```java
package juego;

import java.awt.Color;

public class Aguila {
    private double x;
    private double y;
    private double ancho;
    private double alto;
    private double angulo;
    private double escala;
    private Color color;
    private Image imagen;
    private boolean perdioVida;

    public Aguila(double x, double y) {
        this.x = x;
        this.y = y;
        this.ancho = 90;
        this.alto = 70;
        this.angulo = 0;
        this.escala = 0.7;
        this.color = Color.magenta;
        this.imagen = Herramientas.cargarImagen("aguila.gif");
        this.perdioVida = false;
    }// Arbol

    public boolean chocaConPiedra(Piedra[] piedra) {
        for (int i = 0; i < piedra.length; i++) {
            // Con este metodo recorremos un arreglo de piedras y verificamos si nuestro
            // objeto coliciona con dicha piedra arrojando true si se cumple y si esa es
            // nula.
            if (piedra[i] != null && piedra[i].getX() - piedra[i].getDiametro() / 2 < x + ancho / 2
                    && x - ancho / 2 < piedra[i].getX() + piedra[i].getDiametro() / 2 && piedra[i].getY() < y +
  alto / 2
                    && y - alto / 2 < piedra[i].getY()) {
                piedra[i] = null; // luego del salir del ciclo for volvemos a esa piedra nula
                return true;
            }
        }
        return false;
    }

    public static void agregaAguila(Aguila[] t, Entorno e) { // este metodo recorre un array de aguila
        Random rand = new Random(); // invocamos el metodo Random

        for (int i = 0; i < t.length; i++) {
            int distancia = rand.nextInt(e.ancho(), 5000);
            // si el aguila es nula creamos en esa posicion el objeto.
            if (t[i] == null) {
                t[i] = new Aguila(distancia, 100);
                return;
            }
        }

    }// agregarTigre

    public void descender() {
        if (x < 700 && x > 400) { // si el Eje X es menor a 700 y mayor a 400 incrementamos el eje Y
            this.y += 5;
        }
    }

    public void dibujarAguila(Entorno e) {

        e.dibujarImagen(imagen, x, y, angulo, escala); // dibujamos el aguila con la imagen y sus cordenadas
    }// dibujarTigre

    public boolean saleDePantalla() {

        if (this.x < -100) { // si el eje X es < a -100 retorna true, sino false
            return true;
        } else {
            return false;
        }
    }

```

```
 82     public void desplazar(double v) {
 83         this.x -= v; // Al eje X le restamos el valor del parametro dado
 84     }// desplazar
 85 
 86     public double getX() {
 87         return x;
 88     }
 89 
 90     public double getY() {
 91         return y;
 92     }
 93 
 94     public double getAncho() {
 95         return ancho;
 96     }
 97 
 98     public double getAlto() {
 99         return alto;
100     }
101 
102     public boolean isPerdioVida() {
103         return perdioVida;
104     }
105 
106     public void setPerdioVida(boolean perdioVida) {
107         this.perdioVida = perdioVida;
108     }
109 
110 }
```