

TDM09 d'Informatique Répartie

REST

ASI4 - INSA Rouen
CORRECTION

Service d'annuaire en REST

Le but de cet exercice est de réaliser un service d'annuaire à l'aide d'un Serveur REST. Celui-ci permet de connaître le bureau d'appartenance d'une personne et réciproquement. La logique métier est fournie en annexe (fichier `annuaire.tar.gz`), il ne reste qu'à distribuer le service en REST et à créer un client. Pour cela, vous respecterez les contraintes énumérées ci-dessous.

- Le serveur respectera pour ses *EndPoints* l'API suivante :
 - GET, `annuaire/personnes` → retourne la liste des personnes et leur bureau associé ;
 - GET, `annuaire/personnes/BUREAU` → retourne la personne associée au BUREAU ;
 - GET, `annuaire/bureaux/PERSONNE` → retourne le bureau associé à une PERSONNE ;
 - POST, `annuaire/personnes` → ajoute une nouvelle personne à l'annuaire ;
 - PUT, `annuaire/personnes` → met à jour les informations d'une personne de l'annuaire ;
 - DELETE, `annuaire/personnes/PERSONNE` → supprime une PERSONNE de l'annuaire ;
- Le client sera accessible en ligne de commande, et donnera accès à chacun des EndPoints du serveur. Par exemple il pourra prendre comme premier argument la méthode HTTP appelée (GET, POST, PUT ou DELETE), suivi éventuellement des paramètres nécessaires à l'appel des différentes méthodes de l'API REST :
 - GET : retourne la liste des personnes et leur bureau associé ;
 - GET bureau PERSONNE : retourne le bureau associé à PERSONNE ;
 - GET personne BUREAU : retourne la personne associée au BUREAU ;
 - POST NOM BUREAU : ajoute à l'annuaire la personne NOM, associée au bureau BUREAU ;
 - PUT NOM BUREAU : met à jour les informations de PERSONNE dans l'annuaire, comme étant associé au bureau BUREAU ;
 - DELETE PERSONNE : supprime PERSONNE de l'annuaire ;

NB : il s'agit d'Informatique Répartie donc pensez à tester votre programme avec serveur et client sur des machines différentes !

Correction

annuaire/Personne.java

```
package annuaire;

import javax.xml.bind.annotation.XmlElement;
import javax.xml.bind.annotation.XmlRootElement;

@XmlRootElement(name="personne")
public class Personne {
    private String nom, bureau;

    public Personne() {}

    public Personne(String name, String office) {
        this.nom = name;
        this.bureau = office;
    }

    public String getNom() {
        return this.nom;
    }

    @XmlElement
    public void setNom(String name) {
        this.nom = name;
    }

    public String getBureau() {
        return this.bureau;
    }
}
```

```

@XmlElement
public void setBureau(String office) {
    this.bureau = office;
}

@Override
public String toString() {
    return this.nom + " : " + this.bureau;
}
}

```

annuaire/AnnuaireModule.java

```

package annuaire;

import com.google.inject.Binder;
import com.google.inject.Module;

public class AnnuaireModule implements Module
{
    public void configure(final Binder binder)
    {
        binder.bind(annuaire.AnnuaireRessource.class);
    }
}

```

annuaire/AnnuaireRessource.java

```

package annuaire;

import javax.ws.rs.*;
import javax.ws.rs.core.Response;

import java.util.HashMap;
import java.util.Iterator;

import annuaire.Personne;

@Path("annuaire")
public class AnnuaireRessource
{
    private static HashMap<String,String> listeDePersonnes = new HashMap<String,String>();
    private static HashMap<String,String> listeDeBureaux = new HashMap<String,String>();

    @GET
    @Path("bureau/{nom}")
    @Produces("text/plain")
    public String getBureau(@PathParam("nom") final String nom) {
        return AnnuaireRessource.listeDeBureaux.get(nom);
    }

    @GET
    @Path("personnes/{bureau}")
    @Produces("text/plain")
    public String getPersonne(@PathParam("bureau") final String bureau) {
        return AnnuaireRessource.listeDePersonnes.get(bureau);
    }

    @GET
    @Path("personnes")
    @Produces("text/plain")
    public String toString() {
        String annuaire;
        if (AnnuaireRessource.listeDePersonnes.size() == 0)
            annuaire = "Liste vide";
        else {
            annuaire = "";
            for (Iterator i=AnnuaireRessource.listeDeBureaux.keySet().iterator();i.hasNext();){
                String nom = (String)i.next();
                annuaire += nom + " : " + AnnuaireRessource.listeDeBureaux.get(nom);
                if (i.hasNext())
                    annuaire += "\n";
            }
        }
        return annuaire;
    }

    @POST
    @Path("personnes")
    @Consumes("application/xml")
    public Response addPersonne(Personne personne) {
        AnnuaireRessource.listeDeBureaux.put(personne.getBureau(), personne.getBureau());
        AnnuaireRessource.listeDePersonnes.put(personne.getBureau(), personne.getNom());
        return Response.status(200).entity(personne.toString()).build();
    }

    @PUT
    @Path("personnes")
    @Consumes("application/xml")

```

```

public Response updateTerm(Personne personne) {
    String ancienBureau = this.getBureau(personne.getNom());
    AnnuaireRessource.listeDeBureaux.remove(personne.getNom());
    AnnuaireRessource.listeDeBureaux.put(personne.getNom(), personne.getBureau());
    AnnuaireRessource.listeDePersonnes.remove(ancienBureau);
    AnnuaireRessource.listeDePersonnes.put(personne.getBureau(), personne.getNom());
    return Response.status(200).entity(personne.toString()).build();
}

@DELETE
@Path("personnes/{nom}")
public Response removeTerm(@PathParam("nom") final String nom) {
    String bureau = AnnuaireRessource.listeDeBureaux.get(nom);
    AnnuaireRessource.listeDeBureaux.remove(nom);
    AnnuaireRessource.listeDePersonnes.remove(bureau);
    return Response.status(200).entity(nom).build();
}
}

```

webapp/WEB-INF/web.xml

```

<?xml version="1.0"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3/EN"
    "http://java.sun.com/dtd/web-app_2_3.dtd">

<web-app>
  <display-name>Dictionnaire REST</display-name>

  <context-param>
    <param-name>resteasy.guice.modules</param-name>
    <param-value>annuaire.AnnuaireModule</param-value>
  </context-param>

  <listener>
    <listener-class>
      org.jboss.resteasy.plugins.guice.GuiceResteasyBootstrapServletContextListener
    </listener-class>
  </listener>

  <servlet>
    <servlet-name>Resteasy</servlet-name>
    <servlet-class>
      org.jboss.resteasy.plugins.server.servlet.HttpServletDispatcher
    </servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>Resteasy</servlet-name>
    <url-pattern>*/</url-pattern>
  </servlet-mapping>
</web-app>

```

client/AnnuaireClient.java

```

package client;

import org.jboss.resteasy.client.jaxrs.ResteasyClient;
import org.jboss.resteasy.client.jaxrs.ResteasyClientBuilder;
import org.jboss.resteasy.client.jaxrs.ResteasyWebTarget;
import javax.ws.rs.core.Response;
import javax.ws.rs.client.Entity;

import annuaire.Personne;

public class AnnuaireClient {
    public static void main(String[] args) {
        ResteasyClient client = new ResteasyClientBuilder().build();
        ResteasyWebTarget target;
        Response response;
        String baseURL = "http://localhost:8080/AnnuaireRest-1.0/annuaire";
        Personne personne;

        if (args[0].equals("GET")) {
            if (args.length==1) {
                target = client.target(baseURL + "/personnes");
                response = target.request().get();
                System.out.println(response.readEntity(String.class));
                response.close();
            } else {
                if (args[1].equals("bureau")) {
                    target = client.target(baseURL + "/bureaux/" + args[2]);
                    response = target.request().get();
                    System.out.println(args[2] + " : " + response.readEntity(String.class));
                    response.close();
                } else if (args[1].equals("personne")) {
                    target = client.target(baseURL + "/personnes/" + args[2]);
                    response = target.request().get();
                    System.out.println(args[2] + " : " + response.readEntity(String.class));
                    response.close();
                }
            }
        }
    }
}

```

```

    } else {
        System.out.println("Requête non supportée en GET");
    }
}
} else if(args[0].equals("POST")) {
    personne = new Personne(args[1], args[2]);
    target = client.target(baseUrl + "/personnes");
    response = target.request().post(Entity.entity(personne, "application/xml;charset=UTF-8"));
    System.out.println("POST : " + response.getStatus());
    response.close();
} else if(args[0].equals("PUT")) {
    personne = new Personne(args[1], args[2]);
    target=client.target(baseUrl + "/personnes");
    response = target.request().put(Entity.entity(personne, "application/xml;charset=UTF-8"));
    System.out.println("PUT : " + response.getStatus());
    response.close();
} else if(args[0].equals("DELETE")) {
    target = client.target(baseUrl + "/personnes/" + args[1] + "");
    response = target.request().delete();
    System.out.println("DELETE : " + response.getStatus());
    response.close();
} else {
    System.out.println("Méthode non supportée");
}
}
}
}

```

runClient.sh

```

java -cp target/classes:lib/resteasy-client-3.0.9.Final.jar:lib/jaxrs-api-3.0.9.Final.jar:lib/httpclient-4.2.3.jar:
lib/httpcore-4.2.3.jar:lib/commons-logging-1.1.1.jar:lib/commons-io-2.3.jar:lib/resteasy-jaxrs-3.0.9.Final.jar
:lib/resteasy-jaxb-provider-3.0.9.Final.jar client.AnnuaireClient $1 $2 $3

```

Remarques

- Vous aurez alors accès à la correction du TDM dès la fin de la séance.
- **Déposez un compte-rendu de 2 pages TDMREST-LOGIN.pdf sur moodle chez TOUTES les personnes du binôme. Ce CR contiendra les informations que vous jugerez nécessaires.**
- Votre CR sera disponible pour vous lors de l'examen machine.