

Tema: "Los Dioses del Olimpo"

Parte 1: Clase `Deidad`

Cree una clase `Deidad` que represente a una deidad griega con las siguientes características:

Atributos:

- `String nombre`
- `String dominio` (ej: "guerra", "sabiduría", "mar", etc.)
- `int poder` (rango de 0 a 100)
- `int edadEterna` (edad en años como deidad)
- `boolean esInmortal`
- `double nivelRespeto` (calculado, no introducido manualmente)

Métodos requeridos:

- Constructores: uno por defecto y otro parametrizado.
- Getters y setters.
- `void calcularNivelRespeto()`:
Fórmula $\rightarrow \text{nivelRespeto} = (\text{poder} * 1.5) + (\text{edadEterna} / 10.0)$;
- `void mostrarInformacion()`: muestra todos los datos de la deidad por consola.
- `String toString()`: devuelve una descripción breve como
`Atenea (Dominio: Sabiduría, Poder: 92, Respeto: 183.0)`

Métodos adicionales:

1. `void promocionar()`:
Si la deidad tiene `poder ≥ 80` y `edad < 500`, se incrementa su poder en un 10% y se recalcula el respeto.
2. `boolean esAntiguo()`:
Devuelve `true` si la deidad tiene más de 1000 años.
3. `void convertirseEnLeyenda()`:
Si `nivelRespeto > 200`, la deidad se convierte en leyenda: `esInmortal = false` y `poder = 0`.
4. `void actualizarPoder(int modificador)`:
Aplica un cambio al poder (positivo o negativo), asegurando que el valor permanezca entre 0 y 100.
5. `Deidad compararRespeto(Deidad otra)`:
Devuelve la deidad con mayor `nivelRespeto`.
6. `String mostrarTituloHonorifico()`:
Devuelve un título en función del dominio y poder. Por ejemplo:
 - Dominio: guerra, `poder > 90` \rightarrow "El Invencible"
 - Dominio: mar, `poder ≤ 90` \rightarrow "Guardián del Océano"

Parte 2: Clase Olimpo

Esta clase representa una colección de deidades.

Atributo principal:

- `ArrayList<Deidad> deidades`

Métodos obligatorios:

1. `void añadirDeidad(Deidad d)`
2. `void eliminarDeidadPorNombre(String nombre)`
3. `Deidad buscarDeidadMasPoderosa()`
4. `List<Deidad> filtrarPorDominio(String dominio)`
5. `double calcularMediaPoder()`
6. `void importarDesdeCSV(String rutaFichero)`
→ Leer deidades desde un archivo con el formato:
`nombre,dominio,poder,edadEterna,esInmortal`
7. `void exportarACSV(String rutaFichero)`
→ Guarda las deidades actuales en un fichero CSV.

(Opcional): Implementar un método `Map<String, List<Deidad>> agruparPorDominio()`.

Parte 3: Clase Principal

En una clase `OlimpoApp` o similar, se deberá probar el funcionamiento del programa:

- Crear varias deidades y añadirlas al Olimpo.
 - Ejecutar métodos como `promocionar()`, `convertirseEnLeyenda()`, `mostrarTituloHonorifico()`, etc.
 - Mostrar por pantalla las deidades más poderosas, antiguas o respetadas.
 - Importar y exportar deidades desde/hacia un fichero CSV.
 - Se recomienda crear un menú simple por consola para interactuar con el sistema.
-

Consideraciones adicionales

- Se deben validar los datos al crear objetos (`poder` entre 0–100, edad positiva, etc.).
- Se deben controlar las posibles excepciones durante la lectura y escritura de archivos.
- Se valorará el uso de comentarios, buenas prácticas y modularidad del código.

¡Recuerda invocar a los dioses solo cuando el código falle!