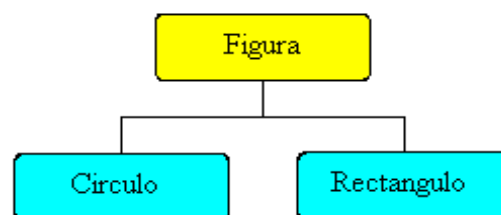


Consideremos las figuras planas cerradas como el rectángulo, y el círculo. Tales figuras comparten características comunes como es la posición de la figura, de su centro, y el área de la figura, aunque el procedimiento para calcular dicha área sea completamente distinto.

Podemos por tanto, diseñar una jerarquía de clases, tal que la clase base denominada *Figura*, tenga las características comunes y cada clase derivada las específicas. La relación jerárquica se muestra en la figura:



La clase *Figura* es la que contiene las características comunes a dichas figuras concretas por tanto, no tiene forma ni tiene área. Esto lo expresamos declarando *Figura* como una clase abstracta, declarando la función miembro *area* **abstract**.

Las clases abstractas solamente se pueden usar como clases base para otras clases. No se pueden crear objetos pertenecientes a una clase abstracta. Sin embargo, se pueden declarar variables de dichas clases.

En el juego del ajedrez podemos definir una clase base denominada *Pieza*, con las características comunes a todas las piezas, como es su posición en el tablero, y derivar de ella las características específicas de cada pieza particular.

Así pues, la clase *Pieza* será una clase abstracta con una función **abstract** denominada *mover*, y cada tipo de pieza definirá dicha función de acuerdo a las reglas de su movimiento sobre el tablero.

## La clase *Figura*

La definición de la clase abstracta *Figura*, contiene la posición *x* e *y* de la figura particular, de su centro, y la función *area*, que se va a definir en las clases derivadas para calcular el área de cada figura en particular.

```
public abstract class Figura {  
    protected int x;  
    protected int y;  
    public Figura(int x, int y) {  
        this.x=x;  
        this.y=y;  
    }  
    public abstract double area();  
}
```

## La clase *Rectangulo*

Las clases derivadas heredan los miembros dato *x* e *y* de la clase base, y definen la función *area*, declarada **abstract** en la clase base *Figura*, ya que cada figura particular tiene una fórmula distinta para calcular su área.

Por ejemplo, la clase derivada *Rectangulo*, tiene como datos, aparte de su posición (*x,y*) en el plano, sus dimensiones, es decir, su anchura *ancho* y altura *alto*.

```
class Rectangulo extends Figura{  
    protected double ancho, alto;  
  
    public Rectangulo(int x, int y, double ancho, double alto){  
        super(x,y) ;  
        this.ancho=ancho;  
        this.alto=alto;  
    }  
  
    public double area(){  
        return ancho*alto;  
    }  
}
```

La primera sentencia en el constructor de la clase derivada es una llamada al constructor de la clase base, para ello se emplea la palabra reservada **super**.

El constructor de la clase derivada llama al constructor de la clase base y le pasa las coordenadas del punto *x* e *y*. Después inicializa sus miembros *ancho* y *alto*.

En la definición de la función *area*, se calcula el área del rectángulo como producto de la anchura por la altura, y se devuelve el resultado

## La clase *Circulo*

```
class Circulo extends Figura{
    protected double radio;

    public Circulo(int x, int y, double radio){
        super(x,y) ;
        this.radio=radio;
    }

    public double area(){
        return Math.PI*radio*radio;
    }
}
```

Como vemos, la primera sentencia en el constructor de la clase derivada es una llamada al constructor de la clase base empleando la palabra reservada **super**. Posteriormente, se inicializa el miembro dato *radio*, de la clase derivada *Circulo*.

En la definición de la función *area*, se calcula el área del círculo mediante la fórmula  $\pi * r^2$ , o bien  $\pi * r * r$ .

La constante *Math.PI* es una aproximación decimal del número irracional  $\pi$ .

## Uso de la jerarquía de clases

Creamos un objeto *c* de la clase *Circulo* situado en el punto (0, 0) y de 5.5 unidades de radio. Calculamos y mostramos el valor de su área.

```
Circulo c=new Circulo(0, 0, 5.5);  
System.out.println("Area del círculo "+c.area());
```

Creamos un objeto *r* de la clase *Rectangulo* situado en el punto (0, 0) y de dimensiones 5.5 de anchura y 2 unidades de largo. Calculamos y mostramos el valor de su área.

```
Rectangulo r=new Rectangulo(0, 0, 5.5, 2.0);  
System.out.println("Area del rectángulo "+r.area());
```

Veamos ahora, una forma alternativa, guardamos el valor devuelto por **new** al crear objetos de las clases derivadas en una variable *f* del tipo *Figura* (clase base).

```
Figura f = new Circulo(0, 0, 5.5);  
System.out.println("Area del círculo "+f.area());  
f=new Rectangulo(0, 0, 5.5, 2.0);  
System.out.println("Area del rectángulo "+f.area());
```

