

Práctica Obligatoria - 3ª Evaluación

Práctica conjunta: Entornos, Base de datos y Programación, Formación y orientación laboral y Sistemas informáticos.

El alumno sólo deberá realizar las partes del enunciado de las que esté matriculado. En esta ~~segunda~~ **tercera** práctica ~~no~~ **sí hay unión entre la base de datos y la capa de programación**, únicamente a nivel de diseño.

Nota: Recuerda agregar en el README.md TODOS los detalles implementados para que sean evaluados.

Enunciado:

MercaDAW es una empresa española de distribución con sede en el municipio de Parla perteneciente a la provincia de Madrid.

El establecimiento constan de una superficie de 130 m², que responden al modelo denominado comercio urbano de proximidad, y mantienen un surtido en alimentación, droguería, perfumería y complementos donde incluyen sus propias marcas blancas, junto con otras marcas comerciales. ~~; pero solo ponen a la venta 1 producto.~~

MercaDAW es un supermercado muy exclusivo ~~y es que solo vende un producto, y mientras no se ordene lo contrario, no se trabaja con ningún otro producto. El supermercado es único, es decir, no tiene más sedes.~~ *que vende productos de primera calidad y que se está expandiendo por Europa: Parla, Madrid, París, Londres, Roma y Berlín.*

Todos productos en MercaDAW están distribuidos por Categorías, que son las siguientes:

- Alimentación: Se les aplica un 4% de iva
- Droguería: Se les aplica un 10% de iva
- Cosmética: Se les aplica un 21% de iva

MercaDAW dispone de una plantilla de 432 empleados con un sueldo estándar de 1000. Dentro del conjunto de trabajadores existen unos puestos particulares:

- Reponedores: Tienen un sueldo de 1200 €
- Cajeros: Tienen un sueldo de 1300 €
- Encargados: Tienen un sueldo de 1500 €

MercaDAW, para enviar publicidad y hacer estadísticas, solicita a sus clientes la siguiente información:

- Código Postal
- Correo electrónico (no obligatorio)
- Nombre (no obligatorio)

Interesa guardar la fecha en la que el cliente realiza la COMPRA, que puede constar ~~de un solo producto~~ de varios productos.

MercaDAW ha contratado a tu empresa para empezar a automatizar tareas que antes hacían en papel, para ello te solicitan una aplicación que permita a los trabajadores (usuarios de la aplicación):

1. Carga de datos del sistema (desde fichero, vía java, a la base de datos). Cargar los datos iniciales de la base de datos (usando sistemas, java y bases de datos): Productos e información de los trabajadores
2. Dar de alta un producto (Desde Java a Base de datos)
3. Alta de un empleado (Desde Java a Base de datos)
4. Visualizar los datos del producto (Desde Base de datos a Java)
5. Visualizar listado de empleados (Desde Base de datos a Java)
6. *Alta compra (procedimiento PL/SQL). Actualiza el stock de cada producto. Identifica cada compra con un número que se obtendrá con la función PL/SQL, OBTENER_IDENTIFICADOR que retorna el número identificador (será el consecutivo al de la última compra realizada) (Solo Base de datos)*
7. *Listado de compras por código postal (vista SQL) (Solo Base de datos)*
8. *Listado de stock de todos los productos (vista SQL) (Solo Base de datos)*
9. Obtener el precio de venta del producto en Euros y en Dólares (Solo java)
10. Imprimir una etiqueta para clasificar el producto y dar información adicional a los consumidores del supermercado sobre el producto. (Desde Base de Datos a Java)
11. Cálculo de nóminas de los trabajadores (Base de datos, Java y FOL). Se generará un fichero con las nóminas de los trabajadores.
12. Creación de copias de seguridad exportando los datos de la base de datos a un fichero y moviendo el fichero (con la fecha y hora de la generación) a una carpeta de backups. Se creará una aplicación en java (que genere el fichero) y un script para su organización, y copiado.
13. Exportación de datos a excel, pdf u otro formato.

Al iniciar la aplicación se abrirá la interfaz de usuario con las opciones.

Carga de datos

Dado un/os fichero/s, se deberán cargar los datos iniciales en la base de datos. Los formatos utilizados serán: CVS, JSON, XML (sólo se pide uno de ellos).

Dar de alta un producto

El sistema preguntará al empleado y guardará la siguiente información del producto:

- Nombre
- Marca
- *Categoría*
- Precio en Euros
- IVA
- Altura en metros
- Anchura en metros

- Peso en kg
- Número de elementos que lo forman (por ejemplo una lata es 1 elemento pero un ordenador pueden ser 4 (cpu, monitor, teclado y ratón)).
- Descripción

Dar de alta un empleado

El sistema pedirá nombre, apellidos y tipo del trabajador. En la creación de un empleado se generará automáticamente un id que deberá ser auto incremental.

Visualizar los datos del producto

Se mostrará la información del producto en venta.

```
* Nombre: Lata de tomate
* Marca: Tiernoplus
* Categoría: Alimentación
* Precio en Euros: 1€
* IVA: 4%
* Altura en metros: 0,20
* Anchura en metros: 0,20
* Peso en kg: 0,4
* Número de elementos: 1
* Descripción: Es una lata de tomate de primera calidad de nuestra marca blanca TiernoPlus
```

Visualizar los empleados

Se mostrará una tabla con los datos de todos los trabajadores con la siguiente información:

- Identificador
- Nombre
- Apellidos
- Categoría

Obtener el precio de venta del producto

El precio de venta se calcula incrementado un 25% el precio original, que es la comisión que se lleva el supermercado. Por ejemplo, si un producto cuesta 1€, su precio de venta es 1.25€

Para cada elemento del producto se cobrará un suplemento sobre el precio de venta (en el ejemplo anterior 1,25€):

- Si pesa más de 1 kg se cobrará un 10%, si pesa más de 5 kg se cobrará un 15%, en caso contrario solo 5%
- Si la altura es mayor de 0,5 metros se cobrará 10% por elemento, en caso contrario solo 5% por elemento
- Si la anchura es mayor de 0,5 metros se cobrará 10% por elemento, en caso contrario solo 5% por elemento
- Si una pieza tiene mas de 2 elementos, se cobrará un gasto adicional de 10 céntimos por elemento.

Se mostrará el precio en Dólares (ratio de conversión 1 dólar es 0,90€)

En el precio total está incorporado el IVA relativo al producto (no hay que calcularlo), pero en el cálculo hay que mostrar que parte de ese precio total es por el iva. En el ejemplo: El precio total

es 1,44, y 0,06€ corresponden al iva (4%).

* Nombre: Lata de tomate
* Marca: Tiernoplus
* Precio en Euros: 1€
* Altura en metros: 0,20
* Anchura en metros: 0,20
* Peso en kg: 0,4
* Número de elementos: 1

* Precio de venta: 1,25
* Recargo por peso (5%): 0,06
* Recargo por altura (5%): 0,06
* Recargo por anchura (5%): 0,06
* Recargo por número de piezas (1 pieza): 0,00 €

* PRECIO TOTAL: 1,44 € / 1,60
* IVA (4%): 0,06 €

Imprimir una etiqueta

Se imprimirá ~~(por pantalla)~~ la siguiente etiqueta:

* Nombre: Lata de tomate
* Marca: Tiernoplus
* Categoría: Alimentación
* Descripción: Es una lata de tomate de primera calidad de nuestra marca blanca TiernoPlus

Alta de compra

Llamada al procedimiento ALTA_COMPRA (cliente, producto). Este procedimiento:

- Actualiza el stock del producto
- Llama a la función SQL OBTENER_IDENTIFICADOR_COMPRA de la nueva compra
- Introduce la fecha del sistema como fecha de compra

Listado de compras por código postal

Muestra la VISTA COMPRAS_POR_CÓDIGO_POSTAL (listado del número de compras realizadas para cada código postal almacenado)

Listado del stock de todos los productos

Muestra la VISTA STOCK_PRODUCTOS (listado con el nombre y el stock de cada producto)

Nota: Recuerda agregar en el README.md TODOS los detalles implementados para que sean evaluados.

Se pide

Nota: Recuerda agregar en el README.md TODOS los detalles implementados para que

sean evaluados.

Usando como base la solución del trimestre anterior, se pide:

Entornos de desarrollo (Grupos de 4 personas):

- Análisis del problema
 - Diagrama de casos de uso y requisitos
- Diseño de la solución
 - Descripción de la arquitectura
 - Patrones de diseño utilizado
 - Diagramas de UML vistos en clase
- Utilización de maven para la creación del proyecto
- Realización de casos de prueba utilizando JUnit y generación del informe de pruebas. Si el alumno lo ve conveniente, puede utilizar la caja blanca o caja negra para determinar los casos de prueba.
- Utilización de git y gitflow para el desarrollo del proyecto.
- Documentación del código: javadoc o Doxygen

Elementos adicionales para subir nota (no obligatorios):

- Prototipo de la aplicación utilizando Pencil o alguna alternativa
- Diseño adicional utilizando C4

Programación (Grupos de 4 personas ~~resolución en parejas o individual~~):

- Diseño de la solución
- Desarrollo de la solución:
 - Código de la solución
 - Documento Markdown con información de interés sobre el desarrollo: problemas, soluciones ingeniosas, elementos incorporados no vistos en clase...

Es imprescindible: Conexión con la base de datos para guardar y obtener la información, interfaz gráfica de acceso a todas las opciones, carga y manipulación de ficheros.

Es imprescindible tener un diseño arquitectónico justificado, establecer claramente las partes del desarrollo y los componentes que las componen.

Elementos adicionales para subir nota (no obligatorios):

- Soportar varios formatos para la carga de datos
- Utilizar librerías ETL para la carga de datos
- Establecer la configuración utilizando ficheros de properties
- Usar una base de datos distinta a SQLite
- Utilizar Docker sí es obligatorio a nivel básico.
- Agregar capacidades Web
- Exportar la información a formatos distintos de texto plano
- Exportar la información utilizando herramientas de reporting
- Exportar la información utilizando un servidor de reporting.

Base de datos (resolución en parejas o individual):

- *Esquema conceptual Entidad/Relación*
- *Esquema lógico relacional*
- *Esquema físico SQL con restricciones*
- *Introducción de datos*
- *Vistas COMPRAS_POR_CÓDIGO_POSTAL y STOCK_PRODUCTOS*
- *Procedimiento ALTA_COMPRA y función OBTENER_IDENTIFICADOR_COMPRA*

Se entregará en un fichero ZIP:

- Markdown, plantuml... (y pdf/imágenes). Se entregará toda la documentación generada y se especificará en que formato se envía.
 - Para Markdown es obligatorio, también, adjuntar también el HTML o PDF
 - Para Mermaid es obligatorio, también, adjuntar la imagen de la solución
 - Para Plantuml es obligatorio, también, adjuntar la imagen de la solución
- Código fuente implementado.

Para los diagramas se recomienda:

- **PlantUML y Mermaid**
- Software día
- Se puede usar cualquier otro en línea o instalable.
- Hacer en papel y escanearlos o hacer una foto
- ERDPLUS para esquemas de la base de datos
- MySQL como SGBD

Sistemas Informáticos:

- Entrega del script .sh que realiza la organización de las copias de seguridad
- Archivo .pdf explicando de forma detallada el funcionamiento del mismo

Formación y Orientación Laboral:

Nota: Recuerda agregar en el README.md TODOS los detalles implementados para que sean evaluados.