```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package metier.modele;

import java.util.ArrayList;
import java.util.Date;
import java.util.List;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.ManyToMany;
import javax.persistence.OneToMany;
import javax.persistence.Temporal;

/**
 *
 * @author Administrateur
 */
@Entity
public class Conseiller {

    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private Integer idConseiller;
    private String civilite;
    private String nom;
    private String prenom;
    @Temporal(javax.persistence.TemporalType.DATE)
    private Date dateNaissance;
    private String adresse;
    private String telephone;
    private String email;
    @ManyToMany(mappedBy = "conseillers")
    List<Pays> paysConseilles = new ArrayList();
    @OneToMany
    List<Client> clients = new ArrayList();

    public Conseiller() {
    }
```

```
    public Conseiller(String civilite, String nom, String prenom,

            Date dateNaissance, String adresse, String telephone,
            String email) {
        this.civilite = civilite;
        this.nom = nom;
        this.prenom = prenom;
        this.dateNaissance = dateNaissance;
        this.adresse = adresse;
        this.telephone = telephone;
        this.email = email;
    }

    public Integer getIdConseiller() {
        return idConseiller;
    }

    public String getCivilite() {
        return civilite;
    }

    public String getNom() {
        return nom;
    }

    public String getPrenom() {
        return prenom;
    }

    public Date getDateNaissance() {
        return dateNaissance;
    }

    public String getAdresse() {
        return adresse;
    }

    public String getTelephone() {
        return telephone;
    }

    public String getEmail() {
        return email;
    }
```

```
    public List<Client> getClients() {

        return clients;
    }

    public void setCivilite(String civilite) {
        this.civilite = civilite;
    }

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

    public void setPrenom(String prenom) {
        this.prenom = prenom;
    }

    public void setDateNaissance(Date dateNaissance) {
        this.dateNaissance = dateNaissance;
    }

    public void setAdresse(String adresse) {
        this.adresse = adresse;
    }

    public void setTelephone(String telephone) {
        this.telephone = telephone;
    }

    public void setEmail(String email) {
        this.email = email;
    }

    @Override
    public String toString() {
        return "Conseiller{" + "civilite=" + civilite + ", nom=" + nom
                + ", prenom=" + prenom + ", dateNaissance=" + dateNaissance
                + ", adresse=" + adresse + ", telephone=" + telephone
                + ", email=" + email + creeListePays() + '}';
    }

    public String creeListePays() {
```

```
        String chaine = "";
        for (int i = 0; i < this.paysConseilles.size(); i++) {


            chaine += ", pays" + i + " "
                    + this.paysConseilles.get(i).getNom();
        }
        return chaine;
    }

    @Override
    public boolean equals(Object obj) {
        if (obj == null) {
            return false;
        }
        if (getClass() != obj.getClass()) {
            return false;
        }
        final Conseiller other = (Conseiller) obj;
        if ((this.civilite == null) ? (other.civilite != null)
                : !this.civilite.equals(other.civilite)) {
            return false;
        }
        if ((this.nom == null) ? (other.nom != null)
                : !this.nom.equals(other.nom)) {
            return false;
        }
        if ((this.prenom == null) ? (other.prenom != null)
                : !this.prenom.equals(other.prenom)) {
            return false;
        }
        if (this.dateNaissance != other.dateNaissance
                && (this.dateNaissance == null
                || !this.dateNaissance.equals(other.dateNaissance))) {
            return false;
        }
        if ((this.adresse == null) ? (other.adresse != null)
                : !this.adresse.equals(other.adresse)) {
            return false;
        }
        if ((this.telephone == null) ? (other.telephone != null)
                : !this.telephone.equals(other.telephone)) {
            return false;
```

```
        }
        if ((this.email == null) ? (other.email != null)
                : !this.email.equals(other.email)) {
            return false;
        }

        return true;
    }

    public void addPays(Pays pays) {
        this.paysConseilles.add(pays);
    }

    public void addClient(Client client) {
        this.clients.add(client);
    }
}
```