

Registro de Naves Galácticas

Parte 1: Implementación de Clases

1. Clase Pasajero

- Atributos privados:
 - nombre (string): Debe ser una cadena no vacía y de 5 caracteres como mínimo.
- Métodos:
 - constructor(nombre): Inicializa el atributo validando que el nombre no esté vacío.
 - toString(): Devuelve una representación textual del pasajero.
 - Setters y getters con validaciones que lancen errores si los valores no cumplen las restricciones.

2. Clase HeroeEstelar (hereda de Pasajero)

- Atributos privados adicional:
 - naturaleza (string): Solo puede ser "humano" o "no humano".
- Métodos adicionales:
 - Setters y getters con validaciones.
 - toString(): Devuelve una representación textual del héroe.

3. Clase Robot (hereda de Pasajero)

- Atributos privados adicionales:
 - puedeHablar (boolean): Indica si el robot tiene capacidad para hablar.
- Métodos adicionales:
 - Setters y getters con validaciones.
 - toString(): Devuelve una representación textual del robot.

4. Clase Nave

- Atributos privados:
 - nombre (string): Debe ser una cadena no vacía.
 - heroes (array de objetos HeroeEstelar): Debe contener al menos 2 héroes.
 - robot (objeto Robot): Solo puede haber un robot por nave.
- Métodos:
 - constructor(nombre, heroes, robot): Inicializa los atributos validando las reglas.
 - Setters y getters con validaciones.
 - toString(): Devuelve una descripción de la nave, incluyendo sus héroes y robot.

Parte 2: Formulario de Registro

Estructura del Formulario:

- Datos de la Nave:
 - Campo nombre de la nave: Obligatorio, debe ser una cadena no vacía de 5 caracteres como mínimo. Realizar esta comprobación en el HTML.
- Selección de Personajes:
 - Lista de imágenes con checkboxes (mínimo 2 seleccionados).
- Selección de Robot:
 - Lista de imágenes con radio buttons (obligatorio seleccionar uno).

Botón:

- Lanzar Nave.

Parte 3: Validaciones

- El nombre de la nave debe ser obligatorio y de tamaño mínimo de 5 caracteres.
- Deben seleccionarse al menos 2 personajes.
- Debe seleccionarse un robot.

Parte 4: Interactividad y Validaciones

- Al hacer clic en "Lanzar Nave":
 - Crear instancias de HeroeEstelar y Robot basándose en las selecciones del usuario.
 - Crear una instancia de Nave y mostrar los detalles en la consola.

- Validar los datos:
 - Lanzar errores si no se cumplen las reglas (e.g., no se seleccionan suficientes personajes).
 - Mostrar mensajes de error en el formulario si algo falla.
- Si los datos son válidos:
 - Mostrar un mensaje de éxito indicando que la nave ha sido lanzada.

Entrega

- Debes entregar los siguientes archivos:
- Pasajero.js, HeroeEstelar.js, Robot.js y Nave.js con las clases implementadas.
- index.html con el formulario.
- index.js con la lógica de validación e interacción.