

Enunciado

Se pretende desarrollar una aplicación con interfaz gráfica para la gestión de una agencia de viajes (tourist agency) que ofrece productos turísticos (tourist products) a los clientes. Los productos turísticos se caracterizan por tener un nombre único, una descripción, un precio de venta (double) y un posible descuento (double). Los productos turísticos pueden ser de los siguientes tipos: vuelos (flights), hoteles (hotels) y paquetes turísticos (tourist packages). Los paquetes turísticos están formados por una mezcla de hoteles, vuelos y/u otros paquetes turísticos. Se tiene también un registro de ventas (sales) de productos turísticos que recoge un listado de ventas realizadas.

El diseño de la aplicación que se plantea requiere la realización de las clases: **TouristProduct**, **Flight**, **Hotel**, **TouristPackage**, **Sale** y **Sales** en el paquete Java **touristagency.source**.

La clase abstracta **TouristProduct** establece las operaciones mínimas (abstractas o no) de los objetos que representan productos turísticos.

Métodos de TouristProduct	Descripción
TouristProduct (String, String)	Constructor al que se le pasa el nombre y la descripción.
String getName()	Devuelve el nombre.
String getDescription()	Devuelve la descripción.
double getPrice()	Devuelve el precio en euros.
double getPriceWithDiscount()	Devuelve el precio en euros aplicado el descuento.
double getDiscount()	Devuelve el descuento en euros. Por defecto es cero.
String toString()	Devuelve una String formada por el nombre del producto, seguido de dos puntos y un espacio (" "), seguido de la concatenación de su descripción seguido de un espacio, y de su precio con descuento seguido del símbolo del euro ("€").

Los objetos de la clase **Flight**, además de las características de **TouristProduct**, tienen un constructor propio y la posibilidad de establecer si el pasajero tiene un descuento por residencia.

Métodos de Flight	Descripción
Flight (String, String, double)	Constructor al que se le pasa el nombre, la descripción y el precio.
void setResident(boolean)	Establece si el pasajero tiene derecho a un descuento por residencia del 75%.

Los objetos de la clase **Hotel**, además de las características de **TouristProduct**, tienen un constructor propio y la posibilidad de establecer un descuento.

Métodos de Hotel	Descripción
Hotel (String, String, double)	Constructor al que se le pasa el nombre, la descripción y el precio.
void setDiscount(double)	Establece el descuento en euros.

Los objetos de la clase **TouristPackage** representan el conjunto de hoteles, vuelos y/u otros paquetes que forman un paquete turístico. El precio de un paquete turístico es la suma de los precios con descuento de sus componentes. Los objetos que componen un paquete turístico son referencias a los objetos que representan, pudiendo verse afectados por cambios en estos. Por ello, muchas operaciones deben calcularse en el momento, ya que el resultado podría variar en el tiempo. Por ejemplo: si un producto turístico de un paquete turístico cambia de precio o descuento, el paquete turístico debe reflejarlo. La clase **TouristPackage** requiere, además, de los siguientes métodos públicos:

Métodos de TouristPackage	Descripción
TouristPackage (String, String)	Constructor al que se le pasa el nombre y la descripción.
boolean add(TouristProduct)	Añade un producto turístico al paquete turístico. Devuelve verdadero si lo ha añadido, o falso si ya existe un producto con el mismo nombre.
boolean remove(String)	Elimina del paquete turístico el producto turístico con el nombre pasado por parámetro. Devuelve verdadero si lo elimina y falso si no se encuentra.
List< TouristProduct > getProducts()	Devuelve una lista con los productos que forman el paquete, ordenada por el nombre del producto turístico.
String toString()	Devuelve una String formada por el nombre del producto, seguido de dos puntos y un espacio (": "), seguido de la concatenación de su descripción seguido de la concatenación de los nombres de los productos de que está formado separados por una coma y un espacio en blanco (", ") entre los caracteres de abre y cierra corchete ("["..."]"), ordenados por el nombre, y seguido de un espacio, su precio con descuento y del símbolo del euro ("€").
void setDiscount(double)	Establece el descuento en euros.

Los objetos de la clase **Sales** representan una secuencia de ventas de productos turísticos. Las ventas se representan con objetos de la clase **Sale**. Los objetos de la clase **Sale** contienen el nombre del producto y el importe en el momento de la venta (ver código de la clase ya implementada). Se permite añadir ventas al final de la secuencia y quitar ventas por posición. Es posible obtener el importe total de las ventas de un producto y la lista de ventas ordenadas de mayor a menor precio.

Métodos de Sales	Descripción
void add(Sale)	Añade una venta al final de la secuencia.
void remove(int)	Elimina la venta ubicada en la posición indicada en el parámetro (las posiciones van de cero al total del número de ventas – 1). No hace nada si la posición está fuera de rango.
double totalSales(String)	Devuelve el importe correspondiente a la suma de las ventas directas del producto indicado en el parámetro.
List< Sale > salesOrderByPrice(int)	Devuelve las ventas ordenadas de forma decreciente, de mayor a menor precio, y a igual precio, en orden alfabético del nombre del producto. El parámetro establece el número máximo de ventas a devolver.
List< Sale > getAllSales()	Devuelve una lista con todas las ventas realizadas en el orden en que se añadieron.

Se suministra

Se dispone de las clases **TouristProduct**, **Flight**, **Hotel**, **TouristPackage**, **Sale** y **Sales** parcialmente desarrolladas. En el código suministrado en la actividad VPL se detallan los métodos públicos y qué funcionalidad tienen. Un programa principal en la clase **TouristAgency** para que pueda crear objetos de las clases desarrolladas y utilizar las operaciones de dichas clases. Revise el código suministrado para poder usarlo en el desarrollo de las funcionalidades solicitadas.

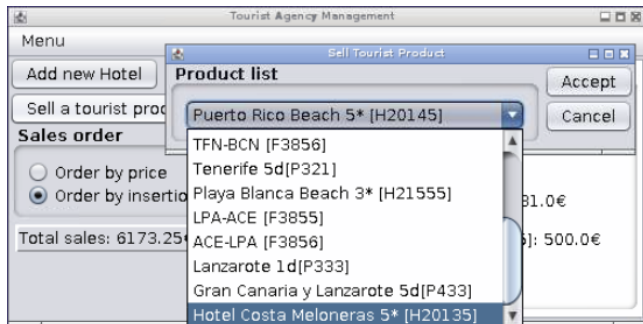
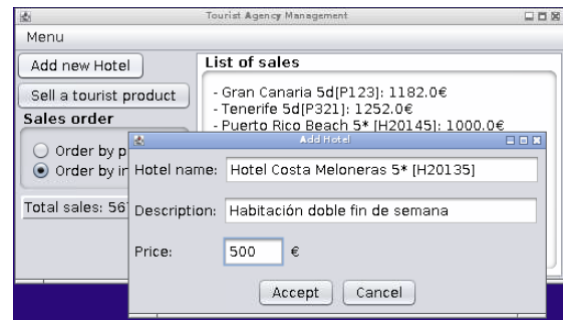
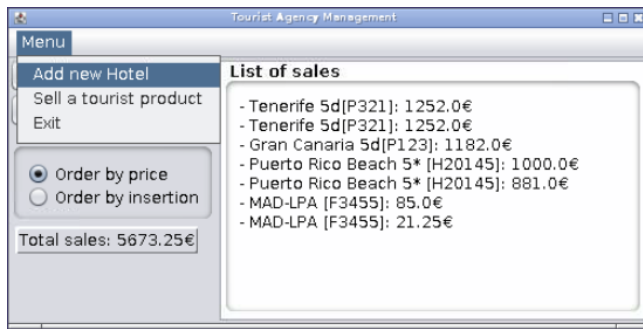
Parte 1 (60%)

Realice el desarrollo de las distintas clases según los requisitos descritos en el enunciado anterior. En este proceso puede necesitar añadir atributos, redefinir métodos de la clase **Object**, implementar comparadores, etc. Use los contenedores de la librería de Java más adecuados en cada caso. Las modificaciones que se hagan en los contenedores devueltos por los métodos en que esté especificado, no tienen que afectar a los contenedores de los objetos que requieran un contenedor.

Parte 2 (20%)

Desarrolle la interfaz gráfica que permita gestionar (parcialmente) la agencia de viajes, debiendo realizar lo siguiente:

- Mostrar la lista de ventas según el orden seleccionado.
- Añadir una nueva venta de un producto turístico actualizando los datos correspondientes.
- Añadir un nuevo hotel a los productos turísticos.
- La estética debe parecerse lo más posible a la que se muestra en las siguientes figuras.



En el diálogo para vender un producto turístico se usa un JComboBox inicializado con un array de Strings y tomando la opción seleccionada de su método `getSelectedIndex()`.

El resultado de la ejecución del método `main()` de la clase **TouristAgency** es el que sigue:

```
## - Products -
- Puerto Rico Beach 5* (H20145): Habitación doble del 2-8-19 al 7-8-19 1000.0€
- MAD-LPA (F3455): Turista 2-8-19 10:25am 85.0€
- LPA-MAD (F3456): Turista 7-8-19 18:55am 97.0€
- Gran Canaria 5d(P123): Gran Canaria salida desde Madrid [LPA-MAD (F3456), MAD-LPA (F3455), Puerto Rico Beach 5* (H20145)] 1182.0€
- Puerto de La Cruz Beach 5* (H20555): Habitación Lux del 12-8-19 al 17-8-19 1030.0€
- BCN-TFN (F3855): Business 12-8-19 10:25am 135.0€
- TFN-BCN (F3856): Turista 17-8-19 18:55am 87.0€
- Tenerife 5d(P321): Tenerife salida desde Barcelona [BCN-TFN (F3855), Puerto de La Cruz Beach 5* (H20555), TFN-BCN (F3856)] 1252.0€
- Playa Blanca Beach 3* (H21555): Habitación doble del 5-8-19 al 6-8-19 231.0€
- LPA-ACE (F3855): Turista 5-8-19 11:15am 37.0€
- ACE-LPA (F3856): Turista 6-8-19 20:55am 81.0€
- Lanzarote 1d(P333): Lanzarote salida desde Gran Canaria [ACE-LPA (F3856), LPA-ACE (F3855), Playa Blanca Beach 3* (H21555)] 349.0€
- Gran Canaria y Lanzarote 5d(P433): Gran Canaria y Lanzarote salida desde Madrid [Gran Canaria 5d(P123), Lanzarote 1d(P333)] 1531.0€

## - Products with discount -
- Puerto Rico Beach 5* (H20145): Habitación doble del 2-8-19 al 7-8-19 881.0€
- MAD-LPA (F3455): Turista 2-8-19 10:25am 21.25€
- LPA-MAD (F3456): Turista 7-8-19 18:55am 97.0€
- Gran Canaria 5d(P123): Gran Canaria salida desde Madrid [LPA-MAD (F3456), MAD-LPA (F3455), Puerto Rico Beach 5* (H20145)] 999.25€
- Puerto de La Cruz Beach 5* (H20555): Habitación Lux del 12-8-19 al 17-8-19 1030.0€
- BCN-TFN (F3855): Business 12-8-19 10:25am 135.0€
- TFN-BCN (F3856): Turista 17-8-19 18:55am 87.0€
- Tenerife 5d(P321): Tenerife salida desde Barcelona [BCN-TFN (F3855), Puerto de La Cruz Beach 5* (H20555), TFN-BCN (F3856)] 1252.0€
- Playa Blanca Beach 3* (H21555): Habitación doble del 5-8-19 al 6-8-19 231.0€
- LPA-ACE (F3855): Turista 5-8-19 11:15am 37.0€
- ACE-LPA (F3856): Turista 6-8-19 20:55am 81.0€
- Lanzarote 1d(P333): Lanzarote salida desde Gran Canaria [ACE-LPA (F3856), LPA-ACE (F3855), Playa Blanca Beach 3* (H21555)] 349.0€
- Gran Canaria y Lanzarote 5d(P433): Gran Canaria y Lanzarote salida desde Madrid [Gran Canaria 5d(P123), Lanzarote 1d(P333)] 1348.25€

## - Sales -
- Gran Canaria 5d(P123): 1182.0€
- Tenerife 5d(P321): 1252.0€
- Puerto Rico Beach 5* (H20145): 1000.0€
- MAD-LPA (F3455): 85.0€
- Tenerife 5d(P321): 1252.0€
- Puerto Rico Beach 5* (H20145): 881.0€
- MAD-LPA (F3455): 21.25€

## - Best Sales -
- Tenerife 5d(P321): 1252.0€
- Tenerife 5d(P321): 1252.0€
- Gran Canaria 5d(P123): 1182.0€
- Puerto Rico Beach 5* (H20145): 1000.0€
```