**EJERCICIOS BÁSICOS DE APLICACIÓN**

1. INDIZADORES. Implementa la clase autobús, la clase autobús dispondrá de un array de booleanos para indicar si una plaza está ocupada, el número de plazas será necesario indicarlo a la hora de instanciar un autobús. Se podrá saber si una plaza está ocupada a través de un indizador que reciba como parámetro el número de plaza y devuelva verdadero o falso en función de si el asiento está ocupado o no.

Implementar una propiedad Ocupadas que me indique el número de plazas reservadas, otra propiedad Plazas que me indique el número de plazas totales del autobús. Una función PlazasLibres que me devuelva los números de plaza libres y otra función ReservarPlaza que reciba el número de plaza y trate de reservarla, si la plaza ya está ocupada devolverá false y se notificara al usuario.

Crear el código necesario para poder probar toda la funcionalidad requerida ofertándola al usuario a través de un menú de selección.

1. Crea una interfaz llamada IDispositivoElectronico que tenga los métodos Encender y Apagar. Luego crea las clases Televisión y Radio, las cuales implementará esta interfaz. Ambas clases deben tener una propiedad Estado que indique si el dispositivo está encendido o apagado Escribe un programa que instancie objetos de ambas clases, los encienda, y luego imprima el estado actual del dispositivo
2. Crea una interfaz IPago que tenga un método RealizarPago(). Luego, crea dos clases PagoTarjeta y PagoPaypal que implementen esta interfaz. Cada una debe imprimir un mensaje distinto cuando se realice el pago. Escribe un programa que simule el pago con tarjeta y el pago con Paypal.
3. Crea una interfaz ITransporte con los métodos Arrancar() y Detener(). Luego, crea una clase base Vehiculo que tenga una propiedad Marca y un método MostrarInformacion() que muestre la marca del vehículo.

A partir de Vehiculo, crea dos clases derivadas: Coche y Moto, que implementen la interfaz ITransporte. Ambas clases deben proporcionar una implementación para los métodos de la interfaz, además de heredar el método MostrarInformacion() de la clase base.

Escribe un programa que cree un objeto de tipo Coche y otro de tipo Moto, llame a sus métodos de la interfaz y muestre la información de cada vehículo.

1. Crea una interfaz ITrabajador que tenga los métodos Trabajar() y CobrarSalario(). Luego, crea una clase base Persona que tenga las propiedades Nombre y Edad, y un método MostrarInformacion() que muestre el nombre y la edad de la persona.

A partir de la clase Persona, crea dos clases derivadas: Empleado y Freelancer, que implementen la interfaz ITrabajador. El método Trabajar() debe mostrar mensajes diferentes para un empleado y un freelancer, y el método CobrarSalario() debe reflejar sus diferencias, por ejemplo, un empleado cobra un salario fijo y un freelancer cobra por proyecto.

Escribe un programa que cree un objeto de tipo Empleado y otro de tipo Freelancer, muestre su información y haga que trabajen y cobren su salario.

1. Crea una interfaz llamada IVolador que tenga un método Volar(). Luego, crea dos clases, Ave y Avion, que implementen esta interfaz. Escribe un programa que haga volar tanto a un ave como a un avión.
2. Interfaces, ordenación, IComparable Crea una clase Mascota que disponga de las propiedades nombre y edad, la clase implementara la interfaz IComparable la cual permite comparar dos objetos. Después se pedirá al usuario los datos de 5 mascotas y el programa mostrará sus nombres ordenados por edad usando el método CompareTo de la interfaz.
3. Interfaces, ordenación, IComparer. Crea una clase Cliente con las propiedades nombre, importe facturación, fecha de alta, fecha de ultima venta. Los campos de tipo fecha serán del tipo DateTime y solo se tendrán en cuenta el día, mes y año. Implementar mediante IComparer un comparador para cada una de las propiedades. Crear una aplicación que mediante un menú me permita dar de alta hasta 10 clientes y mostrarlos ordenados según la propiedad que indique el usuario.
4. Crea una clase llamada Aula que represente un aula con un número determinado de alumnos. La clase debe tener un indizador que permita acceder a los alumnos por su índice
5. Crea una clase llamada Libro que contenga una colección de capítulos. Implementa un indizador que permita acceder a los capítulos por su índice.
6. Crea una clase llamada Producto que tenga propiedades para Nombre y Precio. Luego implementa un comparador que ordene los productos por precio
7. Crea una clase llamada Libro con propiedades para Titulo y Autor. Luego, implementa un comparador que ordene los libros alfabéticamente por título.