

Proyecto 2 Tema 3 - Sistema de Gestion Tesla - SpaceX

Descripción del código:

Program.cs:

Este archivo contiene el punto de entrada de la aplicación. Establece la configuración básica y lanza la aplicación al iniciar el formulario principal Form1.

Form1.cs:

1. Define la clase Form1, que representa el formulario principal de la aplicación.
2. El formulario contiene varios controles, como botones, etiquetas, cuadros de texto, una lista desplegable (combobox) y un control DataGridView.
3. Existe una lista llamada "productos" que almacena objetos de la clase Producto.
4. El método Form1_Load se ejecuta cuando el formulario se carga. En este método se agregan algunos productos a la lista y se configuran los controles en el formulario.
5. El método MostrarProductosEnLista muestra los productos en el control DataGridView.
6. El método BtnAgregar_Click se ejecuta cuando se hace clic en el botón "Agregar". Crea un nuevo objeto de producto basado en la selección del combobox y los valores ingresados en los cuadros de texto, lo agrega a la lista y actualiza el DataGridView.
7. El método BtnEliminar_Click se ejecuta cuando se hace clic en el botón "Eliminar". Elimina el producto seleccionado de la lista y actualiza el DataGridView.
8. El método BtnTeslaMasKm_Click se ejecuta cuando se hace clic en el botón "Tesla con más km". Encuentra el Tesla con la mayor cantidad de kilómetros y muestra su información en un cuadro de mensaje.
9. Hay otros métodos, como LimpiarCamposEntrada, FormatearHeader, LstProductos_MouseClick y BtnEscanear_Click, que realizan diferentes tareas relacionadas con la interfaz de usuario y la manipulación de datos.
10. Algunos métodos y eventos están relacionados con la interacción con el control DataGridView y la selección de productos.

Productos.cs:

Este archivo incluye las clases base abstractas Producto, TeslaBase y SpaceXBase, junto con las clases concretas TeslaModelX, TeslaModelS, TeslaCybertruck, SpaceXStarship y SpaceXFalcon9, que heredan de sus respectivas clases base.

La clase Producto es una clase base abstracta que representa un producto genérico. Contiene propiedades comunes como Modelo, Año, UnidadDeUso, Color, Dueño, Autonomía y Service. También tiene una propiedad calculada llamada CargaRestante, que calcula el porcentaje de carga restante en función de la autonomía y la unidad de uso del producto (kilómetros u horas de vuelo según corresponda). Define un método abstracto llamado ObtenerInformacion() para obtener información específica del producto y un método virtual llamado RealizarEscaneeo() para realizar el escaneo del producto.

La clase TeslaBase es una clase base abstracta para los productos de Tesla, que extiende la clase Producto. Proporciona una implementación para el método abstracto ObtenerResultadoEscaneeo() específico de los productos de Tesla. Calcula la cantidad de servicios realizados en función de la unidad de uso y el intervalo de servicio, y verifica varios componentes durante cada servicio. Cada

modelo específico de Tesla (TeslaModelX, TeslaModelS, TeslaCybertruck) anula el método ObtenerInformacion() y agrega propiedades específicas del modelo.

De manera similar, la clase SpaceXBase es una clase base abstracta para los productos de SpaceX, que también extiende la clase Producto. Proporciona una implementación para el método abstracto ObtenerResultadoEscaneo() específico de los productos de SpaceX. Calcula la cantidad de servicios realizados en función de la unidad de uso y el intervalo de servicio, y verifica los sistemas de propulsión y navegación durante cada servicio. Cada modelo específico de SpaceX (SpaceXStarship, SpaceXFalcon9) anula el método ObtenerInformacion() y agrega propiedades específicas del modelo.

En general, el código proporciona una estructura para gestionar y escanear diferentes tipos de productos, en este caso, automóviles Tesla y cohetes SpaceX. Cada clase derivada define las comprobaciones específicas y genera un informe de los servicios realizados mediante el método ObtenerResultadoEscaneo().