**Exercícios – Web Services**

1) Retome a implementação do sistema de consulta de clientes desenvolvida no tutorial e exercícios sobre JPA.

a) Desenvolva uma camada de serviços SOAP+XML quer permita expor a funcionalidade do objeto FachadaLocal. Escolha a maneira que mais lhe convier para a implementação dessa camada (servlet ou EJB).

b) Caso o objeto FachadaLocal fosse implementado de modo a expor o objeto da entidade JPA no retorno da consulta de clientes, como ficaria o XML na serialização? Teste essa opção.

c) Desenvolva uma camada de serviços REST quer permita expor a funcionalidade do objeto FachadaLocal. Escolha a maneira que mais lhe convier para a serialização dos recursos.

2) Crie um site JSF que faça uso do web service de conversão de valores monetários disponibilizado (faça um teste para verificar qual serviço está online) na URL <http://www.webservicex.net/CurrencyConvertor.asmx?wsdl> ou <http://www.restfulwebservices.net/wcf/CurrencyService.svc?wsdl>. Ambos serviços foram desenvolvidos em .NET. O programa cliente deve apresentar as opções de moeda de origem, moeda de destino, valor a ser convertido e mostrar o resultado da conversão. Lembre-se que esse web service retorna a taxa de conversão. Importante: note que o serviço define um tipo de dado adicional (uma enumeração) para os códigos das moedas dos diversos países (verifique sua definição dentro do arquivo WSDL). Procure entender a definição de tipos exposta no WSDL através de XML-Schema e o mapeamento para as classes proxy criadas.

3) Crie um site JSF que consumirá um serviço REST que não possui uma descrição WADL. O Web Service REST a ser consumido é o serviço de Geolocalização via IP no endereço <http://freegeoip.net>. Utilize a API JAXB para realizar a desserialização dos dados JSON via um objeto DTO (*Data Transfer Object*). Siga os seguintes passos:

* Adicione uma classe com o nome GeoLoc que mapeia todos os dados disponíveis no arquivo JSON resultante da chamada ao web service. Importante: cuide para criar atributos com os mesmos nomes dos campos serializados em JSON. Decore a classe com as seguintes anotações:

@XmlRootElement

@XmlAccessorType(XmlAccessType.FIELD)

public class GeoLoc {

}

* Faça a chamada ao Web Service REST da seguinte maneira:  
  Client cliente = ClientBuilder.newClient();  
  WebTarget destino = cliente.target("http://freegeoip.net/");  
  GeoLoc geo = destino.path("json").request().get(GeoLoc.class);
* Obtenha os dados a partir do objeto GeoLoc.