



Geonames (<https://www.geonames.org/>) es un servicio web y una base de datos geográfica que proporciona información sobre localizaciones en todo el mundo. En este ejercicio usaremos datos de países obtenidos de este sitio web. Para cada país se tiene la siguiente información:

- **Código ISO:** código ISO del país, formado por dos caracteres, de tipo String, consultable.
- **Nombre:** nombre del país, de tipo String, consultable.
- **Población:** número total de habitantes del país, de tipo Long, consultable.
- **Área:** superficie total del país, medida en kilómetros cuadrados, de tipo Double, consultable.
- **Capital:** ciudad principal o centro administrativo del país, de tipo String, consultable. Su valor será null si el país no tiene capital, como ocurre, por ejemplo, en el caso de la Antártida.
- **Continente:** continente al que pertenece el país, de tipo Continente, consultable. Puede tomar los valores: AFRICA, ANTARCTICA, ASIA, EUROPE, NORTH_AMERICA, OCEANIA y SOUTH_AMERICA.
- **Moneda:** moneda oficial, de tipo String, consultable. Su valor será null si el país no tiene habitantes, como ocurre, por ejemplo, en el caso de la Antártida.
- **Idiomas:** lista de idiomas hablados en el país, de tipo List<Idioma>, consultable. Cada objeto de tipo Idioma contiene información sobre su nombre y el porcentaje de la población que lo habla, como se detalla más adelante. Esta lista puede estar vacía si el país no tiene idiomas, como ocurre, por ejemplo, en el caso de la Antártida.
- **Vecinos:** países vecinos del país, donde cada país vecino viene dado por su código ISO, de tipo Set<String>, consultable. El conjunto estará vacío en el caso de que el país no tenga fronteras con otros países, como es el caso de algunas islas.
- **Densidad de población:** población por kilómetro cuadrado (km²) del país, de tipo Double, consultable.
- **Aislado:** indica si el país no tiene vecinos, de tipo Boolean, consultable.

La información de cada país está almacenada en un archivo CSV que, en cada línea, contiene los siguientes datos: código ISO, nombre, población, área, capital, continente, moneda, idiomas y vecinos. A continuación, se muestran varias líneas de ejemplo de este archivo. Note que puede que haya países para los que no se tienen algunos datos (véase, por ejemplo, la Antártida, que tiene como código ISO, AQ; como nombre, AN; como población, 0; como área, 1.4E7; como capital, null; como continente, AN; como moneda, null; como idiomas, un tabulador ("t"); y como vecinos, otro tabulador ("t").

```
AM; Armenia; 2951776; 29800.0; Yerevan; AS; Dram; Armenian: 1.0; AZ, IR, TR, GE
AO; Angola; 30809762; 1246700.0; Luanda; AF; Kwanza; Portuguese: 1.0; CD, ZM, NA, CG
AQ; AN; 0; 1.4E7; null; AN; null; ;
AT; Austria; 8847037; 83858.0; Vienna; EU; Euro; Hungarian: 0.53, German: 0.26, Slovenian:
0.12, Croatian: 0.08; DE, CH, SI, CZ, SK, IT, LI, HU
```

Cree los paquetes `fp.países` y `fp.países.test` en el proyecto, e implemente los tipos que se indican a continuación.

Ejercicio 1: tipo Idioma (0,25 puntos)

Implemente el tipo Idioma mediante un **record**, a partir de la siguiente descripción:

Propiedades:

- **Nombre:** nombre del idioma, de tipo String, consultable.
- **Porcentaje:** porcentaje de los habitantes del país que hablan el idioma, de tipo Double, consultable.

Constructores:

- C1: recibe un parámetro por cada propiedad básica del tipo, en el mismo orden en el que están definidas.

Criterio de igualdad: dos idiomas serán considerados iguales si tienen el mismo nombre.

Representación como cadena: una cadena con todas las propiedades básicas del tipo.

Restricciones:

- R1: el valor del porcentaje debe estar comprendido en el intervalo [0.0, 1.0].

Ejercicio 2: tipo Pais (1,5 puntos)

Implemente el tipo **Pais** mediante una **clase**, a partir de la descripción anterior y la información que se proporciona a continuación. Para ello, deberá crear también un tipo enumerado llamado **Continente**.

Constructores:

- C1: recibe un parámetro por cada propiedad básica del tipo, en el mismo orden en el que están definidas.

Restricciones:

- R1: el valor de la población debe ser mayor o igual a 0.
- R2: el área debe ser mayor o igual a 0.
- R3: si la población es mayor a 0, entonces el país debe tener una capital y una moneda.

Criterio de igualdad: dos países son iguales si tienen el mismo código ISO.

Representación como cadena: una cadena con todas las propiedades básicas del tipo.

Otras operaciones:

- *Boolean sonTodosVecinos(Set<Pais> paises)*: dado un conjunto de países, devuelve true si todos los países del conjunto dado como parámetro son vecinos del país, y false en caso contrario.

Ejercicio 3: Factoría (1,25 puntos)

Añada el siguiente método a la clase **FactoriaPaises**, que se proporciona parcialmente implementada:

- *Pais parseaPais(String lineaCSV)*: devuelve un objeto de tipo Pais a partir de una cadena de caracteres. La cadena debe tener el mismo formato que una de las líneas del archivo CSV. Note que los elementos de cada línea están separados por puntos y comas (;) y comas (,). Cree los métodos auxiliares que considere necesarios.

Ejercicio 4: tipo PaisesDelMundo (7 puntos)

Implemente el tipo **PaisesDelMundo** a partir de la siguiente descripción:

Propiedades:

- **Paises**: conjunto ordenado con los países del mundo, de tipo SortedSet<Pais>, consultable. El conjunto estará ordenado según el nombre de los países.
- **Fecha y hora de actualización**: fecha y hora de creación del conjunto de países, de tipo LocalDateTime, consultable.
- **Número de países**: número de países del mundo, de tipo Integer, consultable.

Constructores:

- C1: recibe un parámetro de tipo LocalDateTime y un Stream<Pais> para inicializar los atributos.
- C2: recibe un parámetro de tipo List<Pais> para los países, y crea un objeto de tipo PaisesDelMundo cuya fecha y hora de actualización son la fecha y hora de creación del objeto.

Criterio de igualdad: dos objetos son iguales si lo son sus conjuntos de países y sus fechas y horas de actualización.

Representación como cadena: una cadena con todas las propiedades básicas del tipo.

Añada además los siguientes métodos al tipo `PaísesDelMundo`. Debe implementar todos los métodos usando **streams**, a menos que se indique lo contrario:

1. `Double porcentajeAreaDeContinente(Continente cont)`: calcula el porcentaje de la superficie del continente dado como parámetro con respecto a la superficie total del mundo. (1 punto)
2. `List<String> nombresDeVecinos(String país)`: devuelve una lista con los nombres de los vecinos del país recibido como parámetro. La lista debe estar ordenada alfabéticamente. Tenga en cuenta que la lista debe contener los nombres de los vecinos (no los códigos iso). (1,5 puntos)
3. `SortedMap<String, List<String>> paisesPorIdiomas()`: devuelve un Map ordenado que contiene los nombres de los países que hablan el mismo idioma. Las claves deben estar ordenadas según su orden natural. Además, las listas con los nombres de los países también deben estar ordenadas según su orden alfabético. **Este ejercicio debe resolverse utilizando bucles (no con streams)**. (1,5 puntos)
4. `SortedMap<String, String> paisMayorPoblacionPorMoneda()`: devuelve un Map ordenado que contiene el nombre del país con mayor población por nombre de moneda. Si hay más de un país por moneda con el mismo valor de población, se tomará el país con densidad de población. (1,5 puntos)
5. `List<String> vecinosMasFrecuentes(Integer n)`: devuelve una lista con los nombres de los países que son los n vecinos más frecuentes. Por ejemplo, si n fuera 1, devolvería una lista con el país que tiene más fronteras con otros países. La lista deberá estar ordenada según el orden alfabético de los países. (1,5 puntos)

Escriba un test para el tipo `PaísesDelMundo`. En el test, deberá leer los datos del CSV y probar todos los métodos del ejercicio 4. Los resultados esperados para los valores indicados en las pruebas son:

EJERCICIO 4.1 (usando América del Sur como parámetro):

```
Test porcentajeAreaDeContinente (SOUTH AMERICA)
-----
0.1187801235604826
```

EJERCICIO 4.2 (usando España como parámetro):

```
Test nombresDeVecinos (Spain)
-----
[Andorra, France, Gibraltar, Morocco, Portugal]
```

EJERCICIO 4.3 (solamente se muestran los tres primeros y últimos elementos del diccionario):

```
Test paisesPorIdiomas
-----
{Adyghe=[Russia], Afar=[Djibouti, Eritrea], Afrikaans=[Namibia, South AF], ...,
Zarma=[Niger], Zhuang=[China], Zulu=[Lesotho, South AF]}
```

EJERCICIO 4.4: (solamente se muestran los tres primeros y últimos elementos del diccionario):

```
Test paisMayorPoblacionPorMoneda
-----
{Afghani=Afghanistan, Ariary=Madagascar, Baht=Thailand, ..., Yen=Japan, Yuan Renminbi=China,
Zloty=Poland}
```

EJERCICIO 4.5 (usando 5 como parámetro para limitar el tamaño de la lista obtenida):

```
Test vecinosMasFrecuentes
-----
[Brazil, China, Democratic Republic of the Congo, Germany, Russia]
```