

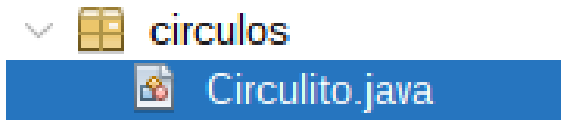
EJERCICIOS REFACTORIZACIÓN DE CÓDIGO JAVA EN NETBEANS

Ejercicio 1:

Realizar las siguientes refactorizaciones sobre el paquete **circulos** que **dispone de una clase de pruebas unitarias TestNG**.

Recuerda que después de realizar la refactorización de cada ejercicio debes comprobar con las pruebas unitarias que la clase sigue pasando las pruebas.

- Renombrar la clase **Circulo** por **Circulito**.



- Renombrar el método **ObtenerArea** por **getArea**.
- Renombrar los campos **x** e **y** por **coordenadaX** y **coordenadaY**.

```
...va User.java x Incidence.java x UserDB.java x IncidenceDB.java x Incider
Source History
1  /*
2  * To change this license header, choose License Headers in Project P
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package circulos;
7
8  /**
9   *
10  * @author Esther Ferreira
11  */
12  public class Circulito {
13
14      private int coordenadax;
15      private int coordenaday;
16      private double radio;
17
18      public Circulito() {
19      }
20
21      public Circulito(int coordenadax, int coordenaday, double radio) {
22          this.coordenadax = coordenadax;
23          this.coordenaday = coordenaday;
24          setRadio(radio);
25      }
26
27      public void setX(int coordenadax) {
28          this.coordenadax = coordenadax;
29      }
30
31      public void setY(int coordenaday) {
32          this.coordenaday = coordenaday;
33      }
34
35      public void setRadio(double radio) {
36          this.radio = radio < 0.0 ? 0.0 : radio;
37      }
38
39      public int getX() {
40          return coordenadax;
41      }
42
43      public int getY() {
44          return coordenaday;
45      }
46
47      public double getRadio() {
48          return radio;
49      }
50  }
```

```
...va User.java x Incidence.java x UserDB.java x IncidenceDB.java x IncidenceManager.java x IncidenceAdmin.java
Source History
25 }
26
27 public void setX(int cordenadax) {
28     this.cordenadax = cordenadax;
29 }
30
31 public void setY(int cordenaday) {
32     this.cordenaday = cordenaday;
33 }
34
35 public void setRadio(double radio) {
36     this.radio = radio < 0.0 ? 0.0 : radio;
37 }
38
39 public int getX() {
40     return cordenadax;
41 }
42
43 public int getY() {
44     return cordenaday;
45 }
46
47 public double getRadio() {
48     return radio;
49 }
50
51 public double getDiametro() {
52     return radio * 2;
53 }
54
55 public double getPerimetro(){
56     return Math.PI * getDiametro();
57 }
58
59 public double getArea(){
60     return Math.PI * radio * radio;
61 }
62
63 public void trasladarCentro(){
64     cordenadax += 5;
65     cordenaday += 5;
66 }
67
68 @Override
69 public String toString() {
70     return "Circulo[" + "x=" + cordenadax + ", y=" + cordenaday + ", radio=" + radio + "]\n";
71 }
72
73
```

- Introducir la constante `LIMITERADIO` de tipo `double` con el valor `0.0`

```
private double radio;
```

```
public final static int LIMITERADIO = (int) 0.0;
```

- Cambiar parámetros del método `trasladarCentro` para que tenga dos parámetros `trasladarX` y `trasladarY` de tipo `int`. Hacer los cambios necesarios para que el código del método permita engadir a la `coordenadaX` el valor de `trasladarX` y a la `coordenadaY` el valor de `trasladarY`.

```
public void trasladarCentro(){
    int trasladarx;
    int trasladary;

    cordenadax += 5;
    cordenaday += 5;
}
```

- ```

1 * @param args the command line arguments
2 */
3 public static void main(String[] args) {
4 facerOperacion();
5
6 Circulito circulo = new Circulito(cordenax: 837, cordenaday: 43, radio: 2.5);
7
8 String salida = " La coordenada X es " + circulo.getX()
9 + "\nLa coordenada Y es " + circulo.getY()
10 + "\nEl radio es " + circulo.getRadio();
11
12 circulo.trasladarCentro();
13
14 circulo.setRadio(radio: 4.25);
15 salida += "\nLa nueva ubicación y el radio del círculo son " + circulo.toString();
16
17 DecimalFormat dosDigitos = new DecimalFormat(pattern: "0.00");
18 salida += "\nEl diámetro es " + dosDigitos.format(number: circulo.getDiametro());
19 salida += "\nEl perímetro es " + dosDigitos.format(number: circulo.getPerimetro());
20 salida += "\nEl área es " + dosDigitos.format(number: circulo.getArea());
21
22 JOptionPane.showMessageDialog(parentComponent: null, message: salida);
23
24 System.exit(status: 0);
25 }
26
27 private static void facerOperacion() {
28 }

```

- Mueve el atributo `marcha`, el método `getMarcha` y el método `setMarcha` desde la superclase `Bicicleta` a la subclase `MTB` con la opción de menú *Refactorizar/Degradar*, probablemente quede algún problema que solucionar y no se haga todo automáticamente, soluciona y explica dicho problema.
- Extrae una interfaz para los métodos `getAltoAsiento` y `setAltoAsiento`, usando la opción de menú *Refactorizar/Extraer Interfaz*.
- Extrae una superclase con el campo `velocidad` y los métodos `getVelocidad`, `acelerar` y `frenar` usando la opción de menú *Refactorizar/Extraer Superclase*.

