CEGEP VANIER COLLEGE
CENTRE FOR CONTINUING EDUCATION
Web Services
420-941-VA

Teacher: Samir Chebbine **Lab 3** **Oct 02, 2024**

Lab 3: Web Services using REST Implementation

Complete all these following programs in class. All *missing coding statements* are presented in this YouTube video with explanation and in Presentation 3.

Create and Submit a Word file ***Lab3WebServicesYourName.doc*** which contains Answers of theory questions if any and output screenshots for every Java EE Project. Submit the Java projects too and submit the whole Lab 3 as compressed zip file

1. **Maven Dynamic Web Project: WebMathOperationsRESTProject**
   a) Create a new Dynamic Web project **WebMathOperationsRESTProject and convert it into Maven Project**. Check the output using Postman. Save your own screenshots.
   b) Add **Maven Project dependencies** in **pom.xml**. Create new package called **mathOperationsREST.**
   c) Deploy **WebMathOperationsRESTProject** within GalssFish Server.
   d) You need to develop a **Java class** called `MathOp`, which takes `x, y, z` as **private** `non static` members. The `MathOp` class contains the following method members:
      - Add a method called `calculateSum()` in `MathOp` class that returns `(x+2*y+3*z).`
      - Add a method called `calculatePrd()` in `MathOp` class that returns `(x*2*y*3*z).`
   e) Create a new REST Resource class **WebMathResource.java**

   1. Add a path URL mapping **("MathOp**") to access REST resource using appropriate REST annotation and call the following methods **calculateHTMLOp()/displayXYZJSON()**.

   2. Add a method **calculateHTMLOp()** that returns a HTML media type using appropriate Java REST annotations.

   Add appropriate statements in **calculateHTMLOp()** using **query string parameters** `x, y, z` that calls implemented methods `calculateSum()/calculatePrd()` in `MathOp`.

   3. Add a method **displayXYZJSON()** that returns a JSON media type and instantiate an `object of MatOp class type`. Set its data attributes to (1, 2, 3).

   4. Add a new path URL mapping ("/listArray…") that calls a method **displayListZYZ()** that returns a HTML media type using appropriate Java REST annotations.

   -Add appropriate statements in **displayListZYZ ()** to instantiate a Java data structure **Array List** `of object of MatOp class type` to be referenced by (`listXYZ`). Add every component of Array List course object to the following values `(1,2,3)(4,5,6)(7,8,9).`

   Skip through A`rray List of object` (`listXYZ`) and display its components as shown hereafter.

   5. Add a new path URL mapping ("/OpHashMap…") with path parameter x as search parameter to access REST resource searching into Hash Map.

   -Add a method **searchHashMapListZYZ()** using **path parameter** `x` and returns a JSON media type using appropriate Java REST annotations and will be fired upon using URL mapping ("/OpHashMap…").

   -Add appropriate statements in **searchHashMapListZYZ ()** to instantiate a data structure **HashMap** `of MathOp class type` to be referenced by (`opHashMap`) where hash map key represents `x (path parameter)` and value hash map of `MathOp class type`. Set every component of hash map to the following values:

`x = 1,y = 2,z = 3/ x = 4,y = 5,z = 6 / x = 7,y = 8,z = 9`

-Skip through `Hash Map collection` (`opHashMap`) and display the result of Hash Map search in JSON media type using **path parameter** `x`.