



“Aterrizar” es una empresa orientada a la venta de viajes de diferente naturaleza (terrestres, marítimos y aéreos) y requiere un sistema modelado e implementado observando los principios de la POO para administrar su catálogo en vigencia.

A tal efecto, se solicita proveer las siguientes clases, cada una con las características que se indican:

1) Clase CPaquete, que presenta:

- a) Los datos miembros de instancia:
 - i) **Código:** cadena de caracteres.
 - ii) **Origen:** cadena de caracteres.
 - iii) **Destino:** cadena de caracteres.
 - iv) **Precio:** Monto (\$) a abonar por el viaje. Número real (punto flotante de precisión **simple**).
 - v) **Interes:** Es un atributo común a todos los objetos y es a tasa de interés que se cobra por las cuotas.
- b) Un método constructor parametrizado que recibe como argumentos el código, el origen y el destino del viaje.
- c) Los métodos **GetOrigen()**, **GetDestino()** y **GetCodigo()**.
- d) Una **propiedad** de lectura y escritura **PrecioViaje**, que permita acceder a la variable de instancia asociada al precio del viaje, tanto para el establecimiento del monto (\$) como para la obtención del mismo.
- e) Un método **darPrecio(cuotas)** que recibiendo como argumento la cantidad de cuotas: 1, 3, 6 o 12, devuelve como número real de precisión simple, el precio a pagar, según la aplicación de un recargo de una tasa de interés que se multiplica por los meses. (Si la tasa de interés es 3%, en tres cuotas será 9%).
- f) Un método redefinible **darDatos()**, que sin argumento alguno, devuelve una cadena de caracteres que concatena los valores de todos los datos miembros.
- g) La clase debe implementar las interfaces IComparable para comparar por precios e IEquatable para poder buscar por códigos.

2) Además se deben realizar dos clases que hereden los atributos de la clase CViaje: CPaqueteEstadia y CPaqueteAuto.

CPaqueteEstadia deberá presentar:

- a) **NombreHotel:** cadena de caracteres.
- b) **Cantidad de noches:** Entero sin signo.
- c) **Costo de la habitación:** Es el monto a pagar por la habitación del hotel por cada noche.

Y los siguientes métodos:

- a) Getters y setters para los nuevos atributos.
- b) Se debe redefinir darDatos() para contemplar los nuevos datos.
- c) Se debe redefinir darPrecio(cuotas) para que calcule el precio del viaje teniendo en cuenta el precio de la habitación y la cantidad de noches.

CPaqueteAuto deberá presentar:

- a) **Patente:** cadena de caracteres.
- b) **ContrataSeguro:** Booleano que indica si el cliente contrató seguro o no.
- c) **CostoSeguro:** Atributo estático que indica el costo del seguro del vehículo.
- d) **CantidadDias:** Dias que el auto será alquilado.
- e) **CostoPorDia:** Es el costo por día del vehículo.

Y los siguientes métodos:

- a) Getters y setters para los nuevos atributos.
- b) Se debe redefinir el método darDatos() para contemplar los nuevos datos.

- c) Se debe redefinir darPrecio(cuotas) para que calcule el precio del viaje teniendo en cuenta el precio del vehículo, la cantidad de días y si contrato o no seguro.
- 3) Se pide realizar un programa que permita el ingreso indefinido de los dos tipos de viajes en dos listas separadas (una para viajes con estadia y otra para viajes con auto). El ingreso finalizara con el código 0 para ambos. Una vez finalizado se pide mostrar ambas listas ordenadas por precio.
- 4) Se pide realizar otro programa que permita el ingreso indefinido de viajes con estadía y auto en una sola lista (el ingreso finalizara con el código 0). Una vez que se termine el ingreso de datos, se permitirá al usuario ingresar un código de viaje y mostrar sus datos (tener en cuenta que debemos parsear a viaje con auto o con estadía), luego se debe solicitar al usuario que ingrese los datos de ese viaje para modificarlo. Este segundo ingreso de códigos finalizara con el código 0. Antes de finalizar el programa se deberá mostrar la lista completa de viajes ordenados por precio.

Nota: Para los ejercicios 3 y 4 se deben realizar dos clases aparte de las mencionadas: interfaz (clase estatica que se dedique a controlar el ingreso y egreso de datos) y una clase controladora que maneje las listas de datos y la lógica del programa.

V. P-20-2P-1C-TU- S.d.J.