



La clase de DAW

Tenemos un aula con alumnado con la siguiente información:

- Nombre: String
- Nota: Double (redondeado siempre a dos decimales)
- Curso: String.

Y la siguiente colección de alumnado:

- "Juan", 7.5, "DAM"
- "Pedro", 8.5, "DAM"
- "Ana", 9.5, "DAW"
- "María", 8.5, "DAM"
- "José", 9.5, "DAW"
- "Alicia", 7.5, "DAW"

Resuelve las siguientes consultas usando la API funcional de colecciones o de Secuencias y en el menú número de operaciones y variables (o pasos intermedios):

- Todos los alumnos
- Alumnos de DAW
- Alumnos con nota ≥ 8.5
- Lista de Alumnos de DAW con nota ≥ 8.5 (sobresalientes) y alumnos con otra.
- Alumnos con nota máxima
- Nota media de los alumnos de DAW
- Alumnos que empieza su nombre por A
- Número de alumnos cuyo nombre empieza por J
- Alumnado agrupado por grupo
- Número de alumnos agrupados por grupo
- Alumnado agrupado por nota mayor que 8.5
- Alumnado ordenado por nota descendente
- Buscar alumnos con nota ≥ 8.5
- Buscar alumnos con nota ≥ 8.5 y curso DAW
- Número de alumnos agrupados por nota mayor que 8.5
- Alumnos agrupados por la primera letra de nombre
- Número de alumnos agrupados por la primera letra de nombre
- Alumnado paginado de 2 en dos con desplazamiento igual al tamaño de la página

Recuerda sacar la información en LOCALE español.

El aparcamiento

Tenemos un aparcamiento con los siguientes vehículos a nivel general:

- Vehículo (matricula: String, marca: String, modelo: String, añoMatriculacion: Int) pero sabemos que realmente siempre tendremos:
- Coche que es un vehículo que tiene numPuertas: Int.
- Moto que es un vehículo que tiene cilindrada: Int
- Camión que es un vehículo con pesoMaximo: Int

Programa un repositorio para el manejo de esta información.

Además, tendremos un validador de datos, teniendo en cuenta que la matricula solo puede ser NNNNLLL y nunca vacía. La marca nunca puede ser vacía y tener entre 2 y 20 caracteres, modelo no puede estar vacío y tendrá entre 2 y 50 caracteres y el año de matriculación entre 2000 y 2024.

Por otro lado, tendremos una cache limitada a los 5 últimos valores consultados

Crea un servicio que integre este repositorio, validador y caché. Además, este servicio tendrá su propio sistema de excepciones.

Testea todos los componentes, usando dobles cuando sea necesario.

Realiza las siguientes consultas usando la API funcionales de colecciones y secuencias usando el menor número de operaciones y variables (o pasos intermedios):

- Listados de vehículos
- Listado de coches
- Obtener los vehículos ordenados por matrícula
- Número de vehículos agrupados por clase/tipo
- Listado de vehículos agrupados por tipo/clase
- Listado de vehículos por año de matriculación de maneta descendiente
- Numero de puertas medio que hay en coches.
- Peso máximo de los camiones
- Cilindrada mínima de motos.
- Vehículo más moderno.
- Marca más importante o que más aparece