

Ejercicios



Ejercicio 1

Crea las clases `Animal`, `Mamifero`, `Ave`, `Gato`, `Perro`, `Canario`, `Pinguino` y `Lagarto`. Crea, al menos, tres métodos específicos de cada clase y redefine el/los método/s cuando sea necesario. Prueba las clases en un programa en el que se instancien objetos y se les apliquen métodos. Puedes aprovechar las capacidades que proporciona HTML y CSS para incluir imágenes, sonidos, animaciones, etc. para representar acciones de objetos; por ejemplo, si el canario canta, el perro ladra, o el ave vuela.



Ejercicio 2

Crea la clase `Vehiculo`, así como las clases `Bicicleta` y `Coche` como subclases de la primera. Para la clase `Vehiculo`, crea los métodos de clase `getVehiculosCreados()` y `getKmTotales()`; así como el método de instancia `getKmRecorridos()`. Crea también algún método específico para cada una de las subclases. Prueba las clases creadas mediante una aplicación que realice, al menos, las siguientes acciones:

- Anda con la bicicleta
- Haz el caballito con la bicicleta
- Anda con el coche
- Quema rueda con el coche
- Ver kilometraje de la bicicleta
- Ver kilometraje del coche
- Ver kilometraje total



Ejercicio 3

Crea la clase `DadoPoker`. Las caras de un dado de poker tienen las siguientes figuras: As, K, Q, J, 7 y 8. Crea el método `tira()` que no hace otra cosa que tirar el dado, es decir, genera un valor aleatorio para el objeto al que se le aplica el método. Crea también el método `nombreFigura()`, que diga cuál es la figura que ha salido en la última tirada del dado en cuestión. Crea, por último, el método `getTiradasTotales()` que debe mostrar el número total de tiradas entre todos los dados. Realiza una aplicación que permita tirar un cubilete con cinco dados de poker.



Ejercicio 4

Queremos gestionar la venta de entradas (no numeradas) de **Expocoches Campanillas** que tiene 3 zonas, la sala principal con 1000 entradas disponibles, la zona de compra-venta con 200 entradas disponibles y la zona vip con 25 entradas disponibles. Hay que controlar que existen entradas antes de venderlas. Define la clase `Zona` con sus atributos (`nombre_zona`, `plazas_totales`, `plazas_restantes`) y métodos correspondientes (getters, `toString`, `vender`) y crea un programa que permita vender las entradas. En la pantalla principal debe aparecer información sobre las entradas disponibles y un formulario para vender entradas. Debemos indicar para qué zona queremos las entradas y la cantidad de ellas. Lógicamente, el programa debe controlar que no se puedan vender más entradas de la cuenta.

Expocoches Campanillas

Bienvenido a Expocoches Campanillas. ¿Cuántas entradas desea comprar?

Zona: Principal | Numero de entradas: | Comprar

Zona principal: 1000 de 1000 entradas.

Zona compra-venta: 200 de 200 entradas.

Zona VIP: 25 de 25 entradas.