

Ejercicio: Avistamientos

Autor: Mariano González. **Revisores:** Fermín Cruz. **Última modificación:** 5/3/2020

En este ejercicio vamos a trabajar con un conjunto de datos con información sobre avistamientos de objetos voladores no identificados (OVNIs) en los Estados Unidos. El objetivo del ejercicio es leer estos datos y realizar distintas operaciones con ellos.

Los datos se encuentran almacenados en un fichero en formato CSV codificado en UTF-8. Cada registro del fichero ocupa una línea y contiene los datos correspondientes a un avistamiento: fecha en la que se produjo del avistamiento, ciudad donde se produjo, duración en segundos, forma observada del avistamiento y latitud y longitud del lugar donde se produjo.

Estas son las primeras líneas del fichero. La primera línea es una cabecera que contiene los nombres de los campos del registro:

```
date;city;duration;shape;location
04/07/2011;muncie;240;OTRA;(40.1933333,-85.3863889)
07/04/2005;deming (somewhere near);1200;OTRA;(32.2686111,-107.7580556)
12/03/2010;erie;300;OTRA;(42.1291667,-80.0852778)
04/07/2013;seattle;600;OTRA;(47.6063889,-122.3308333)
09/09/2003;clearwater;120;TRIANGULAR;(27.9655556,-82.8002778)
```

Los tipos que forman el modelo de datos son los siguientes:

- **Hemisferio:** tipo enumerado con los dos valores posibles de un hemisferio terrestre.
- **Coordenadas:** ubicación geográfica de un avistamiento.
- **Avistamiento:** tipo que representa un avistamiento OVNI.
- **Avistamientos:** tipo contenedor que representa una colección de avistamientos.
- **FactoriaAvistamientos:** tipo factoría para cargar los avistamientos del fichero.

Tipo Coordenadas

Propiedades:

- **latitud**, de tipo Double. Consultable y modificable. Latitud de las coordenadas.
- **longitud**, de tipo Double. Consultable y modificable. Longitud de las coordenadas.
- **hemisferio**, del tipo enumerado Hemisferio, que puede tomar los valores NORTE y SUR. Consultable. Hemisferio al que pertenecen las coordenadas.

Constructores:

- Un constructor que recibe un parámetro por cada propiedad básica del tipo, en el mismo orden en el que están definidas.
- Un constructor sin parámetros que crea un objeto con latitud 0º y longitud 0º.
- Un constructor a partir de String. Ejemplo de formato de la cadena: "{-1.5, 0.22}"

Criterio de igualdad: dos coordenadas son iguales si su latitud y su longitud son iguales.

Criterio de ordenación: por latitud, y a igualdad de esta por longitud.

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

Restricciones:

- La latitud debe estar comprendida entre -90° y $+90^{\circ}$.
- La longitud debe estar comprendida entre -180° y $+180^{\circ}$.

Otras operaciones:

- *Double getDistancia(Coordenadas c)*: calcula la distancia entre dos coordenadas.

Tipo Avistamiento

Propiedades:

- **fecha**, de tipo LocalDate. Consultable. Fecha en la que se produce el avistamiento.
- **lugar**, de tipo String. Consultable y modificable. Ciudad donde se produce el avistamiento.
- **duracion**, de tipo Integer. Consultable y modificable. Duración del avistamiento en segundos.
- **forma**, del tipo enumerado Forma, que puede tomar los valores CIRCULAR, TRIANGULAR y OTRA. Consultable. Forma observada del avistamiento.
- **ubicacion**, de tipo Coordenadas. Consultable. Coordenadas geográficas en las que se produce el avistamiento.
- **año**, de tipo Integer. Consultable. Año del avistamiento, obtenido a partir de la fecha.

Constructores:

- Un constructor que recibe un parámetro por cada propiedad básica del tipo, en el mismo orden en el que están definidas.
- Un constructor que recibe un parámetro por cada propiedad básica del tipo salvo la fecha, para la que se toma la fecha del día actual.
- Un constructor a partir de String. Ejemplo de formato de la cadena: "21/01/2019; Sevilla; 30; CIRCULAR; (37.38, -5.97)"

Criterio de igualdad: dos avistamientos son iguales si su fecha y su lugar son iguales.

Criterio de ordenación: por fecha, y a igualdad de esta por lugar.

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

Restricciones:

- La duración del avistamiento debe ser positiva.
- La fecha debe ser igual o anterior a la fecha del día en que se crea el avistamiento.

Otras operaciones:

- *Double getDistancia(Avistamiento av)*: calcula la distancia entre dos avistamientos.

Tipo Avistamientos

Propiedades:

- **avistamientos**, de tipo `Set<Avistamiento>`. Consultable. Conjunto de avistamientos.

Constructores:

- Un constructor sin parámetros.
- Un constructor a partir de un `Stream<Avistamiento>`.

Criterio de igualdad: dos avistamientos son iguales si sus conjuntos de avistamientos son iguales.

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

Otras operaciones:

- `void añadirAvistamiento(Avistamiento av)`: añade un avistamiento al conjunto.

Tratamientos secuenciales:

- `Integer getNumeroAvistamientosFecha(LocalDate f)`: calcula el número total de avistamientos producidos en una fecha dada.
- `Set<Avistamiento> getAvistamientosCercanosUbicacion(Coordenadas c, Double d)`: obtiene los avistamientos cercanos a una ubicación dada.
- `Boolean existeAvistamientoLugarAño(String l, Integer a)`: indica si existe algún avistamiento en un lugar dado en un año dado.
- `Avistamiento getAvistamientoMayorDuracion()`: obtiene el avistamiento de mayor duración.
- `Map<LocalDate, Set<Avistamiento>> getAvistamientosPorFecha()`: crea un diccionario que relaciona las fechas con los avistamientos producidos en esa fecha.
- `Map<Integer, Long> getNumeroAvistamientosPorAño()`: crea un diccionario que relaciona los años con el número de avistamientos producidos en ese año.

Tipo FactoriaAvistamientos

Operaciones:

- `Avistamientos leerAvistamientos(String nombreFichero)`: lee un fichero de avistamientos y construye un objeto de tipo `Avistamientos`.
- `Avistamiento parsearAvistamiento(String lineaCSV)`: crea un objeto de tipo `Avistamiento` a partir de una cadena de caracteres. La cadena de caracteres debe tener el mismo formato que las líneas del fichero CSV.