

PRÁCTICO 2: MODELOS ESTÁTICOS – INGENIERÍA DIRECTA

DIAGRAMAS DE CLASES Y DIAGRAMAS DE OBJETOS

PARTE A

1.- Se desea realizar un subsistema de proyectos universitarios, que tenga en cuenta las siguientes características:

- Un proyecto tiene un director, quien deber ser docente de la Universidad. La información relevante para el director es el número y el tiempo total (en meses) de proyectos que ha dirigido.
- Cada uno de los integrantes de un proyecto es docente de la Universidad.
- Cada docente pertenece al menos a un departamento, donde la información que nos interesa de éste último es el nombre, su director quien a su vez es un docente y los consejeros docentes departamentales.
- Un departamento pertenece a una facultad, la que tiene un decano quién es también un docente.
- Un proyecto pertenece a una facultad, tiene un período de desarrollo, un objetivo, un plan y la dedicación (horas por semana) de cada uno de los integrantes del proyecto.

Se quiere responder las siguientes preguntas: en qué fecha comenzó un proyecto dado, quién es su director, quién es el director de departamento al cual pertenece el director de un proyecto y quién es el decano de la facultad a la cual pertenece un proyecto.

Se deben tener en cuenta también las siguientes restricciones:

- Un director puede dirigir hasta dos proyectos.
- Un integrante puede pertenecer a más de un proyecto, siempre que la cantidad de horas de dedicación no supere las 40 semanales entre todos los proyectos.

Modele el aspecto estático del sistema a través de un **modelo del dominio** en UML. Construya además un **diagrama de objetos** para el caso de una facultad con dos departamentos y tres proyectos cuyos directores trabajan distribuidos en los dos departamentos. Suponga además que existe al menos un integrante de proyecto que trabaja en dos departamentos.

2. Construya un **modelo del dominio** en UML para modelar la siguiente realidad:

Una compañía de tarjetas de crédito abre una cuenta a nombre de un individuo o de una empresa, cuyos empleados pueden usar las tarjetas de dicha cuenta. La compañía de tarjetas de crédito brinda tres tipos de tarjetas diferentes, con límites distintos, términos de pagos diferentes y otras condiciones.

Bajo una cuenta (empresarial o personal) puede haber una o varias tarjetas autorizadas. Es importante conocer quién tiene cada una. Físicamente se logra escribiendo en relieve el nombre del poseedor de la tarjeta sobre la tarjeta, con el número de cuenta y la fecha de vencimiento. Uno puede tener una cuenta con una tarjeta para sí mismo, y otra para su cónyuge. Nuestra empresa puede tener una cuenta con una tarjeta para nosotros.

La compañía de tarjetas de crédito necesita conocer quién es el propietario de una cuenta, quiénes tiene tarjetas y cuántas tarjetas de distinto tipo son mantenidas en una cuenta personal o de una empresa dada.

a) Considere que límites distintos para cada tipo de tarjeta vienen fijados inicialmente por las constantes LT1, LT2 y LT3.

b) ¿Qué pasaría si los límites de las tarjetas no fueran establecidos inicialmente?

3. Sistema de una compañía aérea:

Un ticket cubre un viaje entre dos ciudades. Este tiene cupones que corresponden a las distintas combinaciones de vuelo que se deben realizar para cubrir el viaje entre las dos ciudades. Cada cupón cubre un vuelo entre dos aeropuertos, y posee información sobre qué compañía realiza el vuelo, a qué número de vuelo corresponde, la clase, la fecha y la hora de partida del vuelo. Los cupones se encuentran ordenados.

Un avión tiene varias salidas cada día, identificado por el día y la hora de partida, número de vuelo y el aeropuerto de salida.

Un ticket, además, tiene la siguiente información: a qué pasajero le corresponde dicho ticket, la fecha de emisión y el precio total del pasaje.

Un avión, que posee un nombre y tipo, tiene una serie de asientos que se identifican por un número. Cada asiento tiene identificado una ubicación (ventanilla, pasillo, ...) y está dentro del sector de fumadores o de no fumadores.

Se quiere dar respuesta a las siguientes operaciones de consulta:

¿Qué avión ha sido asignado a qué vuelo?

¿Cuántos tickets han sido vendidos entre un período dado?

¿Qué asientos ya han sido asignados en un vuelo dado?

Realizar el **modelo del dominio** del sistema y un **diagrama de objetos** para una configuración posible del sistema.

4.- La siguiente es la extracción del dominio de la Dirección de Patrimonio (DP) de una Universidad Nacional.

Una orden de compra es el documento de compra de bienes que refleja las compras por licitación. Existen dos tipos de órdenes, las de capital para comprar bienes capitales; y las de consumo, usadas para comprar bienes de consumo. La información que ambas contienen es similar, excepto donde se indica.

Las órdenes tienen dos partes principales: un encabezado y un cuerpo formado por *designaciones o renglones*. El encabezado está compuesto por la siguiente información:

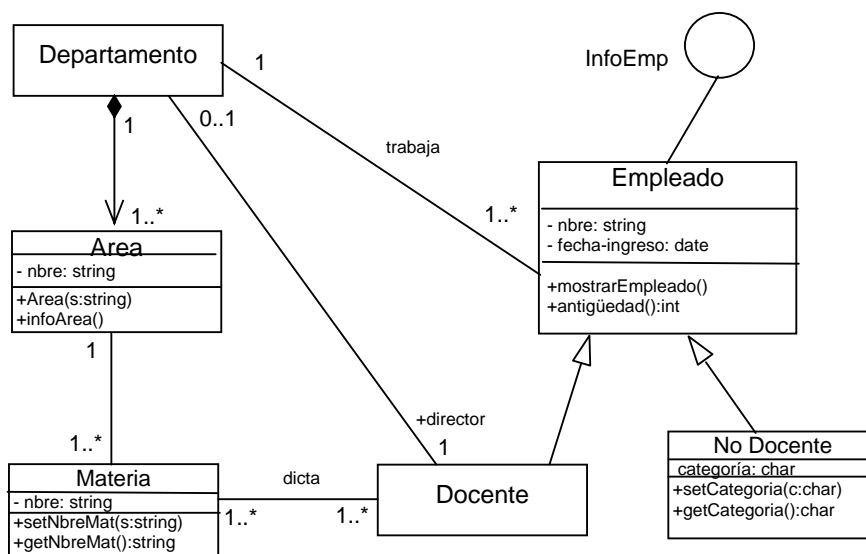
- Número de orden: compuesto por un número creciente y el año de la orden de compra.
- Fecha de elaboración de la orden.
- Número de partida o cuenta.
- Proveedor: nombre, domicilio, localidad, teléfono y número de CUIT.
- Número de expediente de la compra (establecido por Mesa de Entradas)
- Dependencia destino de la compra (mostrando la ubicación de la misma).
- Dependencia de entrega de la compra (usualmente el Dpto. de Almacenes, también deber mostrarse la ubicación de la misma)
- Plazo de la entrega.
- Forma de Pago.
- Monto total de la compra.
- Agente responsable de la compra (sólo las órdenes de capital). Identificar Número de Agente, nombre, apellido, CUIL, domicilio.
- Estado de la orden (Parcial/Total) dependiendo de si los bienes han sido entregados en forma total o parcial.
- Fecha de informe a la DP.

El cuerpo de las órdenes está compuesto por una o más designaciones identificadas con un número de orden (nro. de renglón para las designaciones de las órdenes de consumo) que la identifica dentro de la orden de compra. La información contenida en cada designación es la siguiente:

- Nro. de orden (o de renglón).
- Descripción detallada del bien a comprar.
- Marca y modelo (sólo las órdenes de capital).
- Unidad de medida del bien ('C/U', 'Kg.', etc.)
- Cantidad de bienes.
- Precio unitario.
- Precio total de la designación.

Realizar el **Modelo de Dominio** de este problema considerando todos los elementos indicados en la narrativa, mostrando las distintas, según correspondan, clases, relaciones, roles, multiplicidades, atributos y tipos. No considere especificar los métodos de las clases.

5.- Dado el siguiente diagrama de clases, genere el código Java correspondiente.



Considere, para este ejemplo, que en Java los arreglos se declaran de la siguiente manera:

tipo-base-del arreglo nombre[] o bien *tipo-base-del arreglo[] nombre*

Luego de declararlo, es necesario crearlo en memoria por medio de una sentencia "new", de la siguiente manera:

nombre = new tipo-base-del arreglo nombre[tamaño] .

Ejemplo:

```
float numeros[ ]; // declara el arreglo, es equivalente a float[ ] numeros;
numeros = new float[30]; // crea arreglo de 30 posiciones de números de punto flotante.
```

6.- Represente el modelo del dominio a través de un diagrama de clases de UML.

Una inmobiliaria maneja información acerca de inmuebles que se alquilan (casas, departamentos, locales) y clientes que pueden ser propietarios de inmuebles o inquilinos.

Una misma persona puede ser propietaria de más de un inmueble en alquiler, pero también puede alquilar más de un inmueble.

De un inmueble nos interesa: los metros cuadrados, dirección, servicios; de un departamento: costo de las expensas, cantidad de pisos del edificio; de un local: si tiene baño, cocina y teléfono.

De los clientes necesitamos: apellido y nombre, DNI, dirección; de los propietarios: teléfono de contacto, al igual que los inquilinos.

Para el alquiler de inmueble siempre se exige una propiedad de garantía, de la cual registramos el número catastral.

7.- Explicar una clase parametrizada. Dar un ejemplo donde se muestre claramente su utilidad.

8.- Dado el siguiente diagrama de clases, genere el código Java correspondiente. Considere distintas estructuras de datos para su implementación.

