

Programación Orientada a Objetos

Proyecto Primer Parcial

I PAO 2025

Sistema de Venta y Distribución de Productos

Objetivo:

Desarrollar una aplicación de consola en Java que simule el funcionamiento básico de un sistema de venta de productos en línea, en el que intervienen vendedores, clientes y repartidores. El sistema debe aplicar los principios fundamentales de la programación orientada a objetos: abstracción, encapsulamiento, herencia, polimorfismo, sobrecarga y sobrescritura.

Requerimientos y Funcionalidades:

- Los usuarios del sistema deben acceder con sus credenciales de usuario y contraseña. Todos los usuarios están previamente registrados en un archivo llamado **usuarios.txt** con el siguiente formato.
De los usuarios se conoce entonces su código único, cédula, nombres, apellidos, usuario, contraseña, correo, y rol (C, R).

| CódigoÚnico | Cédula | Nombres | Apellidos | Usuario | Contraseña | Correo | Rol |
|-------------|------------|---------|-----------|---------|------------|-----------------------|-----|
| C001 | 0923456789 | Ana | Torres | atorres | cliente123 | ana.torres@email.com | C |
| R002 | 0967788990 | Luis | Medina | lmedina | entrega654 | luis.medina@email.com | R |
| ... | | | | | | | |

- De los productos que se venden en la plataforma se conoce: su categoría (ROPA, TECNOLOGÍA, DEPORTE u HOGAR), código, nombre, precio y stock. Todos los productos estarán almacenados en el archivo **productos.txt** con el siguiente formato:

| Categoría | Código | Nombre | Precio | Stock |
|------------|--------|---------------------|--------|-------|
| Tecnología | P001 | Audífonos Bluetooth | 35.99 | 20 |
| Deportes | P006 | Balón de Fútbol | 25.00 | 10 |
| ... | | | | |

- Al inicio del programa usted debe cargar una lista polimórfica de usuarios que va a contener clientes, vendedores y repartidores, **nunca va a instanciar objetos usuarios como tal**. Además, una lista de pedidos realizados que constan en el archivo y la lista de productos.
- El sistema genera notificaciones que se elaboran de acuerdo al usuario a quien va dirigida la notificación.

ROLES Y TAREAS

Los usuarios pueden ser clientes, vendedores y repartidores.

Los usuarios pueden acceder al sistema ingresando su usuario y contraseña que deben ser verificados para que se muestre el menú correcto de opciones de acuerdo a su perfil.

1. CLIENTE

De los usuarios clientes, además se conoce su número de celular y dirección. Sus datos están previamente registrados en el archivo Clientes.txt con el siguiente formato:

Clientes.txt

| CódigoÚnico | Cédula | Nombres | Apellidos | Celular | Dirección |
|-------------|--------|---------|-----------|---------|-----------|
|-------------|--------|---------|-----------|---------|-----------|

Los clientes tienen el menú de opciones:

1. Comprar
2. Gestionar pedido

1.1 COMPRAR

Al elegir la opción Comprar, el cliente debe seguir los siguientes pasos:

1. Se le muestran las categorías disponibles.
2. El cliente ingresa la opción de la categoría que quiere consultar.
3. Se le muestran los productos disponibles de la categoría elegida.
4. El cliente entonces elegirá la opción del producto que desea comprar.
5. Ingresar la cantidad que desea comprar (validar que exista stock).
6. Se mostrará el total a pagar.
7. El cliente ingresa su número de tarjeta.
8. Se simula que se realizó el pago (solo mostrando un mensaje).
9. Se asigna un repartidor, de la lista de repartidores, de forma aleatoria.
10. Se crea el pedido de compra con el estado EN PREPARACIÓN, y se registran los datos en el archivo **Pedidos.txt**.

De un pedido se conoce también la fecha del pedido, código de producto comprado, la cantidad comprada, el valor pagado, y un código del pedido generado automáticamente, y el código del repartidor.

| CodigoPedido | Fecha | CodigoProducto | Cantidad | ValorPagado | Estado | CodigoRepartidor |
|--------------|------------|----------------|----------|-------------|----------------|------------------|
| PED1001 | 2025-09-10 | P003 | 2 | 45.98 | EN PREPARACIÓN | R001 |
| PED1002 | 2025-09-11 | P001 | 1 | 35.99 | EN PREPARACIÓN | R001 |
| PED1003 | 2025-09-11 | P006 | 3 | 75.00 | EN PREPARACIÓN | R002 |
| ... | | | | | | |

Se muestra mensaje de éxito.

11. El sistema genera una notificación, por correo electrónico, con los datos del pedido.
12. Se modifica el stock del producto comprando para restar la cantidad comprada (solo en el sistema no en el archivo).

NOTA: En la clase Pedido, definirá una variable estática que irá incrementando cada vez que se cree un Pedido, y se usará para crear el id de cada pedido.

1.2 GESTIONAR PEDIDO

En esta opción el cliente puede consultar el estado de su pedido. Solo debe ingresar el código del pedido y se mostrarán los detalles como:

===== CONSULTA DE ESTADO DE PEDIDO =====

Ingrese el código del pedido: PED1003

Fecha del pedido: 2025-09-11

Producto comprado: Balón de Fútbol (Código: P006)

Cantidad: 3

Valor pagado: \$75.00

Estado actual: EN PREPARACIÓN

Repartidor: Luis Gómez

Su pedido está siendo preparado para su envío.

2. REPARTIDOR

De los usuarios repartidores, además se conoce el nombre de la empresa a la que pertenece. Sus datos están previamente registrados en el archivo **Repartidores.txt** con los siguientes formatos:

Repartidores.txt

| | | | | | |
|-------------|--------|---------|-----------|---------|-------|
| CódigoÚnico | Cédula | Nombres | Apellidos | Empresa | Email |
|-------------|--------|---------|-----------|---------|-------|

2.1 GESTIONAR PEDIDO

Al gestionar un pedido, el repartidor va a cambiar el estado de un pedido. Cuando se crea un pedido, automáticamente tendrá el estado EN PREPARACIÓN, por lo tanto, el repartidor puede cambiar el estado del pedido a EN RUTA o a estado ENTREGADO. Cada vez que el repartidor cambia el estado del producto se notifica por correo al cliente.

Debe validarse que solo pueda cambiar al estado correcto siguiendo el siguiente orden:

EN PREPARACIÓN -> EN RUTA -> ENTREGADO

===== GESTIONAR ESTADO DE PEDIDO =====

Ingrese el código del pedido que desea gestionar: PED1003

Pedido encontrado:

Fecha del pedido: 2025-09-11

Código del producto: P006

Estado actual: EN PREPARACIÓN

Seleccione el nuevo estado:

1. EN RUTA

2. ENTREGADO

Opción: 2

Error: No puede cambiar directamente de EN PREPARACIÓN a ENTREGADO.
Debe cambiar primero a EN RUTA.

Seleccione el nuevo estado:

1. EN RUTA

2. ENTREGADO

Opción: 1

Estado actualizado correctamente a EN RUTA.

Notificación enviada al cliente: El pedido PED1003 ha sido despachado y está en camino.

2.2 CONSULTAR PEDIDOS ASIGNADOS

En esta opción, al repartidor se le muestran los pedidos que le han sido asignados. Solo se mostrarán los pedidos que no han sido entregados. Por cada pedido se mostrará, su código, fecha de pedido y estado.

===== PEDIDOS ASIGNADOS =====

Buscando pedidos asignados no entregados...

Pedidos encontrados:

1. Código: PED1003

Fecha del pedido: 2025-09-11

Estado actual: EN RUTA

2. Código: PED1006
Fecha del pedido: 2025-09-13
Estado actual: EN PREPARACIÓN

Total de pedidos pendientes: 2

Recuerde que solo puede gestionar los pedidos que se encuentren EN PREPARACIÓN o EN RUTA.

3. TAREAS DEL SISTEMA: NOTIFICAR

El sistema genera notificaciones por correo electrónico en algunos casos.

1. Al realizar un pedido.
2. Al asignar un pedido.
3. Cuando un pedido cambia de estado.

En este caso se deben crear diversas versiones del método notificar.

3.1 Al realizar un pedido

Cuando se crea un pedido, se debe notificar por email al cliente. Por ejemplo:

De: correoSistema
Para: correoCliente
Asunto: Pedido realizado

El cliente NOMBRES APELLIDOS ha realizado un pedido con código PEDIDO_CODIGO el día FECHA_PEDIDO.

Producto: NOMBRE_PRODUCTO
Cantidad: CANTIDAD
Valor pagado: \$VALOR
Estado inicial: EN PREPARACIÓN

Gracias por su compra. Recibirá actualizaciones del estado de su pedido por este medio.

Por lo tanto, la versión del método notificar aquí podría ser: notificar(Cliente cliente, Pedido pedidoRealizado).

3.2 Al asignar un pedido

Cuando se asigna un pedido a un repartidor, debe notificársele por email.

De: correoSistema

Para: correoRepartidor
Asunto: Nuevo pedido asignado

Estimado/a NOMBRES APELLIDOS,

Se le ha asignado un nuevo pedido con los siguientes detalles:

Código del pedido: PEDIDO_CODIGO
Fecha del pedido: FECHA_PEDIDO
Cliente: NOMBRE_CLIENTE
Estado actual: EN PREPARACIÓN

Por favor, prepare la logística necesaria para la entrega.

Gracias por su trabajo.

Defina usted la versión del método notificar que necesita aquí.

3.3 Cuando un pedido cambia de estado.

Cuando el repartidor cambia el estado de un pedido debe notificársele al cliente. Por ejemplo:

De: correoSistema
Para: correoCliente
Asunto: Actualización del estado de su pedido

Estimado/a NOMBRES APELLIDOS,

Le informamos que el estado de su pedido con código PEDIDO_CODIGO ha cambiado a: NUEVO_ESTADO.

Fecha del pedido: FECHA_PEDIDO
Producto: NOMBRE_PRODUCTO
Repartidor asignado: CODIGO_REPARTIDOR

Gracias por confiar en nosotros.

Defina usted la versión del método notificar que necesita aquí.

NOTA: Realice sobrecarga del método de forma correcta en la clase Sistema.

NOTA: Las fechas deben manejarse con tipo de dato Date.

NOTA: Investigue como enviar correos desde la aplicación java.

NOTA: Considerar que si los métodos se comportan diferente en cada clase, debe implementar SOBRESCRITURA.

4. TAREAS DEL SISTEMA: INICIAR SESIÓN

El Sistema debe permitir que el usuario ingrese sus credenciales de autenticación y validar que sean correctas. Luego debe, verificar si la instancia es un cliente o un repartidor y mostrar el menú. Si es un Cliente se mostrará su nombre y apellido y su celular, pedir que verifique si es correcto, si el usuario indica que no es correcto, se terminará la ejecución por problemas de seguridad. En cambio, si el usuario es un Repartidor, se mostrará el nombre de la empresa a la que pertenece, pedir que verifique si es correcto, sino se terminará la ejecución por problemas de seguridad.

| | |
|---|--|
| ===== INICIO DE SESIÓN ===== Usuario: atorres Contraseña: ***** Usuario autenticado correctamente. Rol detectado: CLIENTE Bienvenida, Ana Torres Celular registrado: 0991234567 ¿Este número de celular es correcto? (S/N): S Identidad confirmada. Menú de Cliente: 1. Comprar producto 2. Gestionar pedido 3. Salir Seleccione una opción: | ===== INICIO DE SESIÓN ===== Usuario: lmedina Contraseña: ***** Usuario autenticado correctamente. Rol detectado: REPARTIDOR Bienvenido, Luis Medina Empresa asignada: Envíos Express ¿Esta empresa es correcta? (S/N): N Verificación fallida. Por motivos de seguridad se cerrará la sesión. Saliendo del sistema... |
|---|--|

5. TAREAS DEL SISTEMA: MOSTRAR MENÚS

El Sistema debe mostrar el menú adecuado de acuerdo a su tipo. Si es Cliente o Repartidor.

Para implementar esta funcionalidad debe buscar en la lista de usuarios y verificar si la instancia es un Cliente o es un Repartidor, hacer la conversión al tipo de usuario específico, y de acuerdo a ello se mostrarán las opciones específicas.

Requerimientos de desarrollo de la aplicación

Encapsulamiento:

Utilizar getters y setters para el acceso a los atributos privados de las clases. Implementar diferentes tipos de restricción con modificadores de acceso que permitan encapsular los datos de acuerdo a su naturaleza.

Abstracción:

Definir la clase Usuario como abstracta con un método abstracto para gestionarPedido que funciona diferente dependiendo del tipo de usuario que la implemente.

Herencia:

La clase Usuario será la clase padre de la cual heredarán Cliente y Repartidor.

Asegurarse de que las subclases implementen y/o sobrescriban métodos necesarios.

Polimorfismo:

Utilizar polimorfismo para manejar la lista de Usuarios que puedan contener tanto clientes como repartidores.

Cuando se inicie sesión debe verificar si el tipo de usuario que usando **instance of** y hacer la conversión, para mostrar las opciones de tarea adecuada.

Sobrescritura y Sobrecarga:

Sobrescribir métodos como toString para personalizar la representación de los objetos.

Incluya sobrecarga en el método para enviar correo electrónico y la sobrescritura en el método para gestionar pedidos.

Persistencia de Datos:

Almacenar y recuperar datos desde archivos de texto (usuarios.txt, clientes.txt, repartidores.txt, productos.txt, pedidos.txt). En este primer parcial no se enseña manejo de archivos. Sin embargo, los estudiantes tienen conocimientos adquiridos sobre la persistencia de datos en archivos. Por lo tanto, se les proporcionará los métodos necesarios para leer y escribir en archivos. Los estudiantes solo deben entender cómo usar estos métodos (abstracción).

Correo Electrónico:

El ayudante de la materia hará ejemplos en sus clases sobre implementar esta funcionalidad. Debe asistir a ayudantías o investigar sobre cómo hacerlo.

Consideraciones:

Cuando vaya a mostrar la ejecución de la aplicación debe tener datos suficientes en sus archivos, para mostrar todas las funcionalidades y limitaciones que se mencionan en este documento (por ejemplo productos con mucho stock y poco stock para demostrar validaciones).

Criterios de evaluación

Funcionalidad (40 %)

- Se valorará la apariencia del proyecto, sin que esto implique que el estudiante deba utilizar recursos adicionales a los aprendidos en clase. (El proyecto es en consola)
- Se verificará el cumplimiento de toda la funcionalidad requerida, así como las correctas validaciones de los datos.
- El proyecto debe ser debidamente probado antes de presentarlo al profesor. Por lo tanto se espera que el programa no se caiga al ejecutarlo, ni tenga un comportamiento no esperado (Si se cae en la presentación, son 5 puntos menos por cada caída)

Documentación (10 %)

- El programa fuente deberá presentar un código documentado internamente (un correcto uso de comentarios).
- Diagrama de Clases (relaciones, herencias, no multiplicidad, modificadores de acceso y estático).
- Se valorará las facilidades para mantenimiento del programa (comentario de encabezado, propiedades y métodos con nombre correcto, métodos con comentarios, uso de sangrías, **Generación de Javadoc**).
- Reporte del proyecto. Se entregará un formato de reporte que los estudiantes deben completar.

Abstracción y Uso de Objetos (15 %)

- Creación de Clases pertinentes con sus propiedades y métodos.
- Correcta interacción de objetos. Que los objetos se comuniquen entre si y no solo sean llamados todos en el main.
- Usar recursos de herencia y sobrescritura para la solución.

Modularidad y Encapsulamiento (15 %)

- Dividir el problema usando los métodos correspondientes. No escribir bloques inmensos de código.
- Encapsular correctamente el proyecto. Crear paquetes donde se agrupen clases que se relacionen.
- No olvidar que para acceder a las propiedades de las clases se debe proveer los métodos get y set.

Herencia y polimorfismo (20%)

- Identificar y emplear cuando sea necesario los principios de herencia y polimorfismo.
Revisar los puntos obligatorios que van relacionados con esto.

Penalidades

Se disminuirá un 10% de la nota en caso de no seguir buenas prácticas de programación (convención para nombre de clases, nombre de atributos y métodos, uso de constantes cuando es necesario, o algún fallo grave que se detecte en la forma de programar).

Modo de Entrega:

- Subir un zip de su proyecto al link de la tarea en el aula virtual con el siguiente formato: **POO_P#_PY1P_Apellido1_Apellido2_Apellido3**
- Solo un estudiante debe entregar el proyecto. Aquel de los tres, cuyo apellido se registre primero en la lista de asistencia.