



Tenemos un fichero csv para guardar diferentes facturas emitidas por una empresa a sus clientes. Para cada factura se tiene la siguiente información:

- **Número:** Identificador único de la factura.
- **Fecha:** Fecha de emisión de la factura.
- **Cliente:** Nombre del cliente que ha emitido la factura.
- **País del Cliente:** País de residencia del cliente.
- **Impuestos:** Porcentaje de impuestos (en tanto por cien) aplicado a la factura.
- **Tipo de Cliente:** Tipo del cliente que puede ser REGULAR, PREMIUM o VIP.
- **Productos:** Lista de productos vendidos, cada uno con su nombre, precio unitario y categoría.

Una línea de este fichero contiene los siguientes campos:

```
INV-2024-001,2024-05-18,John Doe,Estados Unidos,8.0,PREMIUM, '[Laptop - 999.99 - Electrónica]; [Mouse - 19.99 - Accesorios]; [Teclado Mecánico - 129.99 - Accesorios]; [Auriculares Bluetooth - 59.99 - Electrónica]'
```

Esta fila indica que la factura con número INV-2024-001 fue emitida el 18 de mayo de 2024 para el cliente John Doe, residente en Estados Unidos, con un impuesto del 8%, de tipo PREMIUM, y contiene una lista de productos con sus respectivos precios unitarios y categorías.

Ejercicio 1: Tipo Producto (0,5 pts)

Implemente en el paquete `fp.facturas` el tipo `Producto` mediante un *record*, de acuerdo con la siguiente información:

Propiedades:

- **Nombre**, de tipo `String`, consultable.
- **Precio**, de tipo `Double`, consultable.
- **Categoría**, de tipo `String`, consultable.

Constructores:

- **C1:** recibe un parámetro por cada propiedad básica del tipo, en el mismo orden en el que están definidas.

Representación como cadena: una cadena con todas las propiedades básicas del tipo.

Restricciones:

- **R1:** El precio no puede ser negativo.
- **R2:** El nombre no puede superar los 100 caracteres.

Criterio de igualdad: dos productos son iguales si tienen el mismo nombre, precio y categoría.



Ejercicio 2: tipo Factura (1,5 pts)

Implemente en el paquete `fp.facturas` el tipo `Factura` utilizando una clase, de acuerdo con la siguiente información:

Propiedades:

- **Número**, de tipo `String`, consultable
- **Fecha**, de tipo `LocalDate`, consultable.
- **Cliente**, de tipo `String`, consultable y modificable.
- **País**, de tipo `String`, consultable y modificable.
- **Impuestos**, de tipo `Double`, consultable y modificable.
- **Tipo de cliente**, de tipo `TipoCliente`, consultable y modificable. Puede tomar los valores `REGULAR`, `PREMIUM`, `VIP`.
- **Productos**, de tipo `List<Producto>`, consultable.
- **Precio total**, de tipo `Double`, consultable. Calculado como la suma del precio unitario por la cantidad de cada producto más los impuestos.
- **Número de productos**, de tipo `Integer`, consultable. Calculado como la cantidad total de productos en la factura.
- **Número de categorías distintas**, de tipo `Integer`, consultable. Calculado como el número de categorías únicas en la lista de productos de la factura.

Constructores:

- C1: recibe un parámetro por cada propiedad básica del tipo.

Restricciones:

- R1: El porcentaje de impuestos no puede ser negativo ni superior a 100.
- R2: La lista de productos no puede estar vacía.

Representación como cadena: una cadena con todas las propiedades básicas del tipo.

Criterio de igualdad: dos facturas son iguales si tienen el mismo número y fecha.

Criterio de ordenación: dos facturas se ordenan por fecha y, en caso de igualdad, por número.

Ejercicio 3: Factoría (1 pto)

En la clase `FactoriaFacturas`, que se le da parcialmente implementada, implemente el método:

- *Factura parsearFactura(String lineaCSV):* crea un objeto de tipo `Factura` a partir de una cadena de caracteres. La cadena de caracteres debe tener el mismo formato que las líneas del fichero CSV.



Ejercicio 4: Tratamientos secuenciales (7 ptos)

El tipo Facturas, que se proporciona implementado, tiene la siguiente descripción:

Propiedades:

- **facturas:** lista de factura, de tipo List<Factura>, consultable.

Constructores:

- C1: recibe un parámetro de tipo Stream<Factura>.

Representación como cadena: una cadena con todas las facturas.

Criterio de igualdad: dos objetos de tipo Facturas son iguales si lo son sus facturas.

Se pide implementar en el paquete `fp.facturas` el tipo **EstadisticasFacturas**, que extienda a Facturas, y añada los siguientes tratamientos secuenciales. **EstadisticasFacturas** debe tener la misma representación como cadena que factura, pero añadiendo la información del número de facturas. (0,5 ptos).

Debe resolver todos los métodos mediante *streams*, salvo que se le indique expresamente que debe utilizar bucles:

1. **getPrecioTotalNFacturasMasRecientesCliente:** Dado un nombre de un cliente y un número entero *n*, devuelve la lista con los precios totales de las *n* facturas más recientes de ese cliente. (1 pto)
2. **getProductoMasCaroPosteriorFecha:** Dada una fecha, devuelve el producto de mayor precio de las facturas que se hayan realizado con posterioridad a esa fecha. Si no se puede calcular, eleva `NoSuchElementException`. (1 pto)
3. **getNumeroDeCategoriasDistintasPorCliente:** Devuelve un `SortedMap` en el que se asocia a cada cliente con el número de categorías distintas de productos compradas por ese cliente. **Implemente este método con bucles.** (1,5 ptos)
4. **getPorcentajeFacturasDePaisPorTipoCliente:** Dado un país, devuelve un `SortedMap` ordenado alfabéticamente que indique, para cada tipo de cliente (REGULAR, PREMIUM, VIP), qué porcentaje representan sus facturas respecto al total de facturas emitidas en ese país. (1,5 ptos)
5. **getGastoTotalSuperiorUmbralPorClienteConCategoria:** Dada una categoría y un umbral numérico, devuelve un `Map` en el que a cada cliente le hace corresponder su gasto total acumulado, contando solo aquellas facturas que incluyen al menos un producto de esa categoría. El `Map` resultado solo debe incluir aquellos clientes cuyo gasto total acumulado supere el umbral. (1,5 ptos)



Escriba en el paquete `fp.facturas.test` una clase **TestEstadisticasFacturas**. En la clase se leerán los datos del fichero y se probarán todos los tratamientos secuenciales, definiendo un método de test por cada tratamiento secuencial a probar. No se obtendrá la puntuación máxima del ejercicio si no se realiza el test y éste ejecuta.

Los resultados esperados para el dataset proporcionado, con los valores indicados en los tests, son:

EJ4.1 = = = = =

Los precios totales de las 3 últimas facturas del cliente Gary Santiago son:
[6680.982309999999, 2471.01444, 3765.96809]

Los precios totales de las 2 últimas facturas del cliente Lisa Byrd son:
[4835.6431999999995, 5171.878640000001]

EJ4.2 = = = = =

El producto más caro de las facturas posteriores a 2025-02-01 es:
Producto[nombre=Teclado mecánico, precio=1985.73, categoria=Accesorios]

EJ4.3 = = = = =

El número de categorías distintas por cliente es

{Aaron Barber=1, Alan Phillips=6, Alicia Hubbard=5, Allison Perez=6, Amanda Jones=2, Andrew Shaw MD=2, Anthony Howard=4, April Booth=1, Bernard Webster=2, Bob Pitts=1, Breanna Davis=3, Brett Schultz=2, Brian Barton=3, Brian Lee=3, Brittany Shepherd=4, Carlos Browning=1, Casey Anderson=6, Charles Pitts=4, Christina Cruz=2, Christine Clark=5, Christopher Jones=3, Christopher Parker=5, Cody Cox=3, Crystal Pham=3, Cynthia Rowe=4, Daniel Salinas=2, David Baker=1, David Grant=5, David Scott=3, Dawn Hensley=3, Dawn Summers=4, Debra Morton=2, Denise Davenport=5, Donna Perez=1, Dwayne Gilmore=5, Elizabeth Harrell=7, Elizabeth Oliver DDS=3, Elizabeth Perkins=1, Elizabeth Spence DDS=4, Ellen Alexander=1, Emily Smith=1, Eric Erickson=3, Eugene Higgins=5, Gary Santiago=9, Hailey Barajas=5, Haley Arnold=6, Holly Shaw=2, James Howard=5, Jamie Johnson=2, Jennifer Jones=4, Jeremy Reed=1, Jerry Christensen=4, Jessica Fox=3, Jessica Harrison=2, John McIntyre=2, Joseph Knight=4, Juan Reyes=5, Julie Alexander=5, Julie Petersen=5, Julie Roberts=4, Kathleen Aguilar=2, Kevin Wolf=1, Krista Gibson=4, Kyle Beard=2, Kyle Preston=5, Lisa Byrd=7, Maria Kerr=1, Maria Moore=3, Marie Christian=2, Mary Grimes=8, Matthew Hoover=1, Melissa Lee=2, Michael Berger=3, Michael Harrison=4, Michael Nelson=5, Michael Roberts=5, Michael Warner=8, Natasha Wall=3, Nathan Stewart=4, Nicole Garcia=5, Olivia Harris=2, Pamela Thompson=4, Peter Vaughn DDS=1, Rachel Perez=4, Regina Gonzalez=2, Richard Baker=2, Robert Costa=4, Samuel Rivas=2, Sarah Morgan=4, Sean Osborne=1, Sherry Shields=5, Sydney White=3, Terri Mason=3, Terry Evans=5, Timothy Kane=3, Travis Miller=3, Victor Vaughn=4, William Lynch=3, William Tran=1, Willie Durham=2}

EJ4.4 = = = = =

El porcentaje de facturas por tipo de cliente del país Estados Unidos es:
{REGULAR=21.73913043478261, PREMIUM=21.73913043478261, VIP=56.52173913043478}

El porcentaje de facturas por tipo de cliente del país España es: {REGULAR=20.0, PREMIUM=30.0, VIP=50.0}



EJ4.5 = = = = =

El gasto total de los clientes que superan el umbral 8000 en las facturas con al menos un producto de la categoría Muebles es: {Haley Arnold=8980.891969999999, Gary Santiago=11277.79626}

El gasto total de los clientes que superan el umbral 8000 en las facturas con al menos un producto de la categoría Electrónica es: {Casey Anderson=11587.630019999999, Haley Arnold=8980.891969999999, Mary Grimes=11441.22867}