Problème import java.time.LocalDate;

```
package Modèle;
Public class Contact {
  Private int id; num,
  Private String nom;
  Private int Num_tel;
  Private double solde;
  Public Contact (int id, String nom, int Num_tel, double solde) {
      this.id = id; num num
      this.Num_tel = Num_tel ;
      this.solde = solde;
      this.nom = nom; }
Public class Appels {
  Private int id;
  Private String LocalDate date;
  Private double durée;
  Public Appels (int id, String LocalDate date, double durée) {
      this.id = id;
      this.date = dte ;
      this.durée = durée ; }
Public class Appel_Entrant extends Appels {
  Private double coût ;
  Public Appel_Entrant (int id, LocalDate date, double durée,
                        double coût) {
      Super (id, LocalDate, durée)                    (1)
```

```
Public class AppelUrgent extends Appel {
    Private double coût;
    Public AppelUrgent(int id, LocalDate date, double duré, double coût
    { super(id, LocalDate, duré) coût;
      this.coût = coût; }
}
```

```
Packag dao;
Public Icontact interface IContact {
    public void saveContact(Contact c);
    public void saveAppelEmi(Appel ap, Contact c);
    public Contact getContact(int num);
    public List<Contact> getAllContact();
    public double coutAppel(int num){}; }
Public interface.
```

~~Public interface Contact~~

Public class ContactImp implements IContact{

Private List<Contact> Cont = new ArrayList<Contact>();

connection con;

Public ContactImp(){

try{

Class.forName("com.mysql.jdbc.Driver");

String url = "jdbc:mysql://localHost:3306/gestionAppel",

Con = DriverManager.getConnection(url, "root", "root");

}catch(exception e){

System.out.println(e); } }

Public void saveContact(Contact c){

String req = "insert into Contact (num, nom, numTel, solde)";

try{ PreparedStatement Ps = con.prepareStatement(req);

Ps.setInt(1, c.getNum());

Ps.setString(2, c.getNom());

Ps.setString(3, c.getNumTel());

Ps.setDouble(4, c.getSolde()); (4)

Ps.executeUpdate();

}catch(Exception e){ e.PrintStackTrace(); }

```
Public void SaveAppelEm(Appel ap, Contact c) {
String req = "insert into appelem(num, dateappel, durree, numcont)
             values(?,?,?,?)";
Try{ PreparedStatement ps = Con.PrepareStatement(req);
   Ps.SetInt(1,num);
   Ps.setString(2,dateappel);
   Ps.setDouble(3,durree);
   Ps.
   Ps.setInt(1, ap.getnum());
   Ps.setString(2, ap.getDateAppel());
   Ps.setDouble(3, ap.getDuree());
   Ps.setInt(4, C.getnumcont());
   Ps.executeUpdate();
} Catche(Exception e){
   e.printStackTrace();}}
Public Contact getContact(int num){
  //String req = "Select * from Contact where numcp ?"
  Try{ PreparedStatement ps = Con.PrepareStatement(req);
       Ps.setString(1, num);
       ResultSet res = Ps.executeQuery();
```

```
while (res.next()){
    int contactId = res.getInt("num");
    String contactName = res.getString("nom");
    String phonNum = res.getString("numtel");
    double S = res.getDouble("solde");
    Contact = new Contact(contactId, contactName, phonNum, S);
}
return Contact;
public List<Contact> getAllContact(){
try{
    java.sql.Statement stmt = Con.CreateStatement();
    ~~ResultSet res = stmt.executeQuery();~~
    String req = "Select * from Contact";
    ResultSet res = stmt.executeQuery(req);
    while(res.next()){
        Contact C = new Contact(res.getInt(1),
        res.getNom(2), res.getInt(3), res.getDouble(4))
        Cont.add(C);}}
    catch(Exception e){
        e.PrintStackTrace();}
    return Cont;}
```

```
public double coutAppel (int numC){
    double coutTotal = 0;
    try{
        String req = "Select * from appelem where numcont = ?";
        PreparedStatement stmt = con.PrepareStatement (req);
        ResultSet res = stmt.executeQuery();
        stmt.SetInt(1, numC);
        while(res.next()){
            double duree = resultSet.getDouble("duree");
            coutTotale += duree * 2 }
        resultSet.close();
    } catch (SQLException e){
        e.PrintStackTrace(); }
        return coutTotale;
    }
```

1) public class AjoutContact implements ActionListener {

public AjoutContact () {

f. setLayout (new GridLayout (4, 2));

f.add (lnom);

f.add (txnom);

f.add (lnum);

f.add (txtnum);

f.add (ls);

f.add (txs);

f.add (ba);

f.add (bq);

ba.addActionListner (this);

bq.addActionListner (this);

f.setVisible (true); }

public static void main (String [] args) {

IContact contactD = new ContactImp ();

Contact contact1 = new Contact (1, "Ali", "123456789", 100.0);

contactD.saveContact (contact1);

AppelEmi appelEmi1 = new AppelEmis (1, "01/01/2023", 10.0, 1);

```
AppelRecu appelRecu1 = new AppelRecu(2, "02/01/2023
                                    5.0, 1);
contactD.saveAppelRecu(appelRecu1, contact1);
Contact contactConsulte = contactD.getContact("01234567899
System.out.println("Contact consulté:" + contactConsulte);
List<Contact> contacts = contactD.getAAContact();
System.out.println("Liste des contacts:");
for (Contact contact: contacts){
    System.out.println(contact);}
double coutTotalAppels = contactD.coutAPel(1);
System.out.println("( ___", + coutTotal);
}.
Public void ActionPerformed(ActionEvent e){
IContact C1 = new ContactIM();
if(e.getSource() == ba){
String nom = txtnom.getText();
String numtel = txtnumtel.getText();
Double solde = txtsolde.getDouble();
Contact C1 = new Contact(nom, numtel, solde);}
if(e.getSource() == bq){
    System.exit(0);}}
```

```
Public class SuiviSolde implements ActionListner {
    Public suiVitSolde (){
    br.addActionListener(this);}
Public void ActionPerformed(ActionEvent e){
    if(e.getSource()==br)
```