

Práctica 1

Introducción a Java



3º Ingeniería Informática
“Programación Web”

Componentes del grupo:

- Daniel Hinojosa Sánchez - i02hisad@uco.es
- Martín Del Río Jiménez - i02rijim@uco.es
- Juan Antonio Gálvez Jiménez - i02gajia@uco.es
- Marta Rubio Sánchez - i82rusam@uco.es
- Miguel Castro Martín - i82casmm@uco.es

Índice

1 - Introducción	3
2 - Clases	3
2.1 Kart	3
2.2 Pista	3
2.3 Usuario	3
2.4 Reserva	3
2.5 Gestor de Usuarios	4
2.6 Gestor de Pistas	4
2.7 Gestor de Reservas	4
2.8 Ficheros	4

1 - Introducción

En esta primera práctica hemos tenido en cuenta la importancia que supone dividir las clases pertenecientes a cada una de las funcionalidades como el hecho de usar las colecciones adecuadas para facilitar el manejo de los datos respecto a las reservas y a los usuarios.

Por consiguiente, vamos a explicar cada una de las clases que han hecho posible este proyecto, especificando de manera más detallada los aspectos más importantes de las mismas.

Además cabe mencionar que debido a que algunas funcionalidades de la librería Date se encuentran en desuso, hemos decidido definir nuestras fechas en base a la librería LocalDate.

Nuestro programa tendrá un menú principal desde el cual acceder a los distintos gestores, los cuales contarán con un método menú desde el cual elegir la funcionalidad deseada del gestor en cuestión.

2 - Clases

2.1 Kart

En esta primera clase podemos observar los atributos referentes a un kart así como una única funcionalidad que nos muestra la información de cada kart.

2.2 Pista

En dicha clase, nos encontramos con los atributos pertenecientes a una pista. Tenemos tres funcionalidades: imprimir la información, consulta de karts disponibles; en la cual hemos utilizado el array de karts para buscar en él cuáles son los disponibles, y otra para asociar kart a pista en el cual mediante un método booleano vemos que siendo verdadero obtenemos el kart para adultos y falso para infantil. Para ello añadimos a una lista el kart asociado.

2.3 Usuario

En esta clase nos encontramos con los atributos pertenecientes a un usuario. Tenemos tres funcionalidades, la primera imprime la información de un usuario, la segunda se utiliza para realizar el cálculo de la mayoría de edad mediante métodos booleanos para un uso posterior en la asignación de karts, y por último la tercera indica la antigüedad del usuario registrado, el cual más tarde conlleva a beneficiarse de algunos descuentos. En estas dos últimas hemos usado el método ChronoUnit en función de años.

2.4 Reserva

La clase reserva comienza definiendo los atributos para almacenar una reserva. Para empezar tenemos que saber que la clase reserva se divide en tres tipos de la misma. En este caso son: Reserva Infantil, Reserva Familiar y Reserva Adultos, las cuales pueden dividirse en dos modalidades diferentes que son: Reserva Individual y Reserva Bono. Las cuales se han creado a través de un patrón factoría. Este patrón factoría consta de una clase abstracta "Reserva" en la cual definimos los parámetros que definen a una reserva, así como los métodos funcionales básicos y el método precioFinal(), el cual nos permite calcular el precio de la reserva con el descuento aplicado. De esta clase abstracta extenderán las clases concretas Reserva familiar, infantil y adulto.

Por otro lado tendremos la clase abstracta CreadorReservas en la cual se definen los métodos que se encargan de crear las reservas. Finalmente estos métodos serán implementados en la clases concretas de Individual y Bono que extienden de la clase anterior.

2.5 Gestor de Usuarios

En esta clase podemos observar los atributos relacionados con la gestión de usuarios. Usamos para todas las funciones un array para observar el listado de los usuarios y podemos trabajar con ellos.

2.6 Gestor de Pistas

En dicha clase vemos que los atributos están directamente relacionados con la administración de las pistas y de los karts. Volvemos a usar un array para observar el listado de las pistas y karts. Mediante la función switch, se piden los datos necesarios para asociar los karts necesarios disponibles con alguna pista disponible en ese momento.

2.7 Gestor de Reservas

En esta clase podemos ver la gestión de las reservas de pistas por parte de los usuarios. Se ha creado un menú mediante el cual podemos elegir las distintas opciones que se nos pide para este apartado. Al igual que en la clase anterior, escogemos la opción de utilizar un switch para gestionar de mejor forma las diferentes funcionalidades.

2.8 Ficheros

Por último, hemos decidido crear una clase Ficheros desde la cual definir e implementar métodos que se encargan de cargar datos desde ficheros y escribir en los mismos. Estos métodos se utilizarán en los distintos gestores al interactuar con los ficheros.

Si bien esta funcionalidad y clase no viene especificada en el enunciado de la práctica, hemos considerado importante añadirla de cara a mantener un código lo más ordenado, simple y modularizado posible.

En los ficheros se ha seguido un formato en el cual cada línea será un elemento. Esta línea contiene las variables/datos separadas por punto y coma.