

26 sep 07 22:56

Carre.java

Page 1/1

```

package formes;

import java.awt.geom.*;

public class Carre implements Forme {
    private Rectangle delegue;

    public Carre(Point2D p, double cote) {
        delegue = new Rectangle(p, cote, cote);
    }

    public double perimetre() {
        return delegue.perimetre();
    }

    public double surface() {
        return delegue.surface();
    }

    public void changerCote(double nouvelleTaille) {
        delegue.changerHauteur(nouvelleTaille);
        delegue.changerLargeur(nouvelleTaille);
    }

    public String toString() {
        return this.getClass().getName() + " de centre ("
            + delegue.centreGravite.getX() + ","
            + delegue.centreGravite.getY() + ")" + " de taille "
            + delegue.hauteur();
    }

    public Point2D centreGravite() {
        return delegue.centreGravite();
    }

    public void placer(Point2D nouveauCentreGravite) {
        delegue.placer(nouveauCentreGravite);
    }

    public void traduire(double dx, double dy) {
        delegue.traduire(dx, dy);
    }
}

```

17 sep 08 17:28

Cercle.java

Page

```

package formes;

import java.awt.geom.*;

public class Cercle extends FormeAbstraite {
    private double rayon;

    public Cercle(Point2D p, double rayon) {
        super(p);
        this.rayon = rayon;
    }

    public double perimetre() {
        return 2 * Math.PI * rayon;
    }

    public double surface() {
        return Math.PI * rayon * rayon;
    }

    public double rayon() {
        return rayon;
    }

    public void changerRayon(double nouveauRayon) {
        rayon = nouveauRayon;
    }

    public String toString() {
        return super.toString() + " et de rayon " + rayon;
    }
}

```

14 sep 07 11:32

FormeAbstraite.java

Page 1/1

```

package formes;

import java.awt.geom.*;

public abstract class FormeAbstraite implements Forme {

    protected Point2D centreGravite;

    public FormeAbstraite(Point2D p) {
        centreGravite = (Point2D) p.clone();
    }

    public Point2D centreGravite() {
        return (Point2D) centreGravite.clone();
    }

    public void placer(Point2D nouveauCentreGravite) {
        centreGravite.setLocation(nouveauCentreGravite);
    }

    public void translater(double dx, double dy) {
        centreGravite.setLocation(centreGravite.getX() + dx, centreGravite
te
        .getY()
        + dy);
    }

    public String toString() {
        return this.getClass().getName() + " de centre ("
        + centreGravite.getX() + "," + centreGravite.get
Y() + ")";
    }
}

```

jeudi 18 septembre 2008

15 sep 07 17:16

FormeColoreeDefault.java

Page

```

package formes;

import java.awt.Color;
import java.awt.geom.Point2D;

public class FormeColoreeDefault implements FormeColoree {

    private Color fond;

    private Color encre;

    private Forme delegue;

    public FormeColoreeDefault(Forme f, Color fond, Color encre) {
        this.delegue = f;
        this.fond = fond;
        this.encre = encre;
    }

    public void changerCouleurEncre(Color nouvelleCouleur) {
        encre = nouvelleCouleur;
    }

    public void changerCouleurFond(Color nouvelleCouleur) {
        fond = nouvelleCouleur;
    }

    public Color couleurEncre() {
        return encre;
    }

    public Color couleurFond() {
        return fond;
    }

    public Point2D centreGravite() {
        return delegue.centreGravite();
    }

    public double perimetre() {
        return delegue.perimetre();
    }

    public void placer(Point2D nouveauCentreGravite) {
        delegue.placer(nouveauCentreGravite);
    }

    public double surface() {
        return delegue.surface();
    }

    public void translater(double dx, double dy) {
        delegue.translater(dx, dy);
    }

    public String toString() {
        return delegue.toString() + " d'encre " + encre + " et de fond " +
;
    }
}

```

FormeAbstraite.java, FormeColoreeDefault.java

14 sep 07 9:23

FormeColoree.java

Page 1/1

```
package formes;
import java.awt.Color;
public interface FormeColoree extends Forme {
    public Color couleurFond();
    public Color couleurEncre();
    public void changerCouleurFond(Color nouvelleCouleur);
    public void changerCouleurEncre(Color nouvelleCouleur);
}
```

14 sep 07 11:32

Forme.java

Page

```
package formes;
import java.awt.geom.Point2D;
public interface Forme {
    public double perimetre();
    public double surface();
    public Point2D centreGravite();
    public void placer(Point2D nouveauCentreGravite);
    public void translater(double dx, double dy);
}
```

14 sep 07 11:34

Formes.java

Page 1/1

```
package formes;

public class Formes {
    public static void afficher(Forme f) {
        System.out.println(f);
    }
}
```

10 sep 08 18:05

Rectangle.java

Page

```
package formes;

import java.awt.geom.*;

public class Rectangle extends FormeAbstraite {
    private double hauteur;

    private double largeur;

    public Rectangle(Point2D p, double hauteur, double largeur) {
        super(p);
        this.hauteur = hauteur;
        this.largeur = largeur;
    }

    public double perimetre() {
        return 2. * (hauteur + largeur);
    }

    public double surface() {
        return hauteur * largeur;
    }

    public double hauteur() {
        return hauteur;
    }

    public double largeur() {
        return largeur;
    }

    public void changerHauteur(double nouvelleHauteur) {
        hauteur = nouvelleHauteur;
    }

    public void changerLargeur(double nouvelleLargeur) {
        largeur = nouvelleLargeur;
    }

    public String toString() {
        return super.toString() + " de hauteur " + hauteur + " et largeur "
            + largeur;
    }
}
```

```
import java.awt.Color;
import java.awt.geom.Point2D;

import formes.Carre;
import formes.Cercle;
import formes.Forme;
import formes.FormeColoreeDefaut;
import formes.Formes;
import formes.Rectangle;

public class TestFormes {
    public static void main(String args[]) {
        Rectangle r = new Rectangle(new Point2D.Double(Double.parseDouble(args[0]), Double
        .parseDouble(args[1])), Double.parseDouble(args[2]), Double
        .parseDouble(args[3]));
        Cercle c = new Cercle(new Point2D.Double(Double.parseDouble(args[4]), Double
        .parseDouble(args[5])), Double.parseDouble(args[6]));
        System.out.println(r.surface());
        System.out.println(c.perimetre());
        Carre ca = new Carre(new Point2D.Double(Double.parseDouble(args[7]), Double
        .parseDouble(args[8])), Double.parseDouble(args[9]));
        System.out.println(r.surface());
        System.out.println(c.perimetre());
        System.out.println(ca.perimetre());

        Formes.afficher(r);
        Formes.afficher(c);
        Forme cac = new FormeColoreeDefaut(ca, Color.red, Color.blue);
        Formes.afficher(cac);
    }
}
```