



ECOLE NATIONALE SUPÉRIEURE D'INFORMATIQUE ET D'ANALYSE DES
SYSTÈMES - RABAT

Rapport TP 3

Réalisé par :

Abdelwadoud TAMTAOUI
Rida TAZI

Encadré par :

Mlle. Ibtissam ABNANE

1

TP 3

1.1 Exercice 1

```
1
2 //Classe abstraite Forme
3 public abstract class Forme {
4     protected double xCentre;
5     protected double yCentre;
6
7     public double getX() {
8         return this.xCentre;
9     }
10
11     public double getY() {
12         return this.yCentre;
13     }
14
15     public void setX(double x) {
16         this.xCentre=x;
17     }
18     public void setY(double y) {
19         this.yCentre=y;
20     }
21
22     public abstract void afficher();
23
24     public double estDistantDe(Forme f) {
25         //ne doit pas etre declaree abstract car c la meme pour toute les formes
26         return Math.sqrt(Math.pow(f.xCentre-this.xCentre, 2)-Math.pow(f.yCentre-this.yCentre, 2));
27     }
28
29
30     public abstract double surface(); // abstract car chaque forme a sa propre formule
31
32     public void deplacement(int x, int y ) {
33         //pas abstract car c la meme procedure pour toutes les formes
34         this.xCentre+=x;
35         this.yCentre+=y;
36     }
37 }
38
39
40 //Rectangle
41 public class Rectangle extends Forme implements Deformable {
42
43     protected double largeur;
44     protected double hauteur;
45
46     public void afficher() {
47         System.out.println(" C'est un rectangle");
48     }
49
50     public Forme deformation(double coeffH, double coeffV) {
51         Carre carre=new Carre();
52         this.largeur*= coeffH;
53         this.hauteur*= coeffV;
54         carre.longueurCote=this.largeur;
55         return carre;
```

```

56 }
57
58 public double surface() {
59     return this.hauteur*this.largeur;
60 }
61 }
62
63
64 //Carre
65 public class Carre extends Forme implements Deformable{
66
67     protected double longueurCote;
68
69     public void afficher() {
70         System.out.println(" C'est un carre");
71     }
72
73     public Forme deformation(double coeffH, double coeffV) {
74         Rectangle rect=new Rectangle();
75         rect.hauteur=this.longueurCote*coeffV;
76         rect.largeur=this.longueurCote*coeffH;
77         return rect;
78     }
79
80     public double surface() {
81         return Math.pow(longueurCote, 2);
82     }
83 }
84
85
86 //Triangle
87 public class Triangle extends Forme{
88
89     protected double base;
90     protected double hauteur;
91
92
93     public void afficher() {
94         System.out.println(" C'est un triangle");
95     }
96
97     public double surface() {
98         return hauteur*base/2;
99     }
100 }
101
102
103 //Cercle
104 public class Cercle extends Forme{
105
106     protected double rayon;
107
108
109     public void afficher() {
110         System.out.println(" C'est un cercle");
111     }
112
113     public double surface() {
114         return Math.PI*Math.pow(rayon, 2);
115     }
116 }
117
118
119 //Interface Deformable
120 public interface Deformable {
121     //seul les rects et les carres peuvent etre deforms
122     public Forme deformation(double coeffH, double coeffV);
123 }
124 }

```

1.2 Exercice 2

```

1
2 package exo3;
3
4 public abstract class Document {

```

```

5
6  protected int ISBN ;
7  protected String titre ;
8  protected String[] tabAuteur;
9  protected String editeur;
10 protected int anEdit;
11 protected int nbrExemplaire;
12
13
14 //le numero d'enregistrement
15 private static int numEnreg=0;
16
17
18 public Document(int ISBN, String titre, String[] tabAuteur, String editeur, int anEdit, int
19     nbrExemplaire) {
20     this.ISBN=ISBN;
21     this.titre=titre;
22     this.tabAuteur=tabAuteur;
23     this.editeur=editeur;
24     this.anEdit=anEdit;
25     this.nbrExemplaire=nbrExemplaire;
26     numEnreg++;
27 }
28 //getters
29 public int getISBN() {
30     return ISBN;
31 }
32 public String getTitre() {
33     return titre;
34 }
35 public String[] getTabAuteur() {
36     return tabAuteur;
37 }
38 public String getEditeur() {
39     return editeur;
40 }
41 public int getAnEdit() {
42     return anEdit;
43 }
44 public int getNbrExemplaire() {
45     return nbrExemplaire;
46 }
47
48 //setters
49 public void setISBN(int ISBN) {
50     this.ISBN=ISBN;
51 }
52 public void setTitre(String titre) {
53     this.titre=titre;
54 }
55 public void setTabAuteur(String[] tabAuteur) {
56     this.tabAuteur=tabAuteur;
57 }
58 public void setEditeur(String editeur) {
59     this.editeur=editeur;
60 }
61 public void setAnEdit(int anEdit) {
62     this.anEdit=anEdit;
63 }
64 public void setNbrExemplaire(int nbrExemplaire) {
65     this.nbrExemplaire=nbrExemplaire;
66 }
67
68 //pour interdire le chgmt on change private par final ...
69 //on ne peut changer que le nbrExemplaire
70
71 public int getNumero() {
72     return numEnreg;
73 }
74 //on declare le num d'enreg comme final
75 //c'est unique automatiquement car ne peut pas le modifier et il s'incrémente...
76
77 public String toString () {
78     return " ISBN : " + this . ISBN +
79     " ,Titre : " + this . titre +

```

```

80     ", nombre[U+FFFD]exemplaires " + this . nbrExemplaire ;
81 }
82
83 public void IncNbreEx (int x){
84     this . nbrExemplaire += x;
85 }
86
87 public void DecNbreEx (int x){
88     this . nbrExemplaire -= x;
89 }
90
91 public String destruct() {
92     return " le document ayant les infos suivants est ''detruit'' ISBN : " + this . ISBN +
93     " ,Titre : " + this . titre +
94     ", nombre[U+FFFD]exemplaires " + this . nbrExemplaire ;
95 }
96 }
97
98
99 //Livre
100 public class Livre extends Document{
101
102     private int nbrePages ;
103     private String type ;
104     private int tome ;
105
106     public Livre (int ISBN, String titre, String[] tabAuteur, String editeur, int anEdit, int
107         nbrExemplaire,int nbrePages, String type, int tome){
108         super( ISBN, titre, tabAuteur, editeur, anEdit, nbrExemplaire);
109         this.nbrePages = 0;
110         this.type = " indef ";
111         this.tome = 0;
112     }
113
114     public String toString () {
115         return super.toString() + this .nbrePages + " ,type " + this . type +", tome " + this .
116         tome ;
117     }
118 }
119
120 //Magazine
121 public class Magazine extends Document{
122
123     private String periodicite ;
124     private int moisEdition ;
125     private int jour ;
126
127     public Magazine (int ISBN, String titre, String[] tabAuteur, String editeur, int anEdit, int
128         nbrExemplaire,String periodicite,int moisEdition, int jour){
129         super( ISBN, titre, tabAuteur, editeur, anEdit, nbrExemplaire);
130         this.periodicite = periodicite;
131         this.moisEdition = moisEdition;
132         this.jour = jour;
133     }
134
135     public String toString () {
136         return super.toString()+ " periodicite : " + this .periodicite + " ,mois[U+FFFD]edition: " +
137         this . moisEdition + " , jour : " + this . jour ;
138     }
139 }
140
141 //Dictionnaire
142 public class Dictionnaire extends Document{
143
144     private String langue ;
145     private int nbreTomes ;
146
147     public Dictionnaire(int ISBN, String titre, String[] tabAuteur, String editeur, int anEdit,
148         int nbrExemplaire,String langue, int nbreTomes){
149         super( ISBN, titre, tabAuteur, editeur, anEdit, nbrExemplaire);
150         this.langue = langue;
151         this.nbreTomes = nbreTomes;
152     }
153 }

```

```

151
152 public String toString () {
153     return super.toString() + " langue : " + this . langue +
154     " ,nombre de tomes : " + this . nbreTomes ;
155 }
156
157 }

1 package exo3;
2
3
4 //si on avait des individus qui ont pas le droit d'emprunter...
5 // on y implement cette interface
6 public interface PeutEmprunter {
7
8     public String aEmprunte();
9     public void emprunte(Document d);
10 }
11
12 //Adherent
13 public class Adherent implements PeutEmprunter{
14
15
16     private String nomCompleet_CIN ;
17     private String adrss ;
18     private Document docEmpr[] = new Document [1];
19
20     private static int numDocEmpr=0;
21
22
23     public Adherent(String nomCompleet_CIN, String adrss,Document docEmpr[]){
24         this.nomCompleet_CIN = nomCompleet_CIN;
25         this.adrss = adrss;
26         this.docEmpr=docEmpr.clone();
27     }
28
29     public String aEmprunte() {
30         String info=new String("");
31         for(Document d : this.docEmpr) {
32             info+=d.toString() + "\n";
33         }
34         return info;
35     }
36
37     public void emprunte(Document d) {
38         if (numDocEmpr<1) docEmpr[numDocEmpr]=d;
39         numDocEmpr++;
40     }
41
42 }
43
44
45 //Etudiant
46 public class Etudiant implements PeutEmprunter{
47
48
49     private String nomCompleet_CIN ;
50     private String etablissm ;
51     private Document docEmpr[] = new Document [3];
52
53     private static int numDocEmpr=0;
54
55
56     public Etudiant (String nomCompleet_CIN, String etablissm,Document docEmpr[]){
57         this.nomCompleet_CIN = nomCompleet_CIN;
58         this.etablissm = etablissm;
59         this.docEmpr=docEmpr.clone();
60     }
61
62     public String aEmprunte() {
63         String info=new String("");
64         for(Document d : this.docEmpr) {
65             info+=d.toString() + "\n";
66         }
67         return info;
68     }
69

```

```

70     public void emprunte(Document d) {
71         if (numDocEmpr<3) docEmpr[numDocEmpr]=d;
72         numDocEmpr++;
73     }
74
75 }
76
77
78 //Professeur
79 public class Professeur implements PeutEmprunter{
80
81     private String nomComplet_CIN ;
82     private String etablissm ;
83     private Document docEmpr[] = new Document[5];
84
85     private static int numDocEmpr=0;
86
87
88     public Professeur (String nomComplet_CIN, String etablissm,Document docEmpr[]){
89         this.nomComplet_CIN = nomComplet_CIN;
90         this.etablissm = etablissm;
91         this.docEmpr=docEmpr.clone();
92     }
93
94     public String aEmprunte() {
95         String info=new String("");
96         for(Document d : this.docEmpr) {
97             info+=d.toString() + "\n";
98         }
99         return info;
100     }
101
102     public void emprunte(Document d) {
103         if (numDocEmpr<5) docEmpr[numDocEmpr]=d;
104         numDocEmpr++;
105     }
106
107 }

```