**REST and JSON** 

General part

* *Elaborate on some of the characteristics of REST, like: Stateless, Cacheable, Layered System, Uniform Interface etc.*
* *Explain about the topics Same Origin Policy and CORS and why/how they are relevant when calling REST services from an Ajax-call within a browser.*
* Explain the benefit(s) from having a backend that exposes all data to clients via a REST-API
* *Elaborate on how JSON or XML supports communication between subsystems, even when the subsystems are implemented on different platforms.*

Practical part (You might need to add the gson dependency)

Use NetBeans to create a basic Maven Web Application for this exercise. You don't need any "WEB" for part one, but you do for part two.

**1)** Often we need test-data for our applications. Implement a Java class that can provide random test data as sketched below:

String testdData = Generator.generate(100,1000);

This should return a JSON Array with 100 test data on the form[[1]](#footnote-0):

[

{"fname": "Janne", "lname": "Peterson", "id": 1000, "age": 45},

{"fname": "Jan", "lname": "Olsen", "id": 1001, "age": 65},...

]

Rules for data generation:

* First number supplied to the method is the number of samples to generate
* Id's starts with the number supplied as the second argument. Subsequent id's will be generated by incrementing this number with 1.
* Age is a number between 17-70 (both inclusive)

**2)**

Implement a simple REST Service which, when called like this: GET: http:/……/api/data/50/1

Should return a JSON array with 50 persons as sketched below:

[

{"fname": "Ole", "lname":"Hansen", "id":1, "age": 25},

{"fname": "Peter", "lname":"Jensen", "id":2, "age": 35}

,...

]

**3)** Create a simple web page that, using React or plain javascript, renders a table which should show test data fetched via the REST method implemented in step 2.

1. *Hint: Include some hard coded arrays in your module like ["Ib", "Bo", "Lars", "Henrik"] and pick a random value for each name you generate* [↑](#footnote-ref-0)