

## Práctica 6. Generación de código

---

Una vez completadas las actividades desarrolladas en las prácticas anteriores se debiera disponer de un editor gráfico de diagramas de clases sobre el que es posible dibujar clases y establecer relaciones de herencia entre las mismas. También debiera ser posible gestionar atributos y métodos sobre las clases dibujadas.

El paso siguiente será generar el código java o esqueleto asociado a cada una de las clases generadas. El código se generará de forma **estática**, es decir, una vez dibujada la clase y especificados sus atributos y métodos se dará la posibilidad al usuario de generar el código asociado a la misma.

Véase el siguiente ejemplo,

1. Dibujamos el diagrama de clases, sirva como ejemplo el mostrado en la figura 1. En la figura pueden verse dos clases una de ellas abstracta (*Figura2D*) y una subclase de la anterior (*Circulo*). Sobre ellas hay definidas atributos y métodos. A nivel de la clase *Figura2D* tenemos dos variables privadas enteras: x e y y un método *area()* abstracto. En la clase *Circulo* tenemos un atributo *radio* y la especificación del método *área()*.
2. A partir de ese diagrama el código generado automáticamente y correspondiente a ambas clases debiera ser el siguiente:

```
public abstract class Figura2D {  
  
    // atributos  
  
    private int x;  
    private int y;  
  
    // métodos  
  
    public abstract double area();  
}  
  
public class Circulo extends Figura2D {  
  
    // atributos  
  
    private int radio;  
  
    // métodos  
  
    public double area() {  
        // sitúe aquí su código  
    }  
}
```

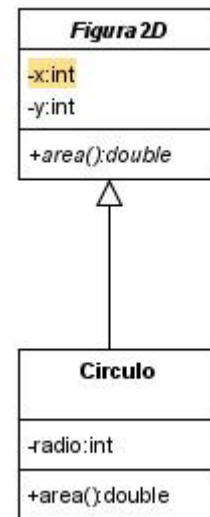


Figura 0. Diagrama de clases de ejemplo

3. Para cada clase dibujada se generará el archivo .java correspondiente con el código asociado a la misma.
4. Los archivos podrán ser compilados y ejecutados desde el propio entorno desarrollado tal y como se vio en la práctica 2. No es necesaria la utilización de *Threads* para este supuesto, aunque si aconsejable.

Una vez desarrolladas las actividades correspondientes a las prácticas 4, 5 y 6 se habrían finalizado las actividades practicas asociadas a la asignatura de Diseño y Programación Orientado a Objetos. El alumno deberá presentar una memoria que consistirá en lo siguiente:

- a) Documento javadoc impreso de los ficheros java desarrollados en las prácticas 4, 5 y 6.
- b) Documento en el que se especifiquen los patrones de diseño utilizados o identificados del API de Java. Para cada patrón de diseño debe especificarse una justificación de su uso, su utilidad y el diagrama de clases que representa el patrón con las clases desarrolladas.
- c) Un disquete o CD con el proyecto y las fuentes documentadas utilizando javadoc correspondientes a las prácticas 4, 5 y 6.

La defensa de las prácticas presentadas se realizará a través de entrevista establecida con anterioridad con el profesor responsable de tu grupo de prácticas. En caso de no estar apuntado a ningún grupo de prácticas se puede entregar la memoria a cualquiera de los profesores de la asignatura.