

## Orientación a Objetos 1 - 2018 - Primera fecha

### Facturación de energía

En una ciudad nos piden desarrollar un sistema de registro de mediciones y facturación de energía. Hay dos tipos de clientes: clientes residenciales e industrias. El sistema debe ofrecer la siguiente funcionalidad:

**Dar de alta una persona como nuevo cliente residencial:** Se indica el nombre de la persona, su apellido y el domicilio donde se provee la energía. El sistema registra el nuevo cliente y lo retorna.

**Dar de alta una industria como nuevo cliente:** Se indica el nombre de la empresa y el domicilio donde se provee la energía. El sistema registra el nuevo cliente y lo retorna.

**Registrar una medición de consumo de un cliente:** Se indica el cliente y el consumo de energía (un número de unidades de energía). Se indica el mes (del 1 al 12) y el año al que corresponde la medición. El sistema registra la medición.

**Actualizar el tarifario.** Se indica el precio de la unidad de energía para clientes y el precio de la unidad de energía para industrias. El sistema actualiza los precios en el tarifario de la empresa.

**Calcular el monto a facturar a un cliente.** Se indica un cliente, un mes y un año. El sistema obtiene el consumo para ese mes y año (asuma que existe), calcula el monto a facturar a ese cliente por ese mes de ese año, y lo devuelve. El calculo se efectúa de la siguiente manera:

Si el cliente es residencial, se multiplica el consumo de ese mes y año, por el precio de la unidad de energía para clientes residenciales. Además, si el consumo de ese mes y año es menor al consumo como del mismo mes del año anterior, se hace un descuento del 10%. Si no hay registro para el mismo mes del año anterior, también se hace el descuento del 10%.

Si el cliente es una industria, se multiplica el consumo de ese mes y año por el precio de la unidad de energía para industrias. Además, si el consumo fue inferior al del mes inmediatamente anterior, se hace un descuento del 2%. Si el sistema no dispone de medición para el mes inmediato anterior, no hace el descuento de 2%.

## Empresa de mantenimiento

Nos contratan para desarrollar una plataforma para una empresa de mantenimiento de viviendas. En particular nos enfocaremos en las contrataciones de los servicios ofrecidos y los montos a pagar por los mismos. El sistema debe permitir la siguiente funcionalidad:

**Registrar un cliente:** se provee un nombre y una dirección. El sistema registra el cliente y lo retorna.

**Registrar un servicio de limpieza:** se provee un precio por hora, una cantidad de horas y una tarifa mínima. El sistema registra el servicio y lo retorna.

**Registrar un servicio de parqueización:** se provee un precio por hora, una cantidad de horas, cantidad de máquinas a utilizar y un costo de mantenimiento de las mismas. El sistema registra el servicio y lo retorna.

Además, se debe dar la posibilidad de contratar estos servicios:

**Contratar por única vez:** se provee un cliente, la fecha y un servicio. El servicio quedará contratado para dicho cliente en la fecha especificada.

**Contratar de forma prolongada:** se provee un cliente, un servicio, una fecha inicial y una cantidad de días (asumir que esta cantidad es siempre mayor a uno). El servicio quedará contratado por dicho cliente la cantidad de días especificada.

**Obtener el monto a pagar por un cliente:** dado un cliente, se retorna el monto a abonar por todos los servicios contratados, según se describe a continuación:

- El monto a abonar por las contrataciones por única vez es el monto a abonar del servicio. En caso de ser contratado para un fin de semana se cobra un recargo del 15%.
- El monto a abonar por las contrataciones prolongadas es el monto a abonar del servicio multiplicado por la cantidad de días que fue contratado. En caso de haber sido contratado por más de 5 días se le realiza un descuento del 10%.

En cuanto a los servicios, el monto a abonar se calcula de la siguiente manera:

- El monto a abonar de los servicios de limpieza se calcula como el precio por hora multiplicado por la cantidad de horas. Este valor no debe ser menor a la tarifa mínima y en caso de serlo, se abonará dicha tarifa.
- El monto a abonar de los servicios de parqueización se calcula como el precio por hora multiplicado por la cantidad de horas y a este monto se le adiciona el costo de mantenimiento de las máquinas multiplicado por la cantidad de máquinas.

**Obtener la cantidad de servicios con monto mayor a un valor:** dado un valor, se retorna la cantidad de servicios que brinda la plataforma cuyos montos a abonar lo superan.

### Tareas:

1. Diseño de su solución en un diagrama de clases UML.
2. Implementación en Java de la funcionalidad requerida.
3. Implemente los tests necesarios para la funcionalidad de obtener el monto a pagar por un cliente considerando sólo contrataciones de única vez para servicio de limpieza.

### Notas:

- Implemente **todos** los constructores que considere necesarios.
- Puede implementar un getter y un setter que necesite, y asumir la existencia del resto.
- Para verificar si una fecha (instancia de LocalDate) cae en fin de semana, puede utilizar:  
`DayOfWeek dia = DayOfWeek.from(fecha);`  
`dia.equals(DayOfWeek.SATURDAY) || dia.equals(DayOfWeek.SUNDAY);`



# Orientación a Objetos I - 5/11/2022

Nos contratan de la agencia de recaudación de la provincia de Buenos Aires para hacer un sistema para el cálculo del impuesto que deben pagar los contribuyentes.

El sistema ofrece la siguiente funcionalidad:

**Dar de alta un contribuyente:** Se provee nombre, dni, email y localidad. El sistema da de alta al contribuyente y lo retorna. El contribuyente no tiene ningún bien a su nombre.

**Dar de alta un Inmueble:** se provee el número de partida, el valor del lote, el valor de la edificación, y el contribuyente (propietario). El sistema da de alta el inmueble y lo retorna.

**Dar de alta un automotor:** se provee patente, marca, modelo, fecha de fabricación, valor y el contribuyente (propietario). El sistema da de alta al automotor y lo retorna.

**Dar de alta una embarcación:** se provee patente, nombre, fecha de fabricación, valor y el contribuyente (propietario). El sistema da de alta la embarcación y la retorna.

**Calcular el impuesto que debe pagar un contribuyente:** dado un contribuyente, se debe calcular cuánto debe pagar de impuestos, según la siguiente especificación:

- Por cada inmueble que posea, el contribuyente debe pagar un 1% del valor del mismo, que se calcula sumando el valor del lote y el valor de la edificación.
- Por cada uno de los otros bienes, se debe pagar un porcentaje del valor de los mismos, en función de su fecha de fabricación.
  - En caso de **superar los 10 años**, no deben pagar nada.
  - En caso contrario, el porcentaje para un automotor es el **5%** mientras que para una embarcación ese porcentaje varía según el valor de la misma. Si éste es menor a 1 millón, es el **10%**, caso contrario, el **15%**.

**Contribuyentes que más pagan de una localidad:** dada una localidad y un número N, se debe retornar los N contribuyentes de la localidad recibida que más deben pagar por sus bienes.

Su tarea es diseñar y programar en Java lo que sea necesario para ofrecer la funcionalidad antes descrita. Se espera que entregue lo siguiente:

1. Diseño de su solución en un diagrama de clases UML.
2. Implementación en Java de la funcionalidad requerida.
3. Implemente los tests necesarios para la funcionalidad de calcular el impuesto, justificando su elección en base a valores de borde y particiones equivalentes. Considere contribuyentes que solamente pueden tener embarcaciones.

## Notas:

- Para calcular los años entre dos fechas puede utilizar la siguiente expresión  
`ChronoUnit.YEARS.between(fecha1, fecha2);`

Donde fecha1 es anterior a fecha2. La expresión retorna la cantidad de años entre ambas fechas

- Implemente todos los constructores que considere necesarios.
- Puede implementar un getter y un setter, y asumir la existencia del resto.

S, constructores, No getters y setters (UML)



# Parcial - Orientación a Objetos I - 26/11/2022

## - Primer recuperatorio -

**MyCarBoxes** es una empresa que vende repuestos y ofrece servicios de reparación a vehículos. El sistema registra órdenes de compra de repuestos y órdenes de reparación. Las reparaciones son llevadas a cabo por personal de MyCarBoxes, quienes cobran por hora sus áreas de reparación. Cada empleado puede cobrar un valor por hora distinto.

Implemente la siguiente funcionalidad:

**Dar de alta un repuesto.** Se provee el nombre, fecha de fabricación y costo. El sistema registra el repuesto y lo retorna.

**Dar de alta un empleado:** Se provee el nombre completo y el valor por hora. El sistema registra el empleado y lo retorna.

**Registrar una orden de compra de repuestos:** Se provee la patente del vehículo y la lista de repuestos. El sistema registra la orden con la fecha actual y retorna la orden creada.

**Registrar una orden de reparación:** Se provee la patente del vehículo, la descripción de la reparación, la lista de repuestos para llevarla a cabo (en caso de ser necesarios), la lista de empleados de MyCarBoxes que intervienen en la reparación (siempre interviene al menos un empleado) y la cantidad de horas necesarias para completarla. El sistema registra la orden con la fecha actual y la retorna.

**Facturar un conjunto de órdenes:** dada una lista de órdenes, se retornan las facturas correspondientes a cada una de ellas. En cada factura debe quedar registrada la fecha, la patente del vehículo, el monto final y el descuento. El monto final se calcula en base a un costo y un incremento de ganancia según el caso:

- El costo de la orden de compra de repuestos se calcula sumando los costos de cada uno de los repuestos involucrados. Sobre este costo se aplica un incremento de ganancia del 15%. En caso de que alguno de los repuestos tenga más de 5 años desde su fabricación, este incremento de ganancia será sólo del 8%.
- El costo de la orden de reparación se calcula sumando:
  - lo que cobra cada uno de los empleados intervinientes y la cantidad de horas de la reparación (para simplificar todos los empleados intervinientes cobrarán esa cantidad de horas).
  - el costo de los repuestos involucrados.Sobre el costo de la orden se aplica un incremento de ganancia del 10%.
- Para ambos casos y luego de aplicar la ganancia, si MyCarBoxes tuviera registrada alguna otra orden para la misma patente en los últimos 12 meses, le aplica un descuento del 5% sobre el total.

### Tareas:

1. Diseño de su solución en un diagrama de clases UML.
2. Implementación en Java de la funcionalidad requerida.
3. Implemente los tests necesarios para la funcionalidad de facturar una orden de compra de repuestos, justificando su elección en base a valores de borde y particiones equivalentes.

### Notas:

- Para calcular los años entre dos fechas puede utilizar la siguiente expresión `ChronoUnit.YEARS.between(fecha1, fecha2);`  
Donde `fecha1` es anterior a `fecha2`. La expresión retorna la cantidad de años entre ambas fechas
- Implemente todos los constructores que considere necesarios.
- Puede implementar un getter y un setter, y asumir la existencia del resto.