

Programación

Clases derivadas

Práctica 3.c

Se desea realizar una aplicación Java que nos permita gestionar las consumiciones de un hotel.

El hotel dispone de dos tipos de clientes diferentes: normales y todo incluido. Al final de la estancia el hotel cobrará, en el caso de los clientes normales, aquellas consumiciones que haya realizado. Para los clientes con todo incluido, se le cobrará una cuota de 100€ más las consumiciones que no entren en el plan todo incluido. Por lo tanto, el hotel necesita llevar la cuenta de todas las consumiciones que ha realizado cada cliente dentro del complejo. Para identificar a cada cliente, se registra su dni y su nombre.

NOTA: hay que utilizar herencia para la realización de esta parte. Será necesario crear las clases junto con los atributos y métodos necesarios.

Las consumiciones pueden ser artículos de cualquier tipo, ya sean bebidas, comidas, visitas al spa, souvenirs, etc. Cada producto estará o no etiquetado con una pegatina de "Todo Incluido" que indicará si ese producto se aplica a dicha tarifa. Si un cliente con la tarifa todo incluido adquiere un servicio que incluye la pegatina "Todo Incluido", no se le cargará su coste al saldo del mismo. Si por el contrario adquiere un producto sin dicha pegatina, sí se le cargará ese coste a su saldo. Para un cliente normal, se le cargan a su saldo todas las consumiciones.

Artículo

Todo artículo dispondrá de los siguientes parámetros. Serán de lectura, sus valores se asignarán únicamente en el constructor.

- **código:** código alfanumérico que permite identificar el servicio.
- **precio :** precio del producto.
- **concepto:** descripción breve del producto adquirido. Ejemplos: "Refresco Coca Cola", "Llavero de Salamanca"
- **pegatina TodoIncluido:** indica si se le aplica, o no, la tarifa "Todo Incluido" al producto.

El hotel dispondrá de una lista de clientes, así como de un listado de todos los artículos disponibles. Se podrá realizar las siguientes acciones:

- **addCliente**
Esta función se encargará de incluir un nuevo cliente en la lista de clientes del hotel. Recibirá como parámetros el dni y el nombre. Deberá comprobar si el cliente ya existe y retornará si se ha podido incluir el cliente o no.
- **addClienteTodoIncluido**
Esta función se encargará de incluir un nuevo cliente con la tarifa "Todo Incluido" en la lista de clientes del hotel. Recibirá como parámetros el dni y el nombre. Deberá comprobar si el cliente ya existe. La función retornará sí se ha podido incluir el cliente o no.
- **addArticulo**
Esta función se encargará de incluir un nuevo artículo en el listado de artículos del hotel. Recibirá como parámetros: código, precio, concepto y si incluye la pegatina "Todo Incluido". Debe comprobar que no existe otro artículo con el mismo código. La función debe devolver si se ha podido incluir el artículo.

- **realizarConsumicion**

Esta función recibirá como parámetros el dni, código de la consumición y cantidad. Se encargará de añadir el numero de consumiciones indicadas al cliente teniendo en cuenta su tarifa.

En caso de error la función retornará:

Valor devuelto	significado
0	Sin errores
-1	No existe el cliente
-2	No existe el artículo

- **facturar**

Este método recibirá como parámetro el dni del cliente. Se encargará de visualizar por pantalla el saldo del cliente indicado. Mostrará un mensaje de error en el caso de que no se encuentre al cliente.

- **mostrarListado**

Mostrará un listado de todos los artículos de los que dispone el hotel.

Para probar el correcto funcionamiento del programa, se debe crear un ejemplo desde la función main. Es decir, se añadirán varios clientes y artículos. Posteriormente se realizaran una serie de consumiciones de dichos clientes, que finalmente van a ser facturados.