



BOOTCAMP BACKEND C#

GUÍA EJERCITACIÓN - C# AVANZADO

1 - Crear la clase Rectángulo, definir dos propiedades: Lado1 y Lado2. Definir dos métodos RetornarSuperficie y RetornarPerimetro. Dividir la clase en dos archivos utilizando el concepto de "partial class".

2 - Crear la clase Vehículo, definir las propiedades estadoMotor, velocidad y los métodos Arrancar() , Frenar(), Acelerar() y Apagar(). Dividir la clase en distintos archivos utilizando el concepto de "partial class".

3 - Crear la clase Persona, definir las propiedades nombre, fecNacimiento y los metodos Edad() , GetNombre(). Dividir la clase en distintos archivos utilizando el concepto de "partial class".

4 - Crear la clase CuentaCorriente, definir las propiedades

- string moneda,
- string propietario
- double saldo

y los metodos GetMoneda() , GetPropietario() y GetSaldo().

Dividir la clase en distintos archivos utilizando el concepto de "partial class".

Si ahora decimos que propietario es del dato del tipo Persona (string nombre, Date fecNacimiento). ¿Cómo harían la clase parcial?

5 - Crear la clase Pila que permite apilar y desapilar elementos. Realizarla en forma genérica para que pueda trabajar con cualquier tipo de datos.

Realizar un programa de consola que apile y desapile 10 elementos. Hacer esto para diferentes tipos de datos.

6 - Crear la clase Cola que permite encolar y desencolar elementos. Realizarla en forma genérica para que pueda trabajar con cualquier tipo de datos.

Realizar un programa de consola que encole y descole 10 elementos. Hacer esto para diferentes tipos de datos.

7 - Crear una interfaz genérica llamada ListaDevPlace que tenga los siguientes métodos:

- AgregarItem() que agrega elementos a la lista,
- BorrarItem() que elimina elementos de la lista,
- ObtenerItem(int i) que devuelve el elemento en el índice determinado. Realizar las validaciones necesarias.
- CantidadItems() devuelve la cantidad de ítems que contiene la lista

Realizar al menos dos clases que implementen dicha interfaz con tipos de datos diferentes.

8 - Agregar dos métodos a la clase string que permitan recuperar la primera mitad y la segunda mitad de la cadena.

9 - Agregar un método de extensión a la clase Persona que permita saber si la persona es mayor de edad.

10 - Agregar un método de extensión a la clase List que permita imprimir todos los elementos de la lista. El método de extensión debe ser genérico.

11 - Agregar un método de extensión a la clase `int` que permita imprimir desde el valor actual que tiene la variable entera hasta el valor que le pasemos como parámetro, de uno en uno.

12 - Agregar un método de extensión a la clase `double` con un método para devolver los centavos y otro métodos para devolver la parte entera.