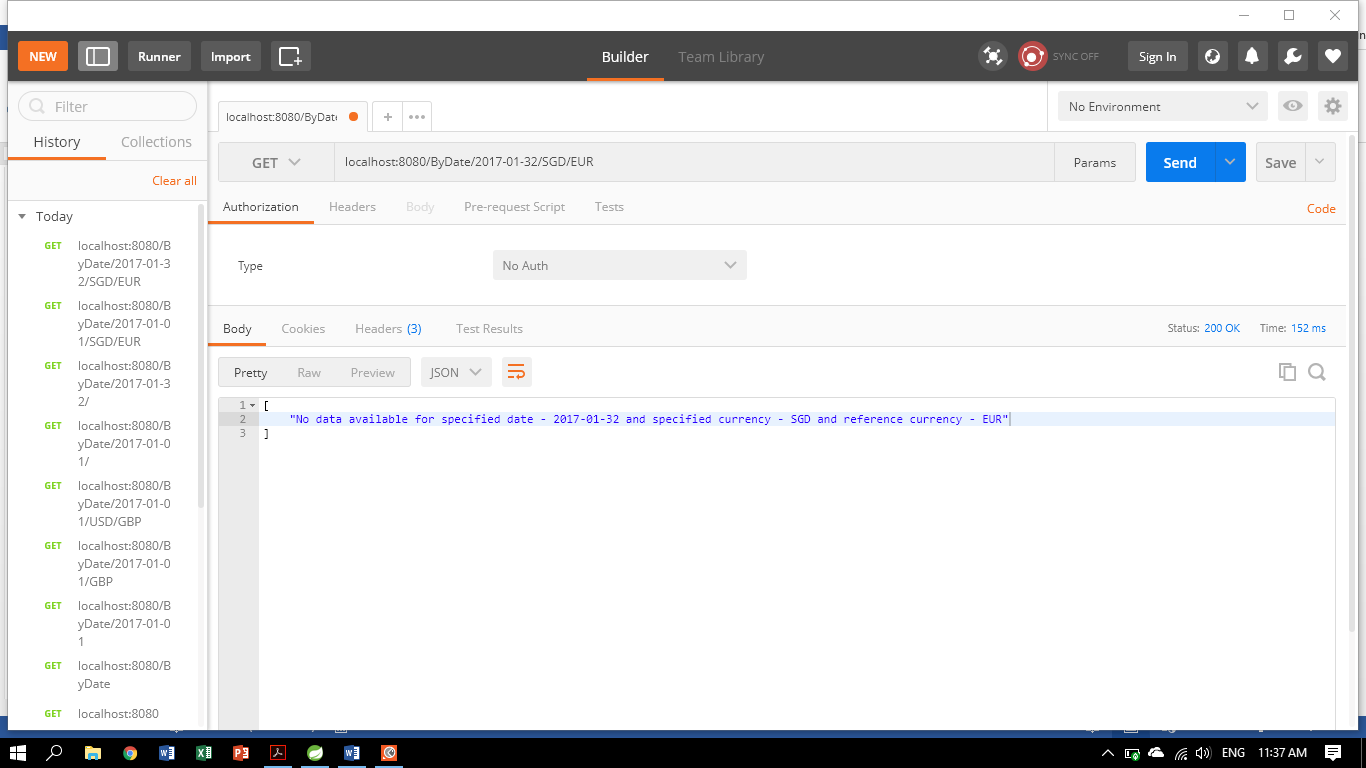


Figure 4 - Finding the exchange rates between two currencies for a given date (**invalid entry** **example**)

**API 3 – Finding the exchange rates for a given currency across a stipulated date range**

1. To find the exchange rates for a given currency across a stipulated date range, use the URL **localhost:8080/ByCurrency/{currencyOfInterest}/{startDate}/{endDate}**. The {startDate} and {endDate} are of format “yyyy-MM-dd”, while {currencyOfInterest} use case-sensitive format “XXX”.
2. This is illustrated below using “2017-01-01” and “2017-01-03” and “SGD” as the arguments –

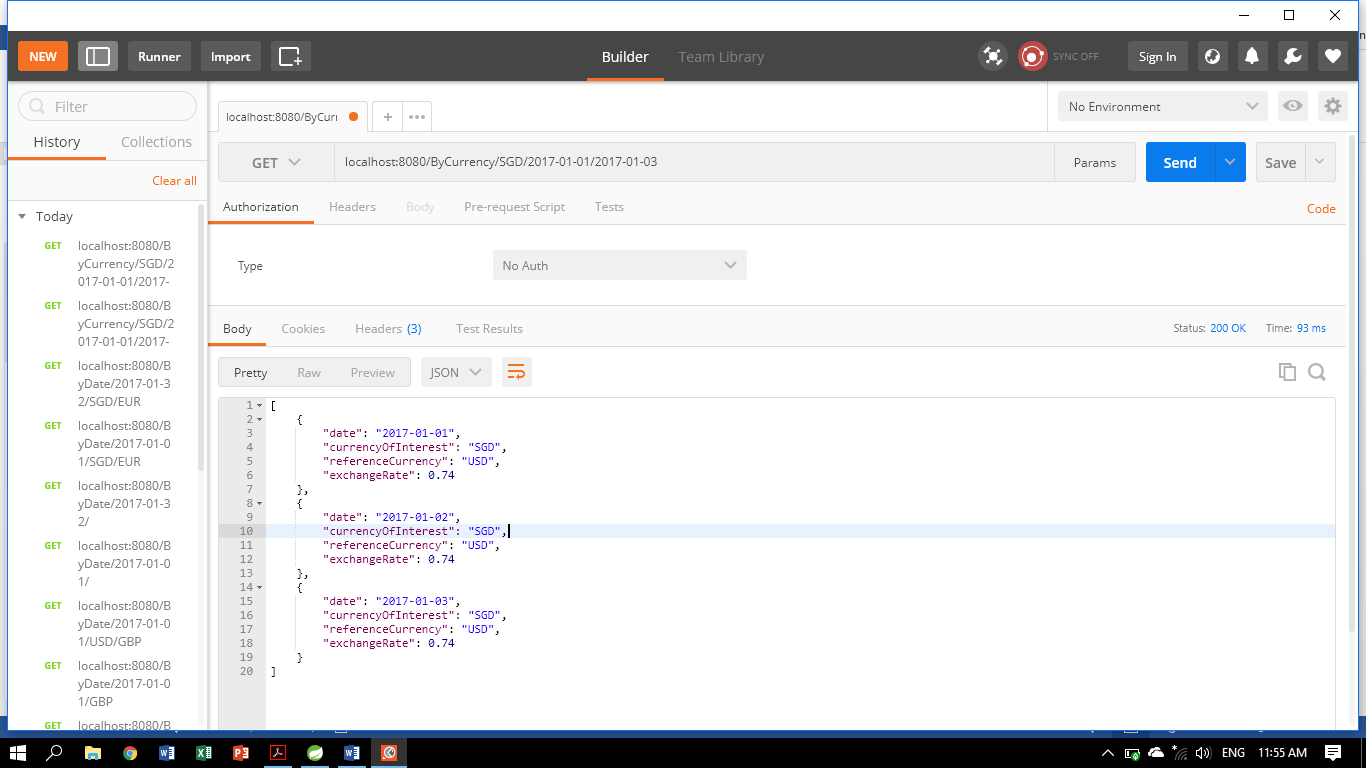


Figure 5 - Finding the exchange rates for a given currency across a stipulated date range (**valid entry** **example**)

1. This would still work even if the start and end dates swap positions, so long as they are valid entries –

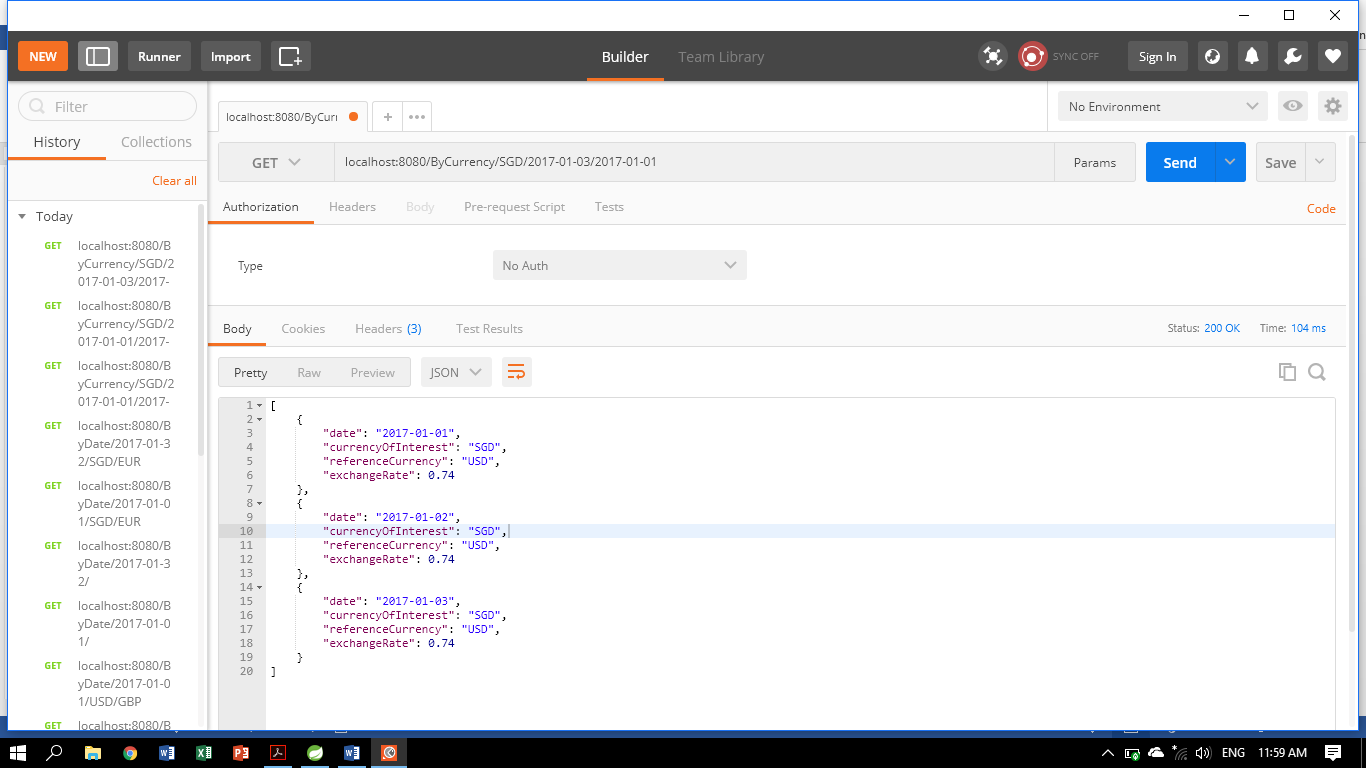


Figure 6 - Finding the exchange rates for a given currency across a stipulated date range (**valid** **start/end swap** **example**)

1. Should any of the fields be formatted wrongly or data not being available, the API would return the following message instead to indicate that there is no data available for whatever reason –

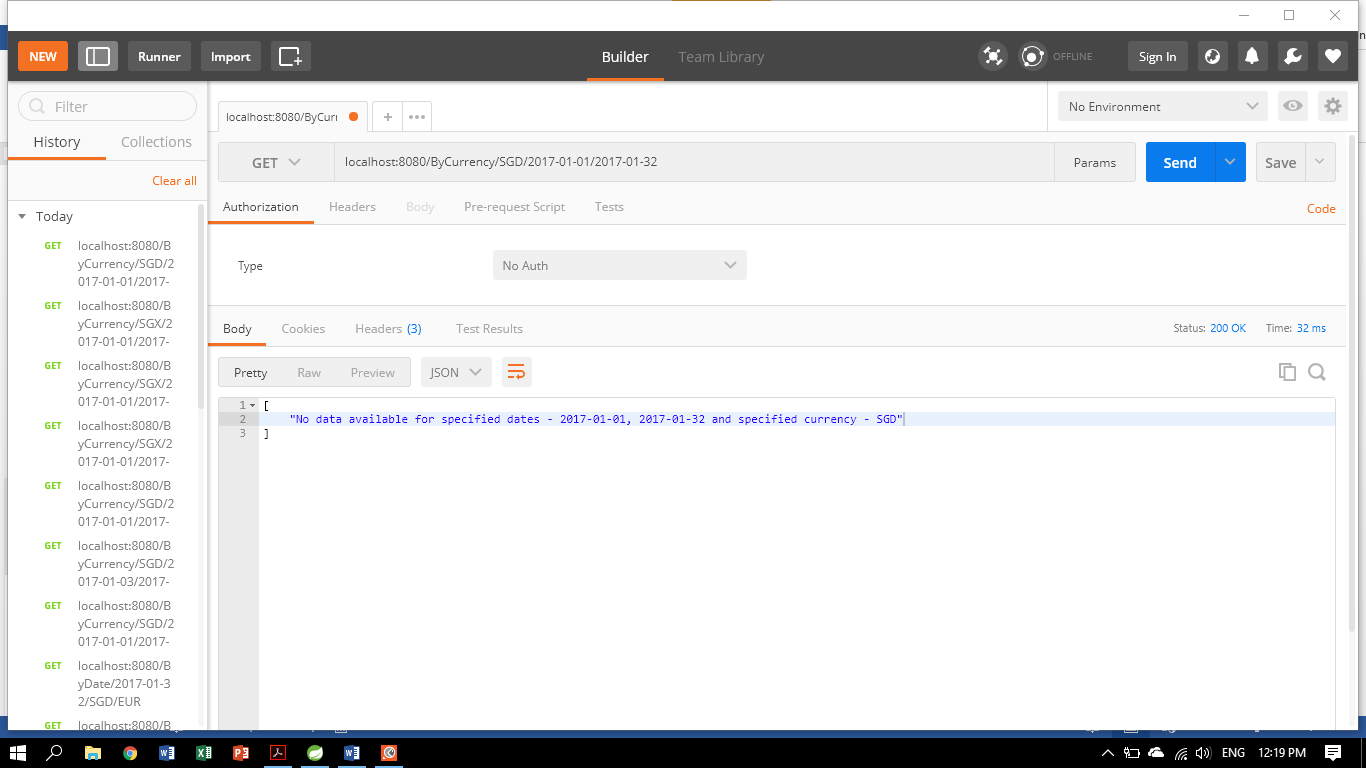


Figure 7 - Finding the exchange rates for a given currency across a stipulated date range (**invalid date** **example**)

~o0o~