



TSU en Software

Práctica 12: Implementar un RESTful Web Service

Desarrollo de Aplicaciones II

M. Omar Vázquez González

Alumno: Carlos Araiza Dionicio

27-02-2021

Código

Código fuente:

<https://github.com/iCharlieAraiza/DDAII/tree/main/Pr%C3%A1ctica%2012>

```
public static void REST() throws SQLException, ClassNotFoundException {  
    int puerto=8000;  
  
    ArrayList<Persona> misPersonas=new ArrayList<Persona>() ;  
    inicializa(misPersonas);  
  
    //mostrarPagina();  
  
    PersonService personService = new PersonService(new ConnectDB() );  
  
    try{  
  
        ServerSocket miServidor = new ServerSocket(puerto);  
        Socket miCliente;  
  
        System.out.println("Esperando Conexion...");  
  
        miCliente= miServidor.accept();  
  
        System.out.println("Recibi Conexion");  
  
        //----- Habilitar lectura y escritura del Socket  
        BufferedReader lectorSocket;
```

```
PrintWriter escritorSocket;

InputStreamReader entradaDatos = new
InputStreamReader(miCliente.getInputStream());

lectorSocket= new BufferedReader(entradaDatos );

escritorSocket= new PrintWriter(miCliente.getOutputStream(), true);

//----

String mensajeRespuesta, mensajeRecibido="", lineaRecibida;
int opcion=0, totalContenido=0;

System.out.println("Esperando mensaje....");

lineaRecibida= lectorSocket.readLine();
mensajeRecibido=lineaRecibida+"\n";

lectorSocket.readLine();
String body = lectorSocket.readLine();

if(lineaRecibida.contains("GET"))
{
    opcion=1;
    //Agregar el ID de la persona
    if(body.length()==0){
```

```

        ArrayList<Persona> personas = (ArrayList<Persona>)
personService.selectAll();

        JSONArray jsonArray2 = new Gson().toJsonTree(personas).getAsJsonArray();
        System.out.println(jsonArray2.toString());
    }
    else{
        System.out.println(personService.select(Integer.parseInt(body)).toJson());
        String contenido = personService.select(Integer.parseInt(body)).toJson();

        mensajeRespuesta="HTTP/1.1 200 OK\n";
        mensajeRespuesta+="Content-Type: text/json\n";
        mensajeRespuesta+="Content-Length: " + contenido.length() + "\n\n";
        mensajeRespuesta+= contenido;
        escritorSocket.println(mensajeRespuesta);
    }

    personService.delete(3);

}
else if (lineaRecibida.contains("POST"))
{
    opcion=2;

    Persona persona = new Persona();
    persona.fromJson(body);

    personService.create(persona);

```

```
mensajeRespuesta="HTTP/1.1 200 OK\n";
mensajeRespuesta+="Content-Type: text/json\n";
mensajeRespuesta+="Content-Length: " + persona.toString().length() + "\n\n" ;
mensajeRespuesta+= persona.toJson();

escritorSocket.println(mensajeRespuesta);

//escritorSocket.println("✅ Se ha creado correctamente");

}
else if (lineaRecibida.contains("PUT"))
{
    opcion=3;
    Persona persona = new Persona();
    System.out.println(body);
    persona.fromJson(body);

    personService.update(persona);

    mensajeRespuesta="HTTP/1.1 200 OK\n";
    mensajeRespuesta+="Content-Type: text/json\n";
    mensajeRespuesta+="Content-Length: " + persona.toString().length() + "\n\n" ;
    mensajeRespuesta+= persona.toJson();

    escritorSocket.println(mensajeRespuesta);

}
else if (lineaRecibida.contains("DELETE")) {
    personService.delete(Integer.parseInt(body));
```

```
mensajeRespuesta="HTTP/1.1 200 OK\n";
mensajeRespuesta+="Content-Type: text/json\n";
mensajeRespuesta+="Content-Length: " + 0 + "\n\n" ;
mensajeRespuesta+=" ";

escritorSocket.println(mensajeRespuesta);


opcion=4;
}
else {
    opcion=-1;
}

System.out.println("Opcion: " + opcion);


}
catch(Exception ex) {
    System.out.println(ex.getMessage());
}

System.out.println("FIN del Programa");
}
```

Referencias

- Códigos de estado de respuesta HTTP - HTTP | MDN. (2020, 1 octubre). Mozilla.
<https://developer.mozilla.org/es/docs/Web/HTTP/Status>