

Tarea

Tu tarea una vez acabado el tercer módulo consiste en:

- Crear un proyecto llamado `Modulo3NombreApellido` donde Nombre sea tu nombre y Apellido tu primer apellido. Ejemplo: `Modulo3PabloRuiz`
- En el proyecto deberá estar organizado por paquetes de modo que tengas, al menos, los paquetes `clase`, `clase_abstracta`, `interf` y `lanzador`.
- Dentro del paquete `lanzador` deberás crear una clase llamada `Principal`. En esta clase estará el método `main`.
- Vamos a modelizar un tienda de bicicletas.
 - Tendrás una clase abstracta llamada `Bicicleta`
 - Tendrás unas subclases de `Bicicleta` llamadas `BiciMontania`, `BiciPaseo`, `Tandem`. Estas subclases deberán sobrescribir al método `toString` de `Object` haciendo que digan de que tipo son, su color y precio.
 - Todas las bicicletas tendrán dos campos comunes `color` (`String`) y `precio` (`double`).
 - Quiero que todas las bicicletas implementen el método `pintar`, que lo que hará será cambiar el color pero me interesa que este método `pintar` también se pueda aplicar a otras posibles modelizaciones futuras que pudiera hacer, no debe ser exclusivo de las bicicletas. Además quiero que el coste de pintar (lo que sea) sea fijo y sea 90.
 - Las bicicletas de montaña deberán almacenar la marcha que tiene la bici (pudiendo tomar valores únicamente entre 1 y 6).
 - Las bicicletas de paseo deberán almacenar la velocidad a la que se circula.
 - Los tandem deberán almacenar el número de asientos (pudiendo tomar únicamente los valores 2 y 3).
 - Todos los campos de todas las clases deberán seguir los principios de encapsulamiento.
 - La clase `BiciMontania` deberá tener un único constructor a través del cual se establezca el valor del color, precio y marcha.
 - La clase `BiciPaseo` deberá tener 2 constructores, uno a través del cual se establezca el valor del color, precio y velocidad y otro a través del cual se establezca el color y precio.
 - La clase `Tandem` deberá tener un único constructor a través del cual se establezca el valor del color, precio y número de asientos.
 - Los constructores deberán controlar las condiciones existentes.
 - La `BiciMontania` tendrá un método llamado `aumentarMarcha` que no tendrá parámetros y aumentará en 1 la marcha actual (siempre que cumpla las condiciones anteriores). También tendrá un método llamado `disminuirMarcha` que no tendrá parámetros y reducirá en 1 la marcha actual (siempre que cumpla las condiciones

anteriores). No se podrá modificar la marcha desde ningún otro lugar.

- En la clase Principal deberás crear una lista con todas las bicicletas que tengas en la tienda. En este momento tienes 2 bicicletas de montaña, 1 bici de paseo y 2 tandems. Crea los objetos y añádelos a la lista. Posteriormente interacciona con ellos y al final recorre la lista para llamar al método toString de cada objeto.

Revision #1

Created 1 February 2022 11:11:22 by Equipo CATEDU

Updated 1 February 2022 11:11:22 by Equipo CATEDU