

Programación Orientada a Objetos

Tarea 3.2. Relaciones entre clases

José Fidel Argudo Argudo Francisco Palomo Lozano
Inmaculada Medina Bulo Gerardo Aburrizaga García
Pedro Delgado Pérez



Versión 2.0



Tarea 3.2. Cuestiones

Ejercicio 1

Se dispone de una clase base `Persona` y dos clases especializadas `Alumno` y `Profesor`. Se quiere saber qué alumnos están matriculados en qué asignaturas y qué profesores imparten qué asignaturas, y viceversa. Para ello hay dos opciones:

- Dos asociaciones bidireccionales varios a varios, una entre `Alumno` y `Asignatura`, y otra entre `Profesor` y `Asignatura`.
- Una única asociación bidireccional varios a varios entre las clases `Persona` y `Asignatura`.

¿Cuál de los dos opciones considera más conveniente?

Tarea 3.2. Cuestiones

Ejercicio 2

Supóngase que existen ya definidas dos clases `Ventana` (ventana gráfica), y `Barra` (barra de desplazamiento) y se quiere definir una nueva clase `VentanaBarra` (ventana con barra de desplazamiento). Indique si definiría la nueva clase utilizando alguna de las anteriores o como una nueva clase independiente. En caso de utilizar alguna de las ya definidas explique qué relaciones son las que se establecen entre ellas y cómo las codificaría. Razone la respuesta.

Tarea 3.2. Cuestiones

Ejercicio 3

Dadas las clases A y B, indicar qué asignaciones son correctas:

```
1  class A { /* ... */ };
2  class B: public A { /* ... */ };

4  main() {
5      A objA, *pA;
6      B objB, *pB;
7      pA = &objA;
8      pB = &objB;
9      objA = objB;
10     objA = (A)objB;
11     objB = objA;
12     objB = (B)objA;
13     pA = pB;
14     pB = pA;
15     pB = (B*)pA;
16 }
```

Tarea 3.2. Cuestiones

Ejercicio 4

Sea cierta clase base B y una derivada D. Ambas tienen definido un cierto método `f()`. Diga si el siguiente código es correcto y a qué método `f()` se llamaría.

```
1 B b, *bp;  
2 D d, *dp;  
  
4 b.f();  
  
6 bp = &d;  
7 bp->f();  
  
9 dp = &d;  
10 dp->f();
```

Tarea 3.2. Cuestiones

Ejercicio 5

Dadas las siguientes declaraciones :

```
1 struct A { int a; };  
2 struct B : public A { int b; };  
3 struct C : public A { int c; };  
4 struct D : public B, public C { int d; } v;
```

¿Cuántos miembros tiene el objeto v? ¿Cómo se accede a cada uno de ellos?