**Exercice 1 : JAX-RS**

L'objectif de cet exercice est de mettre en place un service web RESTful pour la gestion d’une ToDoList. Une tâche est caractérisée par un titre(String), une date(String) et une priorité(String).

On veut effectuer les opérations suivantes sur une tâche :

* Ajouter une tâche
* Chercher les tâches selon une priorité
* Supprimer une tâche
* Afficher toutes les tâches

**Ajout d’une tâche**

URL:http://server:port/nomProjet/**todo/add**

Exemple: http://localhost:8080/examen/todo/add

Requête:le corps de la requête contient une représentation XML de la tâche à ajouter.

<tache>

<titre>sprint2</titre>

<date>15-1-2014</date>

<priorite>moyenne</priorite>

</tache>

Réponse:retourne une chaîne de caractères pour indiquer le succès ou l’échec de l’ajout.

Tâche ajoutée avec succès.

**Recherche des tâches selon une priorité**

URL: http://server:port/nomProjet/**todo/search/*[priorite]***

Exemple: http://localhost:8080/examen/todo/search/**haute**

Réponse:retourne une représentation XML de la liste des tâches ayant la priorité passée en paramètre.

<collection>

<tache>

<titre>backlog product</titre>

<date>25-12-2013</date>

<priorite>haute</priorite>

</tache>

<tache>

<titre>use cases</titre>

<date>10-12-2013</date>

<priorite>haute</priorite>

</tache>

<collection>

**Suppression d’une tâche**

#### URL: http://server:port/nomProjet/todo/delete?titre=*[valeur]*

Exemple: http://localhost:8080/examen/todo/delete?titre=**sprint3**

#### Réponse: retourne une représentation XML de la tâche supprimée.

<tache>

<titre>sprint3</titre>

<date>28-1-2014</date>

<priorite>basse</priorite>

</tache>

**Affichage de toutes les tâches**

URL:http://server:port/nomProjet/**todo**

Réponse:retourne une représentation XML de la liste des tâches.

<collection>

<tache>

<titre>sprint2</titre>

<date>15-1-2014</date>

<priorite>moyenne</priorite>

</tache>

<tache>

<titre>backlog product</titre>

<date>25-12-2013</date>

<priorite>haute</priorite>

</tache>

<collection>

**Travail demandé :**

#### Développez le service web RESTful en respectant les URLs indiquées dans l’énoncé.

#### Testez les opérations demandées avec le client REST de votre navigateur.

**Exercice 2 : JAX-WS**

L'objectif de cet exercice est de mettre en place un service web étendu pour déposer des annonces dans une agence immobilière.

1. A partir du fichier WSDL fourni ci-dessous, développez le service web correspondant avec les annotations JAX-WS ainsi que la classe entité nécessaire.

[<wsdl:definitions xmlns:xsd="**http://www.w3.org/2001/XMLSchema**" xmlns:wsdl="**http://schemas.xmlsoap.org/wsdl/**" xmlns:tns="**http://ws.immobiliere.agence.tn/**" xmlns:soap="**http://schemas.xmlsoap.org/wsdl/soap/**" xmlns:ns1="**http://schemas.xmlsoap.org/soap/http**" targetNamespace="**http://ws.immobiliere.agence.tn/**" name="**AgenceImmobiliereWSService**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

[<wsdl:types>](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

[<xs:schema xmlns:tns="**http://ws.immobiliere.agence.tn/**" targetNamespace="**http://ws.immobiliere.agence.tn/**" xmlns:xs="**http://www.w3.org/2001/XMLSchema**" version="**1.0**" elementFormDefault="**unqualified**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

<xs:element name="**deposerAnnonce**" type="**tns:deposerAnnonce**"/>

<xs:element name="**deposerAnnonceResponse**" type="**tns:deposerAnnonceResponse**"/>

[<xs:complexType name="**deposerAnnonce**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

[<xs:sequence>](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

<xs:element name="**bien**" type="**tns:bienImmobilier**" minOccurs="**0**"/>

</xs:sequence>

</xs:complexType>

[<xs:complexType name="**bienImmobilier**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

[<xs:sequence>](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

<xs:element name="**adresse**" type="**xs:string**" minOccurs="**0**"/>

<xs:element name="**description**" type="**xs:string**" minOccurs="**0**"/>

<xs:element name="**prix**" type="**xs:float**"/>

<xs:element name="**telProprietaire**" type="**xs:int**"/>

</xs:sequence>

</xs:complexType>

[<xs:complexType name="**deposerAnnonceResponse**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

[<xs:sequence>](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

<xs:element name="**estDepose**" type="**xs:boolean**"/>

</xs:sequence>

</xs:complexType>

</xs:schema>

</wsdl:types>

[<wsdl:message name="**deposerAnnonceResponse**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

<wsdl:part name="**parameters**" element="**tns:deposerAnnonceResponse**"/>

</wsdl:message>

[<wsdl:message name="**deposerAnnonce**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

<wsdl:part name="**parameters**" element="**tns:deposerAnnonce**"/>

</wsdl:message>

[<wsdl:portType name="**AgenceImmobiliereWS**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

[<wsdl:operation name="**deposerAnnonce**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

<wsdl:input name="**deposerAnnonce**" message="**tns:deposerAnnonce**"/>

<wsdl:output name="**deposerAnnonceResponse**" message="**tns:deposerAnnonceResponse**"/>

</wsdl:operation>

</wsdl:portType>

[<wsdl:binding name="**AgenceImmobiliereWSServiceSoapBinding**" type="**tns:AgenceImmobiliereWS**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

…

</wsdl:binding>

[<wsdl:service name="**AgenceImmobiliereWSService**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

[<wsdl:port name="**AgenceImmobiliereWSPort**" binding="**tns:AgenceImmobiliereWSServiceSoapBinding**">](http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS?wsdl)

<soap:address location="**http://localhost:8080/AgenceImmobiliereService/AgenceImmobiliereWS**"/>

</wsdl:port>

</wsdl:service>

</wsdl:definitions>

1. Développez le client permettant la consommation du service web développé.

**Exercice 3 : JAX-RS**

L'objectif de cet exercice est de mettre en place un service web RESTful pour la gestion d’une réservation d'un vol. La réservation d'un vol est caractérisée par un **id** (int), une **date de départ** (Date), une **date d'arrivée** (Date), un **lieu de départ** (String), un **lieu de destination** (String) et un **passager**.

Un passager est caractérisé par un **cin** (String), un **nom** (String), et un **prénom** (String).

On veut effectuer les opérations suivantes sur une réservation:

* Faire une réservation.
* Afficher toutes les réservations.
* Chercher les réservations selon le lieu de destination.
* Supprimer une réservation selon le cin d'un passager.

**Faire une réservation**

URL:http://server:port/nomProjet/**vol/reservation**

Requête:le corps de la requête contient une représentation XML de la réservation.

<reservation id="1">

<dateDepart>2014-11-27</dateDepart>

<dateArrivee>2014-11-28</dateArrivee>

<destination>tunisie</destination>

<passager>

<cin>123</cin>

<nom>salah</nom>

<prenom>ahmed</prenom>

</passager>

</reservation >

Réponse:retourne une chaîne de caractères pour indiquer le succès ou l’échec de la réservation.

Réservation effectuée avec succès.

**Affichage de toutes les réservations**

URL:http://server:port/nomProjet/**vol/affichage**

Réponse:retourne une représentation XML de la liste des réservations.

<collection>

<reservation id="1">

<dateDepart>2014-11-27</dateDepart>

<dateArrivee>2014-11-28</dateArrivee>

<destination>tunisie</destination>

<passager>

<cin>123</cin>

<nom>salah</nom>

<prenom>ahmed</prenom>

</passager>

</reservation>

</collection>

**Recherche des réservations selon le lieu de la destination**

URL: http://server:port/nomProjet/**vol/recherche?destination=*[valeur]***

Réponse:retourne une représentation XML de la liste des réservations ayant la destination passée en paramètre.

<collection>

<reservation id="1">

<dateDepart>2014-11-27</dateDepart>

<dateArrivee>2014-11-28</dateArrivee>

<destination>tunisie</destination>

<passager>

<cin>123</cin>

<nom>salah</nom>

<prenom>ahmed</prenom>

</passager>

</reservation>

< reservation id="2">

<dateDepart>2014-11-29</dateDepart>

<dateArrivee>2014-11-230</dateArrivee>

<destination>tunisie</destination>

< passager>

<cin>456</cin>

<nom>karima</nom>

<prenom>karima</prenom>

</passager>

</reservation>

</collection>

**Suppression d’une réservation selon le cin du passager**

#### URL: http://server:port/nomProjet/vol/*[valeur]*

Réponse: retourne une chaîne de caractères pour indiquer le succès ou l’échec de la suppression.

Réservation supprimée avec succès.

**Travail demandé :**

Développez le service web RESTful en respectant les URLs indiquées dans l’énoncé et en ajoutant les annotations adéquates.

**Questions de cours**

#### Commentez les différentes intératctions du scénario de consommation d’un service web.

#### 

#### 1 …………………………………………………………………………………………

#### 2 …………………………………………………………………………………………

#### 3 …………………………………………………………………………………………

#### 4 …………………………………………………………………………………………

#### 5 …………………………………………………………………………………………

#### Soit les classes java suivantes:

#### package tn.carrefour;

#### package tn.carrefour;

#### *@WebService*

#### public GestionProduits{

#### List<Produit> produits=new ArrayList<Produit>();

#### *@WebMethod*

#### public boolean ajouterProduit(Produit produit) {

#### return produits.add(produit);

#### }

#### public List<Produit> getProduits() {

#### return produits;

#### }

#### }

#### public class Produit {

#### private int code;

#### private String libelle;

#### private String description;

#### public int getCode() { return code; }

#### *@XmlAttribute(name="id")*

#### public void setCode(int code) {this.code = code; }

#### public String getLibelle() { return libelle; }

#### *@XmlElement*

#### public void setLibelle(String libelle) { this.libelle = libelle; }

#### public String getDescription() { return description; }

#### *@XmlElement*

#### public void setDescription(String description) { this.description = description; }

#### }

#### Combien d'éléments <message> existe-t-il dans le contrat WSDL relatif à ce service web?

#### …………………………………………………………………………………………………………

#### En essayant de consommer l'opération getProduit du service web exposé, l'exception suivante s'est produite: « *unable to marshal type "tn.carrefour.Produit" as an element* »

#### Expliquez la cause de l'erreur et proposez une solution pour y remédier.

#### ………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

#### Complétez les parties manquantes du contrat WSDL.

#### <wsdl:definitions

#### xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"

#### xmlns:crf="http://www.webserviceX.NET/"

#### xmlns:xs="http://www.w3.org/2001/XMLSchema"

#### xmlns:wsdl="http://schemas.xmlsoap.org/wsdl"

#### targetNamespace="http://www.webserviceX.NET/">

#### <xs:element name="getProduitsResponse" type="1:getProduitsResponse"/>

#### <xs:complexType name="getProduitsResponse">

#### 2

#### </xs:complexType>

#### <xs:element name="produit" type="crf:produitType"/>

#### <xs:complexType name="produitType">

#### 3

#### </xs:complexType>

#### </xs:schema>

#### 

#### <wsdl:message name="getProduitsResponse">

#### <wsdl:part element="crf:getProduitsResponse" name="parameters"> </wsdl:message>

#### 

#### <wsdl:portType name="GestionProduits">

#### <wsdl:operation name="getProduits">

#### <wsdl:input message="crf:getProduits" />

#### <wsdl:output message="crf:getProduitsResponse" />

#### </wsdl:operation>

#### </wsdl:portType>

|  |  |
| --- | --- |
| 1 |  |
| 2 | …………………………………………………………………………………………………………………………………………………………………………………………. |
| 3 | …………………………………………………………………………………………….…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………… |