

# PRÁCTICA INCREMENTAL 1

## Objetivos:

- Diseño y la manipulación de objetos
- Diseño y manipulación de matrices de objetos
- Detección y modelado de relaciones entre clases
- Aplicación de todos los conocimientos sobre programación aprendidos hasta el momento.
- Uso de modificadores de visibilidad

## Enunciado

Desarrolle un programa que simule parte del funcionamiento de un sistema de gestión de compra y venta de productos de una tienda. El funcionamiento normal de este tipo de sistemas se detalla a continuación. Supongamos que llega un empleado delante del ordenador y lo primero que hace es autenticarse (login y password) en el sistema. Una vez autenticado, el empleado puede realizar las siguientes tareas: modificación de los productos en stock, la realización de pedidos según las peticiones de un cliente o la modificación de su contraseña.

Así pues, tenemos un sistema principal que es el encargado de interactuar con el empleado a través de ciertos menús y comunicarle con los subsistemas correspondientes dependiendo de la operación que desee realizar. El sistema principal consta de dos subsistemas: el sistema de gestión de empleados y el sistema de gestión de pedidos.

El sistema de gestión de empleados se encargará de cargar desde un archivo de texto plano todos los empleados de la empresa, cuando comience a funcionar el programa. La función principal de dicho subsistema es la comprobación de que el login y password de cada empleado que entra en el sistema son correctos.

Cada empleado se describe con tres características: código de acceso (valor entero único), nombre del usuario (cadena de texto) y password (cadena de texto).

Mientras tanto, el subsistema de ventas se encarga de todas las acciones propias de la venta de productos. Este subsistema es el encargado de la gestión de todos los productos disponibles en la tienda, los cuales se leen desde un archivo de texto plano que se carga cuando comienza a funcionar el programa. Cada producto se caracteriza por un código (valor entero), un nombre (cadena de texto) y un precio (valor real). Se supone que siempre hay de todos los productos en el stock.

El subsistema de ventas se encarga de todas las tareas relativas a los productos. Entre esas tareas está listar todos los productos que hay disponibles en el sistema, actualizar los datos de los productos comprobando que no se repitan datos y consultar los productos existentes.

El funcionamiento detallado será el siguiente. En primer lugar, cuando el sistema arranque solicitará al empleado su código de acceso y su password, si ambos son correctos entonces el proceso continuará, en caso contrario, volverá a solicitar los datos hasta que estos sean correctos:

Introduzca código de empleado

1234

Introduzca password de empleado

pozno

Una vez autenticado el empleado se mostrarán las acciones principales del sistema general:

1. Hacer pedido
2. Modificar producto
3. Cambiar contraseña empleado
4. Log out

Si el usuario selecciona la opción 4 (log out) el sistema desconectará la sesión del empleado y volverá a la primera acción del sistema, volver a solicitar el código y password de un nuevo empleado. Obviamente esta acción simula la marcha de empleado y la espera de que otro nuevo llegue para registrarse en el sistema.

Si el usuario elige la opción 3 (cambiar contraseña) el sistema solicitará al usuario una nueva contraseña y modificará la ya existente a través del subsistema de gestión de empleados.

Si el usuario selecciona la opción 2 (modificar un producto) primero se listarán todos los productos registrados en el sistema y se solicitará al empleado que indique cuál desea modificar. Una vez hecho esto, aparecerá el siguiente menú:

1. Modificar nombre
2. Modificar precio
3. Modificar código

El usuario seleccionará qué atributo desea modificar y el sistema le solicitará el nuevo dato y lo actualizará. En el caso del nombre y el código del producto es necesario comprobar que el nuevo nombre o código no coincidan con el nombre o código de otros productos ya existentes, dado que estos identificadores son únicos. En caso de error se mostrará un mensaje al empleado y volverá a solicitarse un valor distinto hasta que el usuario escriba un dato que no esté repetido.

Por último, si el empleado decide hacer un pedido (opción 1) lo primero que se le solicitará será indicar el número de productos que desea comprar. Una vez leído este dato aparecerá el siguiente menú:

- 1.1 Añadir producto
- 1.2 Visualizar precio total
- 1.3 Imprimir factura
- 1.4 Terminar pedido

Para añadir un nuevo producto (opción 1.1) se listarán todos los productos disponibles en el sistema y se le solicitará al empleado que indique qué producto desea añadir al pedido. En caso de que ese producto haya sido añadido anteriormente se mostrará un mensaje de error y se volverá otra vez al mismo menú, en caso contrario se añadirá el producto. Si el usuario ya

añadió todos los productos que indicó en la primera acción antes de este menú entonces el sistema le mostrará un mensaje de error advirtiéndole de que no puedo añadir más productos.

La opción 1.2 muestra por pantalla el precio del pedido en ese momento.

La opción 1.3 visualiza la factura total, indicando los datos de cada producto adquirido, el precio final de la factura y la persona que le atendió. Un sistema real debería imprimir la factura mediante una impresora y guardarla en algún archivo o base de datos, pero en este caso se limitará a mostrarla por pantalla para simplificar el problema.

La opción 1.4 terminará este menú aunque no se haya terminado de realizar el pedido y se volverá otra vez el menú principal.

**Nota 1:** El sistema no deja nunca de funcionar, es decir, siempre realiza alguna acción, no hay ninguna acción de finalizar el sistema

**Nota 2:** El formato del archivo de empleados o productos tiene el siguiente formato. Primero aparecerá el número de empleados o productos disponibles que tiene la empresa y a continuación el aparecerá en cada línea el nombre de una propiedad que define a un empleado o producto y en la línea siguiente el valor que le corresponde a esa propiedad.

Así una empresa con 2 productos (1524, champu, 12.3€) y (1523, gel, 12.8€) se simularía con un archivo con el siguiente contenido:

```
productos:
2
codigo:
1524
nombre:
champu
precio:
12.3
codigo:
1523
nombre:
gel
precio:
12.8
```

**Nota 3:** Es necesario recordar que para leer cadenas a través de un objeto de la clase Scanner mediante el método `nextLine()` es necesario colocar antes otra llamada `nextLine()`.

### Condiciones de la entrega:

La fecha de entrega de la práctica será **el lunes 29 de marzo de 2016 antes de las 10:00**. Dicha semana, cada grupo defenderá su práctica a su profesor en su horario de prácticas habitual. Se recuerda que es necesario la asistencia de ambos miembros de la pareja de prácticas. En caso de que uno de ellos No se presente, ese alumno no será evaluado y por tanto aparecerá como suspenso en la convocatoria ordinaria. Igualmente

es necesario recordar que es necesario que la documentación entregada contenga análisis, diseño, manual de usuario y código fuente. En la fase de diseño deberá entregarse tanto el diagrama de clases global como el diagrama UML de cada clase.

Toda esta información se subirá a través de Moodle en un espacio habilitado para tal efecto, antes de la hora de la defensa para que quede constancia de la entrega. Posteriormente a esta fecha no se permitirá subir ningún otro documento. Toda la documentación deberá ir comprimida en un único archivo con extensión .rar o .zip. Preferiblemente la documentación debería ir en formato PDF y el código fuente en su correspondiente archivo .java. Los archivos binarios .class no son necesarios ya que deberán generarse de forma fácil siguiendo las instrucciones redactadas en el manual de usuario.

El funcionamiento de la práctica no garantiza el aprobado de la misma, se valorarán otros aspectos como la robustez y eficiencia de la misma, el correcto diseño y perfecta utilización de las técnicas y metodologías comentadas en clase de teoría, la documentación entregada y el cumplimiento de los requisitos definidos en el enunciado.