TD05 Correction

Matière : PROGRAMMATION OBJET	Classes : SEM21

```
package fg;

public abstract class Polygone {
  protected int nbCote;

  public Polygone(int nbCote) {
    super();
    this.nbCote = nbCote;
  }

  public void afficher() {
    System.out.println("Je suis un polygone");
    System.out.println("      Nb Cote : " + nbCote);
    System.out.println("     Surface :" + getSurface());
    System.out.println("     Perimetre :" + getPerimetre());
  }

  public abstract float getSurface();
  public abstract float getPerimetre();
}
package fg;
public class Triangle extends Polygone {
  protected float cote1;
  protected float cote2;
  protected float cote3;
  protected float base;
  protected float hauteur;

  public Triangle(float cote1, float cote2, float cote3, float base,
      float hauteur) {
    super(3); // un triangle est un polygone qui a 3 cotés
    this.cote1 = cote1;
    this.cote2 = cote2;
    this.cote3 = cote3;
    this.base = base;
    this.hauteur = hauteur;
  }
  public float getSurface() {
    return base * hauteur / 2;
  }
  public float getPerimetre() {
    return cote1 + cote2 + cote3;
  }
  public void afficher() {
    super.afficher();
    System.out.println("Je suis un triangle");
  }
}
```

```java
package fg;
public class Rectangle extends Polygone {
  protected float longueur;
  protected float largeur;

  public Rectangle(float longueur, float largeur) {
    super(4); // un rectangle est un polygone qui a 4 cotés
    this.longueur = longueur;
    this.largeur = largeur;
  }
  public float getSurface() {
    return longueur * largeur;
  }
  public float getPerimetre() {
    return 2 * (longueur + largeur);
  }
  public void agrandir(int coef) {
    longueur *= coef;
    largeur *= coef;
  }
  public void afficher() {
    super.afficher();
    System.out.println("Je suis un Ractangle");
  }
}
package fg;
public class Carre extends Rectangle {
  public Carre(float cote) {
    super(cote, cote);
    // un carre est un rectangle qui a
    // longueur = largeur
  }
  public void afficher() {
    super.afficher();
    System.out.println("Je suis un Carre");
  }
}
package fg;
public class StructurePolymorphe {
  public static final int MAX_POLYGONE = 10;
  private Polygone[] tPolygone;
  public StructurePolymorphe() {
    // réservatin des case pour le tableau
    tPolygone = new Polygone[MAX_POLYGONE];
  }
  public void init() {
    // initialisation du tableau
    tPolygone[0] = new Rectangle(3, 6);
    tPolygone[1] = new Carre(3);
    tPolygone[2] = new Triangle(3, 6, 1, 8, 9);
    tPolygone[3] = new Triangle(7, 18, 10, 3, 5);
    tPolygone[4] = new Rectangle(3, 6);
    tPolygone[5] = new Triangle(3, 6, 1, 3, 5);
    tPolygone[6] = new Rectangle(4, 6);
    tPolygone[7] = new Triangle(3, 3, 5, 3, 5);
    tPolygone[8] = new Rectangle(3, 6);
    tPolygone[9] = new Carre(6);
  }
```

```
public void afficher() {
  System.out.println("---------------");
  for (int i = 0; i < tPolygone.length; i++) {
    if (tPolygone[i] instanceof Carre)
      System.out.print("C");
    else if (tPolygone[i] instanceof Triangle)
      System.out.print("T");
    else
      System.out.print("R");
    System.out.println("(" + tPolygone[i].getSurface() + ","
        + tPolygone[i].getPerimetre() + ")");
    System.out.println("---------------");
  }
}
public void afficherStat() {
  int nbCarre = 0;
  int nbTriangle = 0;
  int nbRectangle = 0;
  for (int i = 0; i < tPolygone.length; i++)
    if (tPolygone[i] instanceof Carre)
      nbCarre++;
    else if (tPolygone[i] instanceof Triangle)
      nbTriangle++;

    else
      nbRectangle++;
  System.out.println("NB CARRE: " + nbCarre);
  System.out.println("NB TRIANGLE: " + nbTriangle);
  System.out.println("NB RECTANGLE: " + nbRectangle);
}
public float calculerSommeSurfaceRectangle() {
  float somme = 0;
  for (int i = 0; i < tPolygone.length; i++)
    if ((tPolygone[i] instanceof Rectangle)
        && !(tPolygone[i] instanceof Carre))
      somme = somme + tPolygone[i].getSurface();
  return somme;
}
public float calculerSommeSurfaceCarre() {
  float somme = 0;
  for (int i = 0; i < tPolygone.length; i++)
    if (tPolygone[i] instanceof Carre)
      somme += tPolygone[i].getSurface();
  return somme;
}
public float calculerSommeSurfaceTriangle() {
  float somme = 0;
  for (int i = 0; i < tPolygone.length; i++)
    if (tPolygone[i] instanceof Triangle)
      somme += tPolygone[i].getSurface();
  return somme;
}
public float calculerSommeSurfaceTotale() {
  float somme = 0;
  for (int i = 0; i < tPolygone.length; i++)
    somme += tPolygone[i].getSurface();
  return somme;
}
```

```java
  public void afficherSurfaceRectangleMax() {
    float max = 0;
    for (int i = 0; i < tPolygone.length; i++)
      if (tPolygone[i] instanceof Rectangle)
        max = Math.max(max, tPolygone[i].getSurface());
    if (max == 0)
      System.out.println("la structure ne contient aucun rectangle");
    else
      System.out.println("la surface maximale des rectangles: " + max);
  }
  public void afficherSurfaceTriangleMin() {
    int indiceMin = -1;
    for (int i = 0; i < tPolygone.length; i++)
      if (tPolygone[i] instanceof Triangle)
        if (indiceMin == -1)
          indiceMin = i; // c'est le premier triangle
        else
          indiceMin = (tPolygone[indiceMin].getSurface() <= tPolygone[i]
              .getSurface()) ? indiceMin : i;
    if (indiceMin == -1)
      System.out.println("la structure ne contient aucun triangle");
    else
      System.out.println("la surface minimale des triangles: "
          + tPolygone[indiceMin].getSurface());
  }
}
package fg;
public class TestStructurePolymorphe {
  public static void main(String[] args) {
    // TODO Auto-generated method stub
    StructurePolymorphe st;
    st = new StructurePolymorphe();
    st.init();
    st.afficher();
    st.afficherStat();
    System.out.println("La somme des surfaces des carres : "
        + st.calculerSommeSurfaceCarre());
    System.out.println("La somme des surfaces des rectangles : "
        + st.calculerSommeSurfaceRectangle());
    System.out.println("La somme des surfaces des triangles : "
        + st.calculerSommeSurfaceTriangle());
    System.out.println("La somme des surfaces totales : "
        + st.calculerSommeSurfaceTotale());
    st.afficherSurfaceRectangleMax();
    st.afficherSurfaceTriangleMin();
  }
}
```