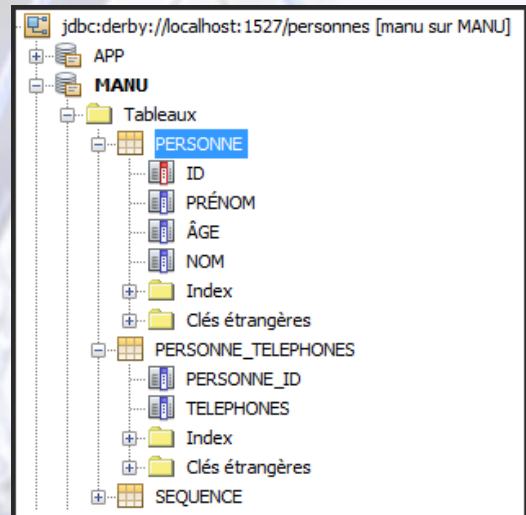


x JAVA EE 6

Au travers de ce TP, nous allons découvrir comment transiter un bean entité au travers d'un Web service en passant par un bean session intermédiaire.

x VISUALISATION DES RÉSULTATS



x BEANS ENTITÉ

```

F:\BTS IRIS\TP Java\ModuleWebService\src\java\entité\Personne.java

package entité;

import java.io.Serializable;
import java.util.*;
import javax.persistence.*;
import javax.xml.bind.annotation.XmlRootElement;

@XmlRootElement
@Entity
@NamedQuery(name="toutes", query="SELECT personne FROM Personne AS personne ORDER BY personne.nom")
public class Personne implements Serializable {
    @Id
    @GeneratedValue(strategy=GenerationType.AUTO)
    private long id;
    private String nom;
    private String prénom;
    private int âge;
    @ElementCollection(fetch=FetchType.EAGER)
    private List<String> telephones = new ArrayList<String>();

    public long getId() { return id; }
    public void setId(long id) { this.id = id; }

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

    public String getPrénom() { return prénom; }
    public void setPrénom(String prénom) { this.prénom = prénom; }

    public int getÂge() { return âge; }
    public void setÂge(int âge) { this.âge = âge; }

    public void setTelephones(List<String> telephones) { this.telephones = telephones; }
    public List<String> getTelephones() { return telephones; }

    public Personne() {}

    public Personne(String nom, String prénom, int âge) {
        this.nom = nom;
        this.prénom = prénom;
        this.âge = âge;
    }
}
    
```

Annotation qui permet de réaliser le parsing entre un document XML et le bean entité correspondant.

Doit être placé en premier.

Il est également impératif d'avoir systématiquement tous les getter et setter (toutes les propriétés) de tous les attributs si vous désirez qu'ils transitent sur le réseau.



x BEANS SESSION

F:\BTS IRIS\TP Java\ModuleWebService\src\java\session\GestionRemote.java

```
package session;

import javax.ejb.Remote;
import entité.Personne;
import java.util.List;

@Remote
public interface GestionRemote {
    void nouveau(Personne personne);
    void modifier(Personne personne);
    void supprimer(long identification);
    List<Personne> listePersonnes();
    Personne rechercher(long identification);
}
```

F:\BTS IRIS\TP Java\ModuleWebService\src\java\session\Gestion.java

```
package session;

import java.util.List;
import javax.ejb.Stateless;
import javax.persistence.*;
import entité.Personne;

@Stateless
public class Gestion implements GestionRemote {
    @PersistenceContext
    private EntityManager stocker;

    public void nouveau(Personne personne) {
        stocker.persist(personne);
    }

    public void modifier(Personne personne) {
        stocker.merge(personne);
    }

    public List<Personne> listePersonnes() {
        return stocker.createNamedQuery("toutes").getResultList();
    }

    public Personne rechercher(long identification) {
        return stocker.find(Personne.class, identification);
    }

    public void supprimer(long identification) {
        stocker.remove(rechercher(identification));
    }
}
```





x WEB SERVICE

F:\BTS IRIS\TP Java\ModuleWebService\src\java\ws\GestionPersonneWS.java

package ws;

```
import entité.Personne;
import java.util.List;
import javax.jws.*;
import javax.ejb.*;
import session.GestionRemote;
```

```
@WebService()
@Stateless()
public class GestionPersonneWS {
    @EJB
    private GestionRemote ejbRef;

    @WebMethod(operationName = "nouveau")
    @Oneway
    public void nouveau(@WebParam(name = "personne") Personne personne) {
        ejbRef.nouveau(personne);
    }

    @WebMethod(operationName = "modifier")
    @Oneway
    public void modifier(@WebParam(name = "personne") Personne personne) {
        ejbRef.modifier(personne);
    }

    @WebMethod(operationName = "supprimer")
    @Oneway
    public void supprimer(@WebParam(name = "identification") long identification) {
        ejbRef.supprimer(identification);
    }

    @WebMethod(operationName = "listePersonnes")
    public List<Personne> listePersonnes() {
        return ejbRef.listePersonnes();
    }

    @WebMethod(operationName = "rechercher")
    public Personne rechercher(@WebParam(name = "identification") long identification) {
        return ejbRef.rechercher(identification);
    }
}
```



x PERSONNE GÉNÉRÉ AUTOMATIQUEMENT PAR WSIMPORT CÔTÉ CLIENT

F:\BTS IRIS\TP Java\ClientWebService\build\generated-sources\jax-ws\ws\Personne.java

```

package ws;

import java.util.*;
import javax.xml.bind.annotation.*;

@XmlAccessorType(XmlAccessType.FIELD)
@XmlType(name = "personne", propOrder = {
    "id",
    "nom",
    "pr\u00e9nom",
    "telephones",
    "\u00e2ge"
})
public class Personne {

    protected long id;
    protected String nom;
    protected String prénom;
    @XmlElement(nillable = true)
    protected List<String> telephones;
    protected int âge;

    public long getId() { return id; }
    public void setId(long value) { this.id = value; }

    public String getNom() { return nom; }
    public void setNom(String value) { this.nom = value; }

    public String getPrénom() { return prénom; }
    public void setPrénom(String value) { this.prénom = value; }

    public List<String> getTelephones() {
        if (telephones == null) {
            telephones = new ArrayList<String>();
        }
        return this.telephones;
    }

    public int getÂge() { return âge; }
    public void setÂge(int value) { this.âge = value; }

    public Personne() { }

    public Personne(String nom, String prénom, int âge) {
        this.nom = nom;
        this.prénom = prénom;
        this.âge = âge;
    }

    @Override
    public String toString() {
        return nom + ' ' + prénom;
    }

    public void modifier(String nom, String prénom, int âge) {
        this.nom = nom;
        this.prénom = prénom;
        this.âge = âge;
    }
}

```

Attention, les accents ne sont peut-être pas pris en compte, il faut alors apporter les corrections nécessaires.

Partie du code à rajouter pour que cela soit plus convivial côté client.



x APPLICATION CLIENTE EN CONNEXION AVEC LE WEB SERVICE

F:\BTS IRIS\TP Java\ClientWebService\src\clientwebservice\Client.java

```

package clientwebservice;

import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.*;
import ws.*;

public class Client extends JFrame {
    private JToolBar outils = new JToolBar();
    private JTextField nom = new JTextField();
    private JTextField prénom = new JTextField();
    private JFormattedTextField âge = new JFormattedTextField(0);
    private JComboBox téléphones = new JComboBox();
    private JPanel panneau = new JPanel();
    private JComboBox liste = new JComboBox();
    private static GestionPersonneWS gestion;
    private Personne personne = new Personne();
    private boolean effacer = true;

    public Client() {
        super("Personne");
        add(outils, BorderLayout.NORTH);
        outils.add(new AbstractAction("Enregistrer") {
            public void actionPerformed(ActionEvent e) {
                Personne personne = new Personne(nom.getText(), prénom.getText(), (Integer)âge.getValue());
                for (int i=0; i<téléphones.getItemCount(); i++)
                    personne.getTelephones().add((String) téléphones.getItemAt(i));
                gestion.nouveau(personne);
                listingPersonnes();
            }
        });
        outils.add(new AbstractAction("Ajout téléphone") {
            public void actionPerformed(ActionEvent e) {
                String téléphone = (String) téléphones.getSelectedItem();
                téléphones.addItem(téléphone);
            }
        });
        outils.add(new AbstractAction("Modifier") {
            public void actionPerformed(ActionEvent e) {
                personne.modifier(nom.getText(), prénom.getText(), (Integer)âge.getValue());
                gestion.modifier(personne);
                listingPersonnes();
            }
        });
        outils.add(new AbstractAction("Supprimer") {
            public void actionPerformed(ActionEvent e) {
                gestion.supprimer(personne.getId());
                listingPersonnes();
            }
        });
        outils.add(new AbstractAction("Effacer") {
            public void actionPerformed(ActionEvent e) {
                personne = new Personne();
                rafraichir();
            }
        });
        panneau.setLayout(new GridLayout(4, 2));
        panneau.add(new JLabel("Nom :"));
        panneau.add(nom);
        panneau.add(new JLabel("Prénom :"));
        panneau.add(prénom);
        panneau.add(new JLabel("Âge :"));
        panneau.add(âge);
        panneau.add(new JLabel("Téléphones :"));
        panneau.add(téléphones);
        téléphones.setEditable(true);
        add(panneau);
        add(liste, BorderLayout.SOUTH);
        liste.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                if (effacer) {
                    personne = (Personne) liste.getSelectedItem();
                    rafraichir();
                }
            }
        });
        listingPersonnes();
        pack();
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setVisible(true);
    }

    private void listingPersonnes() {
        effacer = true;
        liste.removeAllItems();
        effacer = false;
        for (Personne personne : gestion.listePersonnes())
            liste.addItem(personne);
    }

    private void rafraichir() {
        nom.setText(personne.getNom());
        prénom.setText(personne.getPrénom());
        âge.setValue(personne.getÂge());
        téléphones.removeAllItems();
        for (String téléphone : personne.getTelephones()) téléphones.addItem(téléphone);
    }

    public static void main(String[] args) {
        gestion = new GestionPersonneWSService().getGestionPersonneWSPort();
        new Client();
    }
}

```

