

```

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

import java.util.ArrayList;
import java.util.GregorianCalendar;
import java.util.Iterator;
import java.util.List;
import javax.persistence.CascadeType;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.ManyToMany;
import javax.persistence.ManyToOne;
import javax.persistence.OneToMany;
import javax.persistence.Temporal;
import javax.persistence.TemporalType;

/**
 *
 * @author Administrateur
 */
@Entity
public class Client
{
    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private int numClient;
    private String nom;
    private String prenom;
    private String adressePostale;
    private String email;
    private String tel;
    @Temporal(TemporalType.DATE)
    private GregorianCalendar dateNaissance;
    @ManyToOne
    private SigneAstrologique signeAstrologique;
    @ManyToOne(cascade =
    {
        CascadeType.MERGE, CascadeType.REFRESH
    })
    private Employe referent;
    @ManyToMany
    private List<Medium> mediumsFavoris;
    @OneToMany(cascade = CascadeType.REMOVE, mappedBy = "leClient")
    private List<Horoscope> horoscopes;

    public Client()
    {
        mediumsFavoris = new ArrayList<Medium>();
    }

    public Client(String nom, String prenom, String adressePostale,
        String email, String tel, GregorianCalendar dateNaissance,
        SigneAstrologique signe, List<Medium> mediums, Employe referent)
    {
        this.nom = nom;
        this.prenom = prenom;
        this.adressePostale = adressePostale;
        this.email = email;
        this.tel = tel;
        this.dateNaissance = dateNaissance;
        this.signeAstrologique = signe;
        this.mediumsFavoris = mediums;
        this.referent = referent;
    }
}

```

```
        this.horoscopes = new ArrayList<Horoscope>();
    }

    @Override
    public String toString()
    {
        StringBuilder contenu = new StringBuilder();
        contenu.append(nom);
        contenu.append(" ");
        contenu.append(prenom);
        contenu.append("\n");
        contenu.append(adressePostale);
        contenu.append("\n");
        contenu.append("Votre numéro de client : ");
        contenu.append(numClient);
        contenu.append("\n");
        contenu.append("Votre signe astrologique : ");
        contenu.append(signeAstrologique.getNom());
        contenu.append("\n");
        contenu.append("Vos Mediums favoris : ");

        Iterator<Medium> it = mediumsFavoris.iterator();

        Medium medium = null;
        while(it.hasNext())
        {
            medium = it.next();
            contenu.append(medium.getNom());
            if(it.hasNext())
            {
                contenu.append(", ");
            }
        }

        return contenu.toString();
    }

    public List<Medium> getMediumsFavoris()
    {
        return mediumsFavoris;
    }

    public void setMediumsFavoris(List<Medium> mediumsFavoris)
    {
        this.mediumsFavoris = mediumsFavoris;
    }

    public String getAdressePostale()
    {
        return adressePostale;
    }

    public GregorianCalendar getDateNaissance()
    {
        return dateNaissance;
    }

    public String getEmail()
    {
        return email;
    }

    public String getNom()
    {
        return nom;
    }

    public int getNumClient()
    {

```

```
    return numClient;
}

public String getPrenom()
{
    return prenom;
}

public String getTel()
{
    return tel;
}

public void setAdressePostale(String adressePostale)
{
    this.adressePostale = adressePostale;
}

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

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

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

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

public void setTel(String tel)
{
    this.tel = tel;
}

public Employe getReferent()
{
    return referent;
}

public void setReferent(Employe referent)
{
    this.referent = referent;
}

public SigneAstrologique getSigneAstrologique()
{
    return signeAstrologique;
}

public void setSigneAstrologique(SigneAstrologique signeAstrologique)
{
    this.signeAstrologique = signeAstrologique;
}

/**
 * A utiliser avec précaution, compare juste la date de naissance
 * @param autreClient
 * @return vrai si la date de naissance a été modifié, faux sinon
 */
public boolean isBirthModified(Client autreClient)
```

```
{
    if (dateNaissance.equals(autreClient.getDateNaissance()))
    {
        return false;
    }
    else
    {
        return true;
    }
}

public List<Horoscope> getHoroscope()
{
    return horoscopes;
}

public void addHoroscope(Horoscope unHoroscope)
{
    horoscopes.add(unHoroscope);
}
}
```