Ejercicios TypeScript

Nota: se puede usar un editor de JavaScript para resolver algunos de los ejercicios o el playground predefinido por TypeScript.

ejercicio0001.ts

- Declarar variables y constantes de tipos de datos simples y arrays. Experimentar también con los tuples.Colocar algunas trazas para que se puedan ver los resultados
- Compilar el programa desde la línea de comandos con tsc
- Ejecutarlo con Node.js

ejercicio0002.ts

- Crear un programa que rellene un array de números y muestre sus contenidos por consola
- Examinar los métodos push y forEach de los arrays e intentar emplear una expresión lambda para mostrar los datos

ejercicio0003.ts

- Crear un programa que rellene un array de números y calcule su media aritmética
- Mostrar el resultado por consola

ejercicio0004.ts (Parte I)

- Crear la clase Animal con el atributo peso, un constructor y un método que aumente el peso en cualquier cantidad
- Opcionalmente, definir un get y un set para el peso
- Instanciar un Animal y modificar su peso inicial
- Mostrar los resultado por consola

ejercicio0004.ts (Parte II)

- Definir una jerarquía de herencia basada en la clase Animal con al menos un descendiente
- Instanciar algunos animales específicos y modificar sus datos
- Mostrar los resultado por consola

ejercicio0005.ts

- Definir una jerarquía de herencia basada en la clase Persona con un descendiente, Empleado
- Definir la interfaz persistente con un único método, llamado guardar
    - interface Persistente{ guardar(): void}
- Simular que una llamada a ese método hace persistente a una Persona, empleando una “base de datos” simulada: const db:Persona[] = [];
- Hacer persistentes algunas instancias y mostrar los resultados

ejercicio0006.ts

- Basándonos en el ejercicio0005, convertir la interfaz en genérica:
    - interface PersistenteGenerica<T>{ guardar(): T}
- Conseguir que el programa funcione como antes del cambio