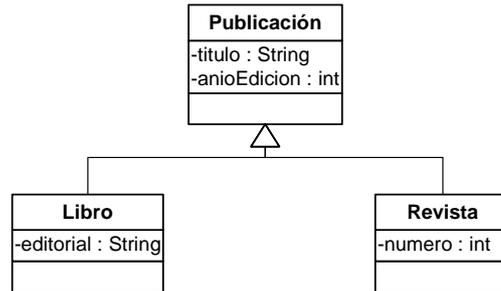


Auxiliar VI

METODOLOGÍAS DE DISEÑO Y PROGRAMACIÓN CC3002 @ 2009

Ejercicio 1

Implementar en Java el siguiente diagrama de clases de diseño:

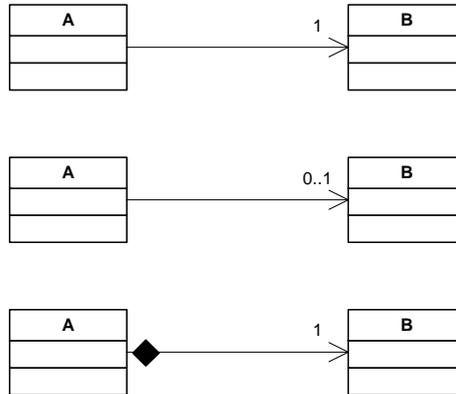


Nota:

- Implementar además las operaciones *get* y *set* para cada uno de los atributos.

Ejercicio 2

Implementar en Java los siguientes diagramas de clases:



Nota:

- Implementar operaciones que permitan establecer los links.

Ejercicio 3

A partir del diseño realizado en el ejercicio 4 del auxiliar IV (Usina y Turbinas), implementar en Java la operación `potenciaUsina()`. Utilizar un arreglo de tamaño 10 para almacenar las turbinas de una usina.

Ejercicio 4

Implementar en Java un datatype llamado `Racional` que permita definir y manipular números racionales, representados como un par de enteros (p/q). El datatype deberá permitir realizar como mínimo las siguientes operaciones sobre racionales:

```
suma      : racional × racional → racional
diferencia : racional × racional → racional
producto  : racional × racional → racional
cociente  : racional × racional → racional
equals    : racional × racional → boolean
```

También se deben implementar las operaciones habituales entre racionales y enteros. Se debe sobrecargar los operadores que considere necesario. Recordar que Java no dispone de una construcción específica para datatypes, por lo que `Racional` deberá ser implementado como una clase.

Como ejemplo, el datatype `Racional` debería soportar la siguiente aplicación:

```
public static void main(String[] args) {
    Racional r1(2,3), r2;
    Racional r3(4), r4 = r1;

    if (r1.equals(r2))
        r1 = r1.suma(r2);
    else
        r1 = r1.diferencia(r2);

    if (!r3.equals(r4))
        r3 = r3.producto(r4);
    else
        r3 = r3.cociente(r4);

    if (r1.equals(2))
        r1 = r1.suma(1);
    else
        r1 = r1.diferencia(1);
    if (!r3.equals(3))
        r3 = r3.producto(2);
    else
        r3 = r3.cociente(2);
}
```

Probar la implementación con el ejemplo.

A pesar de que la implementación en Java del datatype `Racional` sea en realidad como una clase, ¿es posible que mediante una inspección del código un lector sepa que `Racional` es conceptualmente un datatype? En caso afirmativo, ¿cómo es posible saberlo?

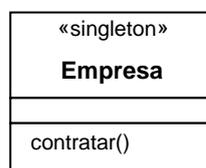
Ejercicio 5

Considere el problema de una empresa con diferentes tipos de empleados presentado en las transparencias del curso correspondientes al tema “03 - Conceptos Básicos”.

- a) Implemente en Java la jerarquía de empleados presentada en la transparencia número 55.
- b) Implemente en Java la clase `Empresa` presentada en la transparencia número 70 teniendo en cuenta lo siguiente:
 - i. Para implementar la asociación desde `Empresa` a `Empleado`, deberá utilizar las colecciones genéricas discutidas en el curso (para elegir el tipo de colección a utilizar, tenga en cuenta que el nombre de un empleado lo identifica).
 - ii. La clase `Empresa` deberá proveer operaciones para contratar a empleados fijos y empleados jornaleros.
 - iii. La clase `Empresa` deberá proveer la operación `getLiquidaciónTotal()`, la cual deberá exhibir un comportamiento análogo a la implementación dada en la transparencia número 70 (pero teniendo en cuenta su implementación puede verse afectada por la decisión tomada en i.)
 - iv. La clase `Empresa` deberá proveer una operación `getLiquidacion(String)`, que reciba el nombre de un empleado (independientemente de su tipo) y devolver la liquidación del mismo.

Ejercicio 6

Escriba el código completo en Java para la clase `Empresa` mostrada en la figura, y realice un `main()` en el que se invoque a la operación `contratar()`.



Ejercicio 7

- a) Implementar en Java la realidad descrita en el ejercicio 7 del auxiliar V (reloj digital).
- b) Considere ahora el siguiente requisito adicional:

“El reloj digital debe generar un evento indicando la hora y los minutos actuales, cada vez que éstos cambian su valor. Se necesita que este evento pueda ser recibido por cualquier cantidad de objetos. Dichos objetos deberán registrarse previamente como interesados ante el reloj.”

Implemente además un módulo de prueba.

Ejercicio 8

Considere el mecanismo de fábricas presentado en la transparencia número 12 correspondiente al tema “14 - Diseño - Patrones”. Implemente en Java dicho mecanismo, teniendo en cuenta que la clase `Formulario` deberá proveer una operación `eventoUsuario()` la cual invoque a las operaciones del sistema en el orden en que aparecen declaradas en la interfaz `ISistema`.

Ejercicio 9

Implemente en Java el diseño completo resultante del ejercicio 2 del auxiliar V.

Ejercicio 10

La ANFP (Asociación Nacional de Fútbol Profesional) desea desarrollar un sistema que automatice la actividad contractual en el fútbol profesional chileno. El equipo de desarrollo decidió implementar un pequeño prototipo que brinde las funcionalidades básicas del sistema. La primera tarea realizada fue obtener una descripción de la realidad sobre la cual trabajar:

“La ANFP cuenta con jugadores afiliados de los cuales se tiene conocimiento de su nombre y edad. Al igual que los jugadores, existen clubes afiliados, de los cuales se conoce el nombre. Un jugador puede haber jugado en el mismo club durante varios períodos.

Los contratos firmados por un jugador son de carácter anual por un monto en moneda nacional. Como estrategia de control se decidió fijar el período de contratos al comienzo del año y no permitir firmar contratos por más de un año.

Los contratos firmados entre un jugador y un club determinado se archivan en un mismo expediente el cual es identificado por un número de expediente.

Los contratos se firman por medio de un contratista. Cada contratista puede participar en varios contratos durante el mismo año.

Cada entidad afiliada a la ANFP (jugadores, clubes y contratistas) poseen número (que los identifica entre sus pares) y año de afiliación. No interesa conocer datos de estos antes de su afiliación.”

Para la realización del prototipo, el equipo de desarrollo determinó seis etapas a seguir.

Se pide:

Lleve a cabo cada una de las etapas para culminar con el desarrollo del prototipo.

- 1) Realizar el Modelo de Dominio que represente la realidad planteada.
Escriba en lenguaje natural las restricciones del modelo. No incluya las restricciones que indiquen que un atributo deba ser positivo ni de unicidad de identificadores.
- 2) Realizar el Diagrama de Secuencia del Sistema para el siguiente caso de uso:
“Dar los números de expediente de los contratos realizados por un contratista determinado, con un club en particular, a partir del año indicado”
Los datos necesarios (contratista, club, y año) serán indicados por el usuario.
- 3) Realizar los contratos de las operaciones del proceso anterior.
Las pre- y poscondiciones deben ser especificadas en lenguaje natural.
- 4) Realizar los diagramas de colaboración de las operaciones involucradas en el caso de uso anterior.
- 5) Realizar el diagrama de clases de diseño de esta realidad, incluyendo solamente las clases que participan en el diseño.
- 6) Implemente completamente su solución junto con un caso de prueba.
Para implementar las asociaciones utilice las colecciones genéricas vistas en el curso, eligiendo apropiadamente, para cada utilización, el tipo de cada una.