



RAPPORT SEMESTRIEL N°2 : POO

19-05-2018

*PROJET SUR LA GESTION DES APPELS TELEPHONIQUES DANS
UN OPERATEUR TELEPHONIQUE*

MI. Lic 2 Acad/Section A/Groupe 2

Encadré par :

Madame S.BOUKHADOUMA-
chargé de cours et de td POO

Madame OUAZAR-chargé de tp

Réalisé par :

AKLI Yassamine && MEDDAS
Nesrine

TABLE DES MATIERES

Contenu

Introduction	1
Description des classes avec leurs schéma	2
Présntation de l'application	13
Conclusion	19

INTRODUCTION

Introduction

Ce projet consiste à réaliser une simulation d'une application réel qui gère les clients et appels /sms d'un opérateur téléphonique ce petit logiciel est établit en l'usage de l'héritage, exceptions,

DESCRIPTION DES CLASSES AVEC LEURS SHEMA

Description des classes avec leurs shéma

SHEMA CONCEPTUEL DES CLASSES IMPLEMENTEE :

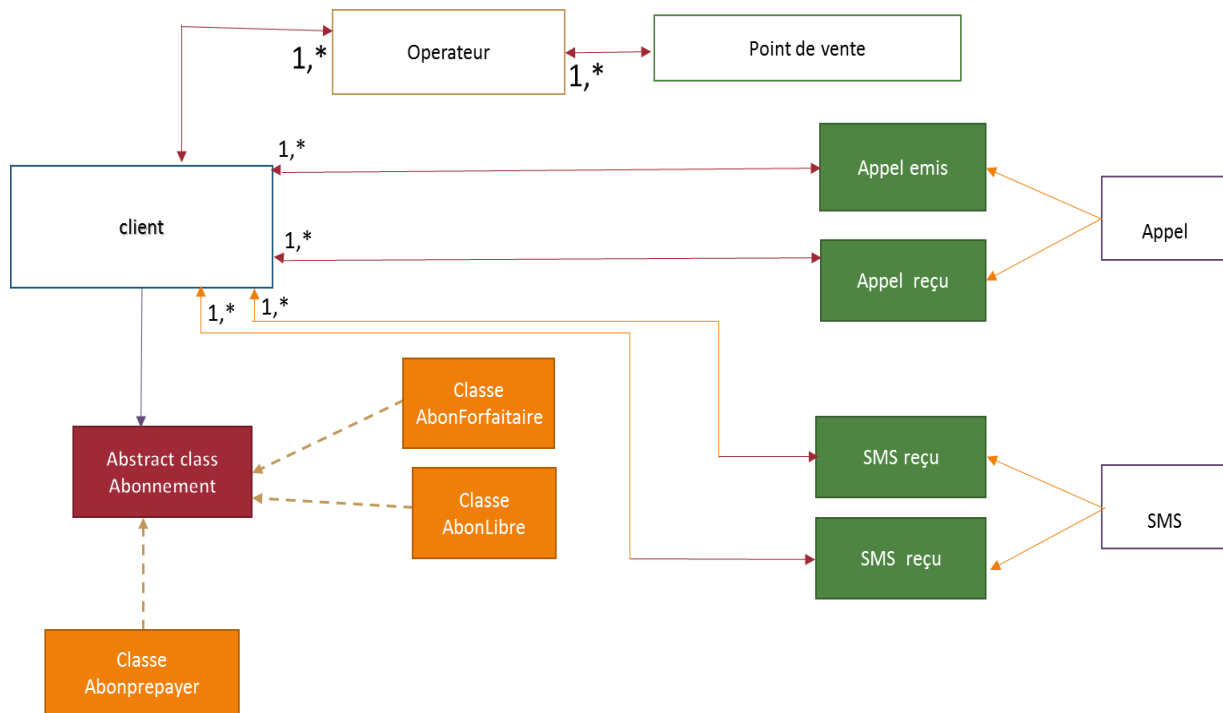


Figure 1 Modélisation des classe avec leurs associations

Lgende :
 : Appel emis hérite de Appel : Client a une variable de classe de type Abonnement ,
 : Operateur a plusieurs variables de classe de type Client/point de vente

DESCRIPTION DES CLASSES

- Client :

DESCRIPTION DES CLASSES AVEC LEURS SHEMA

```
Abonnement :Type;  
/*attribut*/String:nom,prénom;AdresseMail:ma  
il;  
Adresse:Adress;Set<SMSEmis>SMSsortant;  
Set<SMSReçu>SMSentrant;  
Set<AppelEmis>APPELsortant; //de type  
HashSet Set<AppelReçu>APPELentrant;  
String NumTel;String numéroContrat; Date  
dateContrat;
```

//constructeur avec parametre:

```
Client(String nom, String prénom, String numTelephone,Date d,AdresseMail mail,Adresse adr,  
Abonnement A)
```

throws NegativeException, TelephoneExceptions, DateException, AdresseException

//constructeur sans parametre:

```
Client() );
```

//methode saisir et affichage et to String:

```
Saisir()(void)-Client;
```

```
Affichage()(void)-Client;
```

```
toString()(String)-Client;
```

```
AfficherAppel()(void)-Client
```

DESCRIPTION DES CLASSES AVEC LEURS SHEMA

/*-----appel: dans ces méthodes on a opter a vérifier les dates; l'appel doit être lancer après la date de contrat et la date d'abonnement---si la date d'appel vient pendant ou après la date fin d'abonnement de blocage alors le client sera bloquée avec stBloqué

si client est bloqué=>envoyé null+lancer une exception « client bloqué impossible d'effectuer cet appel » -----meme principe pour AppelSortant et SMSEntrant et SMSSortant*/

/*-----la vérification du solde s'implémente dans la méthode AjouterAppelSortant et AjouterSMSSortant via l'abonnement=>ces methodes sont appelé pour crée l'appel ou sms sortant et récupéré le retour dans un objet appel crée dans ces methode qui suivent (c'est lui-même qui sera AjouterAppelEntrant(String num, Date datAppel, Heure

HeureAppel, int durée):(AppelEntrant)-Client;

AjouterAppelSortant(String num, Date datAppel, Heure HeureAppel, int durée):(AppelSortant)-Client

AjouterSMSEntrant(String num, Date datAppel, Heure HeureSMS,String text):(SMSEntrant)-Client

AjouterSMSSortant(String num, Date datAppel, Heure HeureSMS, Strnig text):(SMSSortant)-Client

//-----ses méthodes sont appelé dans les methodes:

//-----: AjouterAppelsEmis, AjouterAppelRecu,

/*AjouterSMSEmis, AjouterSMSRecu; permettent d'ajouter un appel ou sms dans la collection correspondante ; ses méthodes verifie d'abord la validitée du numéro donnée si la date de blocage est atteinte alors on envoie trois sms entrant à des dates différente , puis on bloque le client , sinon on ajoute exemple :(this,A,add(this,AjouterApellEntrant(,,,,));**des exceptions peuvent être levé au cas de blocage, solde insuffisant ou date invalide elle seront catché dans cette methode*/

DESCRIPTION DES CLASSES AVEC LEURS SHEMA

- Operateur :

```
String nom;  
//ensemble de point de vente:  
ArrayList <PointDeVente> ptrVente=new ArrayList <PointDeVente>();  
//pourcentage de couverture par wilaya  
long pourcentage[] = new long[49];  
//ensembles de clients abonnées:  
ArrayList <Client> Abonnées=new ArrayList <Client>();
```

```
Operateur()-Operateur;//constructeuru  
  
Operateur(String nomoperatur, ArrayList<PointDeVente> ptrVente, ArrayList<Client>  
abonnées)-Operateur  
  
Operateur(String nom)-Operateur;  
  
toString()(String)-Operateur; /*renvoie la description de l'objet operateur avec ses abonnées  
et wilaya avec leurs pourcentage et points de ventes*/  
  
//---Affichage:  
  
AfficherPointDeVente()(void)-Operateur//afficher les //points de ventes  
  
//pour afficher les client triée par type d'abonnement: //ajouter les abonnées dans setAbon  
puis on affiche  
  
AfficherClientParTypeAbonnement()(void)-Oprateur;  
  
//si client bloqué on affiche :  
  
AfficherClientBloqué()(void)-Operateur;  
  
//Afficher les clients non bloquée :  
  
AfficherClientNonBlqué  
  
//afficher les wilaya avec leur pourcentage:  
  
AfficherWilayaPourcentage()(void)-Operateur;
```

DESCRIPTION DES CLASSES AVEC LEURS SCHEMA

```
pourcentageModifier(int numWilaya,long nouvPourcentage)(void);/*la verification de la validité du
numéro de la wilaya est en dehors de cette méthode */

SupprimerClient(Client c)(void);/*supprimer un client on suppose qu'il existe_ le contrôle de
l'existence est effectué en dehors de ces méthodes*/

SupprimerPtrVente(PointDeVente ptrVente)(void);

/*on suppose que les objets à ajouté n'existe pas*/

AjouterPtrVente(PointDeVente p)(void);

//-----les setters et getters

//-----Methode de recherche:

/*rechercher un numero donné et retourner l'objet si il existe null sinon*/

RechercherNumeroDonnéeClient(String numéro)(Client)-Operateur;

/*rechercher un numero donné si on le trouve on retourne son indice pour pouvoir le modifier
plutart et effectué les traitements qu'on veut*/

getMemeNuméroEmplacementPtrVente(String num)(int)-Operateur;

//rechercher un numero donné

RechercherNumeroDonnésPointDeVente(String s)(PointDeVente)-Operateur;

// retourne l indice du numéro du point de vente:

getMemeNuméroEmplacement(String num)(int );

//-----methodes de remplissage:

//-----REEMPLISSAGE CLIENT:

RemplissageClients()(void)-Operateur;

//-----REEMPLISSAGE POINT DE VENTE:

RemplissagePointDeVente() ( void)-Operateur;
```


DESCRIPTION DES CLASSES AVEC LEURS SHEMA

- Point de vente :

```
//Attribut
String nomAgence;
Adresse adresse;
Type typ;
String numeroTelephone

//-----
PointDeVente()-PointDeVente;//constructeur

PointDeVente(String nomAgence, Adresse adresse,
Type typ, String numeroTelephone) –Opérateur; /*si
le numéro entrée est invalide une exception ssera
lance*/

//----getter et setter

//----method saisit():

Saisir()(Void)-PiontDeVente;

toString()(String)-PiontDeVente;
```

- Appel/SMS :

DESCRIPTION DES CLASSES AVEC LEURS SHEMA

- Abonnement :

Abonnement (abstract classe) :

```
protected Date dateDebAbon;

protected int durée;//minute

// (solde):à consommé(cumulé pour abnnement libre / decrements pour les autre)

protected double soldeMontant;

Abonnement()-Abonnement;//constructeur sans //parameter

Abonnement(Date dateDebAbon, int durée, double soldeMontant)-Abonnement;/*constructeur
avec paramètre : envoie deux exceptions en cas d'erreur throws DateException(date invalide),
NegativeException (durée ou montant négative)*/

abstract void Saisir();-Abonnement;

toString()(String)-Abonnement;//envoie la date debut d'abonnement

Afficher()(void)-Abonnement;//affiche la description de l'objet via sa référence

abstract verifierAtteinteDateFinAbon()(Date);//envoie la date limite de l'abonnement

abstract verifierAtteinteDateFBloqué();//envoie la date limite de blocage de l'abonnement

//-getter et setter

//*****Méthodess pour Ajouter appel et calcul facture*****

isBloqué(Date d)(boolean)-Abonnement ;/*on comparant la date donnée supposant la date actel
avec la date limite de blocage si elle est atteinte ou la dépasse alors on renvoie vrai sinon faaux*/
abstract AjouterAppel(Client client, String s)(void) -Abonnement;
abstract AjouterAppel(Client client, String s, Date D,Heure h, int durée )(Appel) throws
SoldeException, DateException-Abonnement ;
abstract AfficherChoixAbonnement()(int);/*affiche un menu pour choisir les differents propositions
que donne l'opérateur à son client*/

Abstract Facture(Client c):void-Abonnement; ,//afficher la facture (description)
```

DESCRIPTION DES CLASSES AVEC LEURS SHEMA

Abonnement Libre :

```
//attribut ; private :  
final double montantFixe=500;  
final int TVA= 19;  
int duréeFixe;//on en a besoin pour le renouvellement  
        private int duréePourBloqué;/*cette durée est définit dans le constructeur comme étant :  
duréePourBloqué=durée+durée;*/  
  
//constructeur sans paramètre :  
AbonLibre () –AbonLibre ;  
  
//constructeur avec paramètres :  
AbonLibre (Date dateDebAbon, int durée, double soldeMontant ) throws DateException,  
NegativeException-AbonLibre ;//ici soldeMontant est=à 0  
  
Saisir()(void)-AbonLibre ;  
  
Afficher()(void)-AbonLibre ;/*appelé super.Afficher() puis donné la description de l'objet AbonLivre  
via sa référence */  
  
toString()(String ) ;//envoi : « Type Abonnement , Date Fin Abonnement , solde cumulée »  
  
verifierSoldeRestant()(Boolean) –AbonLibre//on n'en aura pas besoin ; verifier que les dates  
verifierAtteinteDateFinAbon()(Date )-AbonLibre;/*renvoie la date après duréefixe de la (Date Debut  
Abonnement;*/  
  
Facture(Client c):void-AbonLibre;/* affiche la date de blocage, description du client, u type  
d'abonnement, des listes d'appels/sms effectué reçu, et le solde à payee*/  
  
//ajouter appel  
  
AjouterAppel(Client client, String s, Date D, Heure h, int durée) : Appel-AbonLibre;/* //on envoie les  
trois sms entrant puis on va voir si on bloque ou non ; si la date limitée est atteinte le client ainsi ne  
peut que recevoir= arrive à la date de blocage le client est bloqué (ne peut ni recevoir nii emmettre  
des appel ) on peux le debloqué en payant la facture ; réinitialisé l'abonnement de ce client*/
```

DESCRIPTION DES CLASSES AVEC LEURS SHEMA

Abonnement Forfaitaire :

```
//attributs
private int duréeFixe;
private int duréeFixeBloqué;
private Double SoldeFixe; //on en aura besoin au cas de renouvellement
```

//constructeur sans paramètre :

AbonForfaitaire () –AbonForfaitaire ;

//constructeur avec paramètres :

AbonForfaitaire (Date dateDebAbon, int durée, double soldeMontant) throws DateException, NegativeException-AbonLibre ; //ici soldeMontant est à > 0

Saisir() :void-AbonForfaitaire ; //on utilise ici un menu à choix (2500da, 1000da....)

Afficher() :void-AbonForfaitaire ; //

toString () : String-AbonForfaitaire ; /* « /envoi : -Type Abonnement , date fin du délais, solde restant » */

verifierSoldeRestant()(Boolean) – AbonForfaitaire /*on en aura besoin ; verifier que le solde est >=0*/

verifierAtteinteDateFinAbon()(Date)- AbonForfaitaire; /*renvoie la date après durée fixe de la (Date Debut Abonnement;*/

Facture(Client c):void- AbonForfaitaire; /* affiche la date de blocage, description du client, u type d'abonnement, des listes d'appels/sms effectué reçu, et le solde à payee*/

//ajouter appel

AjouterAppel(Client client, String s, Date D, Heure h, int durée) : Appel- AbonForfaitaire /*on envoie les trois sms entrant puis on va voir si on bloque ou non ; si la date limitée est atteinte le client ainsi ne peut que recevoir= arrive à la date de blocage le client est bloqué (ne peut ni recevoir ni emettre des appel) on peut le debloqué en payant la facture ; réinitialisé l'abonnement de ce client=si le solde est insuffisant on lance une exception solde insuffisant et on rend null*/

DESCRIPTION DES CLASSES AVEC LEURS SCHEMA

Abonnement Prépayer :

Classe CarteRecharge/*on a définit cette classe dans la classe Abonnement Prépayer*/

//Attribut :

Int Durée ; //mois

//-----setter et getter

//----constructeur sans paramètre

CarteRecharge() ;

CarteRecharge(**int** durée) ;

/*Le numéro de série s'incrémente automatiquement à la fin de sa durée elle ne peut plus être utilisé*/

Saisir() :void/* donnez la durée et validité+ le numéro de série qui doit être valide (de longueur 14)*/

DESCRIPTION DES CLASSES AVEC LEURS SCHEMA

- Autres classes supposé nécessaire :

Date/Heure/Mois/AdresseMail : On a utilisé les classes du devoir numéro 1

Adresse : On a utilisé 3 attribut ; code rue, nomrue et wilaya ;

Classes énumération :

- Site – Status - Type – Wilaya – mois

Clases Exceptions (qui hérite de Exception) :

DateException, TelephoneException, HeureException, ClientException//pour le blocage,
SoldeException//pour fin de paiement ou solde == 0, SoldeInssufisantException//pour un appel
qui ne s'effectue pas correctement => fin du solde

PRESENTATION DE L'APPLICATION

Présentation de l'application

PROGRAMME PRINCIPALE

Menu principal

1. Remplissage automatique des données
2. Gestion de l'opérateur
3. Gestion des clients (numéros)
4. Gestion des factures
5. Gestion des bonus
6. Quitter

Gestion des Clients

1. Ajouter numéro
2. Modifier numéro (modifier adresse, débloquer)
3. Supprimer numéro
4. consultation des données
- 5 Appeler
6. Recevoir Appels
7. Envoyer SMS
8. Recevoir SMS
9. Retourner au menu Principal
10. Quitter

Gestion d'opérateur téléphonique

1. Ajouter un point de vente
2. Supprimer un point de vente
3. Modifier un point de vente (changer type numéro ou adresse)
4. Changer pourcentage de couverture pour une wilaya donnée
5. Afficher les informations concernant l'opérateur
6. retour
7. Quitter

PRESENTATION DE L'APPLICATION

CAPTURE D ECRAN

```
5. Afficher les informations concernant l'opérateur
Operateur [nom=WOOREDO, ptrVente=[ [nomAgence=NomAgence1, adresse=ptrVentRue1,451,16, typ=PRINCIPALE, numeroTelephone=0244864501],
nom :nom1 | prénom : prenom1 | numéro de téléphone : 0560626365
e_mail : nom1prenom1@gmail.com | adresse : nomRue1,412,42
date de contrat :18/5/2018

-----
---Type Abonnement : Libre
---à partir du :18/5/2018
---facture à payer le : 18/6/2018
---solde cumulée: 40.0
-----

,

nom :nom2 | prénom : prenom2 | numéro de téléphone : 0560626377
e_mail : nom2prenom2@gmail.com | adresse : nomRue2,412,41
date de contrat :1/2/2018

-----
---Type Abonnement : Libre
---à partir du :18/5/2018
---facture à payer le : 18/6/2018
---solde cumulée: 104.0
-----

,


```

Figure 2 - Dans la gestion de l'opérateur

PRESENTATION DE L'APPLICATION

```
,  
  
nom :nom3 | prénom : prenom3 | numéro de téléphone : 0560626387  
e_mail : nom3prenom3@gmail.com | adresse : nomRue3,412,1  
date de contrat :20/3/2018  
  
-----  
---Type Abonnement : Libre  
---à partir du :12/5/2018  
---facture à payer le : 12/6/2018  
---solde cumulée: 168.0  
-----  
  
-----  
nom :nom4 | prénom : prenom4 | numéro de téléphone : 0560626322  
e_mail : nom4prenom4@gmail.com | adresse : nomRue4,412,16  
date de contrat :12/4/2017  
  
-----  
---Type Abonnement : Libre  
---à partir du :13/6/2018  
---facture à payer le : 13/7/2018  
---solde cumulée: 0.0  
-----  
  
-----  
], setWilaya=[], setAbon=[]
```

Figure 3 : suite - dans la gestion de l'opérateur

```
1.Ajouter un point de vente  
-----point de vente ajout:-----  
-----nom de l'agence:-----  
agence  
-----Numéro de téléphone:-----  
0555270444  
-----Adresse:-----  
  
numéro de la Wilaya:  
16  
  
code de la rue:  
2313  
  
nom de rue:  
ASZDZ  
-----type 'principale ou secondaire':-----  
PRINCIPALE  
[-----ajout avec succès-----  
[nomAgence=agence, adresse=ASZDZ,2313,15, typ=null, numeroTelephone=0555270444]
```

Figure 4 : suite - dans la gestion de l'opérateur- cas 1 ajouter point de vente

PRESENTATION DE L'APPLICATION

```
2-changer numéro:
donnez num a changer !!
0555270444

donnez nouveau numéro !!
0555271234
[
nouveau : [nomAgence=agnce1, adresse=kECHIDA,983,15, typ=null, numeroTelephone=0555271234]
```

Modifier point de vente :

Figure 5: suite - dans la gestion de l'opérateur – dans cas 3 : Modifier un point de vente(changer type numéro ou adresse)

```
Gestion d'opérateur téléphonique

1.Ajouter un point de vente
2. Supprimer un point de vente
3. Modifier un point de vente(changer type numéro ou adresse)
4. Changer pourcentage de couverture pour une wilaya donnée
5. Afficher les informations concernant l'opérateur
6. retour
7.Quitter4

4. Changer pourcentage de couverture pour une wilaya donnée
donnez numéro de wilaya et nouveau pourcentage de couverture
16
90
[adrar , pourcentage :2%
chlef , pourcentage :4%
laghouat , pourcentage :6%
oum_el_bouaghi , pourcentage :8%
batna , pourcentage :10%
béjaïa , pourcentage :12%
biskra , pourcentage :14%
béchar , pourcentage :16%
blida , pourcentage :18%
bouira , pourcentage :20%
tamanrassset , pourcentage :22%
tébessa , pourcentage :24%
tlemcen , pourcentage :26%
tiaret , pourcentage :28%
tizi_ouzu , pourcentage :30%
alger1 , pourcentage :90%
djelfa , pourcentage :34%
jijel , pourcentage :36%
sétif , pourcentage :38%
```

Figure 6: suite - dans la gestion de l'opérateur – dans cas 4 : chager le pourcentage de couverture d'une wilaya donnée

PRESENTATION DE L'APPLICATION

```
|
4-Liste des clients par wilaya(selon l'adrese du client)

-----
client par wilaya :-----:
[

  nom :nom3 | prénom : prenom3 | numéro de téléphone : 0560626387
  e_mail : nom3prenom3@gmail.com | adresse : nomRue3,412,1
  date de contrat :20/3/2018

  -----
  ---Type Abonnement : Libre
  ---à partir du :12/5/2018
  ---facture à payer le : 12/6/2018
  ---solde cumulée: 168.0
  -----

  ,

  nom :nom2 | prénom : prenom2 | numéro de téléphone : 0560626377
  e_mail : nom2prenom2@gmail.com | adresse : nomRue2,412,41
  date de contrat :1/2/2018

  -----
  ---Type Abonnement : Libre
  ---à partir du :18/5/2018
  ---facture à payer le : 18/6/2018
  ---solde cumulée: 104.0
  -----

  ,

  nom :nom2 | prénom : prenom2 | numéro de téléphone : 0560626377
  e_mail : nom2prenom2@gmail.com | adresse : nomRue2,412,41
  date de contrat :1/2/2018

  -----
  ---Type Abonnement : Libre
  ---à partir du :18/5/2018
  ---facture à payer le : 18/6/2018
  ---solde cumulée: 104.0
  -----

  ,

  nom :nom1 | prénom : prenom1 | numéro de téléphone : 0560626365
  e_mail : nom1prenom1@gmail.com | adresse : nomRue1,412,42
  date de contrat :18/5/2018

  -----
  ---Type Abonnement : Libre
  ---à partir du :18/5/2018
  ---facture à payer le : 18/6/2018
  ---solde cumulée: 40.0
  -----

  ]
```

Figure 7 suite- dans la gestion des client – dans cas 4 : consultation des données-clients par wilaya

PRESENTATION DE L'APPLICATION

```
2-liste des numéros bloqué(date de blocuage motif de blocuage)

-----
client bloqué :-----:

-----
nom :nom6 | prénom : prenom6 | numéro de téléphone : 0560626365
e_mail : nom6prenom6@gmail.com | adresse : nomRue1,412,42
date de contrat :18/5/2018

-----
---Type Abonnement : Forfaitaire
---à partir du :22/6/2018
---fin du délais : 22/7/2018
---solde restant : 0.0
-----

-----
bloqué le :22/8/2018
```

Figure 8: suite - dans la gestion des client – dans CAS 4: consultation des données- -clients bloqué

Conclusion

On a utilisé la généricité sur les objets ceci a servi à gérer les collections. Ce concept est très utile pour développer des objets pouvant travailler avec plusieurs types de données On a utilisé la généricité sur les objets ceci a servi à gérer des collections. Ce concept est très utile pour développer des objets pouvant travailler avec plusieurs types de données ; pour les tris des collections par type d'abonnement ou wilaya définition des autres attributs liste des appels, sms, client, point de vente...).

Voir aussi la réutilisabilité du polymorphisme sans aucun risque grâce à la notion d'héritage.