Handling Cookies in RESTful Web Services

* We know that cookies are little piece of information in the form of “name=value String pair” exchanged between Client & Server
* Servers can store state information in cookies on the client, and can retrieve that information when the client makes its next request.
* Many web applications use cookies to set up a session between the client and the server.
* They also use cookies to remember identify and user preferences between requests. These cookies values are transmitted back and forth between the client and server

1. Sending Cookie as part of the Response:-

* An instance of javax.ws.rs.core.NewCookie is used to create a new HTTP cookie and transferred as part of a response.
* NewCookie is a Concrete Class part of JAX-RS API
* To Set Cookies as part of response,
* Create instances of NewCookie and
* pass then to ResponseBuilder.cookie() method.
* example:

@Path(“/myServices”)

public class MyServices

{

@GET

public Response get(){

NewCookie cookie = new NewCookie(“key”,”value”);

ResponseBuilder builder = Response.ok(“hello”,”text/plain”);

return builder.cookie(cookie).build();

}

}

1. Getting Cookie from the from the Request:

* javax.ws.rs.core.Cookie Represents the value of a Http cookie transferred in a request.
* The @javax.ws.rs.CookieParam annotation allows us to inject cookies sent by a client request into your JAX-RS resource methods
* For example, let’s say if web service sends a customerId cookie to clients so that we can track users as they invoke and interact with our web services, code to pull in this information might look like this:

@Path(“/myservice”)

public classs MyService

{

@GET

@Produces(“text/html”)

public String get(@CookieParam(“customerId”) int custId)

{

…

}

}

* @CookieParam makes JAX-RS to search all cookies with the name “customerId” . It then converts the corresponding value in to an int and injects it into the custId parameter
* If we need more information about the cookie other that its value, we can instead inject a javax.ws.rs.core.Cookie Object.

@Paht(“/myservice”)

public class MyService

{

@GET

@Produces(“text/html”)

public String get(@CoolieParam(“customerId”) Coolie custCookie)

{

…

}

}

* The Cookie class has additional contextual information about the cookie beyond its name and value

package javax.ws.rs.core;

public class Cookie

{

public String getName() {…}

public String getValue() {…}

public String getDomain( {…}

public String getPath() {…}

}

Note: - Use javax.ws.rs.core.NewCookie to send the Cookies as part of the Response.