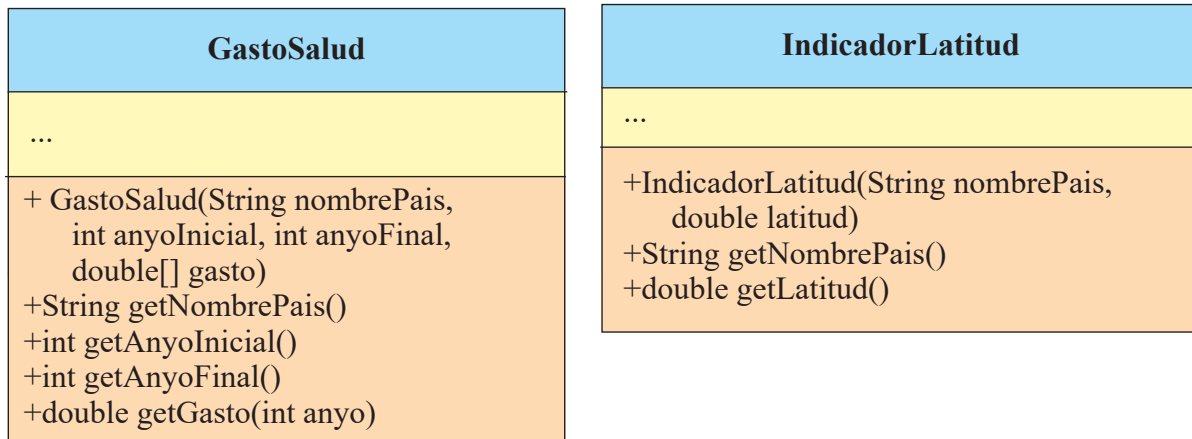


Examen de Prácticas de Introducción al Software (Ingeniería Informática)

Septiembre 2018

Se desea escribir parte de una aplicación para analizar datos de gastos en salud de diferentes países. Se dispone ya de dos clases ya realizadas cuyos diagramas de clase se muestran aquí. La clase GastoSalud contiene los datos de gastos en salud per cápita de un país en diferentes años e IndicadorLatitud contiene datos de la latitud de un país:



Los métodos de la clase GastoSalud hacen lo siguiente:

- *Constructor*: Construye el objeto con los datos que se pasan como parámetros
- `getNombrePais()`: Retorna el nombre del país
- `getAnyoInicial()`: retorna el año inicial para el que hay datos de gasto para este país
- `getAnyoFinal()`: retorna el año final para el que hay datos de gasto para este país
- `getGasto()`: Retorna el gasto en salud en el año indicado, en US\$, o Double.NaN si no hay datos para el año indicado

Y los de IndicadorLatitud:

- *Constructor*: Construye el objeto con los datos que se pasan como parámetros, estando la latitud en grados.
- `getNombrePais()`: observador del atributo nombrePais
- `getLatitud()`: observador del atributo latitud, en grados

Se pide escribir parte de la clase DatosSaludPaises. Sus atributos son una lista de objetos de la clase GastoSalud y una lista de objetos de la clase IndicadorLatitud. Los métodos aparecen en el diagrama de clase y deben hacer :

- *Constructor*: crea las listas de gastos en salud y latitudes y posteriormente las lee de sendos ficheros mediante las operaciones `leeGastosSalud()` y `leeLatitudes()`.
- `buscaLatitud()`: Busca un dato de latitud por el nombre del país. Retorna el dato si se encuentra, o null si no se encuentra. Este método se da ya hecho.
- `gastoMedioFranjaLatitud()`: Calcula y retorna el gasto medio en salud, en US\$ considerando solo los países que tienen dato de latitud y de gasto en salud, y cuyo valor absoluto de latitud está en la franja entre `latMin` (incluida) y `latMax` (excluida). Si no hay ningún país que cumple los criterios se retornará cero.

- `buscaGastoMayor()`: Busca y retorna el nombre del primer país cuyo gasto en salud en el año indicado excede el parámetro `gastoMinimo`. Los países sin datos para ese año no se tienen en cuenta. Si no encuentra ningún país que cumpla los criterios se retorna `null`.
- `leeGastosSalud()`: Lee datos de gastos de salud del fichero cuyo nombre se indica almacenándolos en la lista de gastos. Si se producen errores en la lectura se detiene el programa con un mensaje de error. Este método se ofrece ya hecho.
- `leeLatitudes()`: Lee datos de latitud del fichero cuyo nombre se indica almacenándolos en la lista de latitudes. Si se producen errores en la lectura se detiene el programa con un mensaje de error. Este método se ofrece ya hecho.

DatosSaludPaises
<ul style="list-style-type: none"> - <code>ArrayList<GastoSalud> listaGasto</code> - <code>ArrayList<IndicadorLatitud> listaLatitud</code>
<ul style="list-style-type: none"> +<code>DatosSaludPaises</code> (<ul style="list-style-type: none"> <code>String nombreFicheroGastos,</code> <code>String nombreFicheroLatitudes)</code> -<code>IndicadorLatitud buscaLatitud</code>(<ul style="list-style-type: none"> <code>String nombrePais)</code> +<code>double gastoMedioFranjaLatitud</code>(<ul style="list-style-type: none"> <code>double latMin, double latMax)</code> +<code>String buscaGastoMayor</code>(<ul style="list-style-type: none"> <code>int anyo, double gastoMinimo)</code> +<code>void leeGastosSalud</code>(<code>String nombreFichero)</code> +<code>void leeLatitudes</code>(<code>String nombreFichero)</code>

Escribir además en una clase aparte un programa de prueba con un main que haga:

- a) Crear un objeto de la clase `DatosSaludPorPais` leyendo de los ficheros `health-spending.csv` e `indicator-latitude.csv`.
- b) Muestra en pantalla una gráfica de los gastos medios en salud por franjas de latitud: `[0,10)`, `[10,20)`, `[20,30)`, `[30,40)`, `[40,50)`, `[50,60)` y `[60,70)` grados. Abajo se muestra un ejemplo con los resultados que se deben obtener.
- c) Muestra en pantalla el nombre del primer país cuyo gasto en salud en el año 2009 excede 6000 US\$, usando el método `buscaGastoMayor()`. Si el valor retornado por ese método es `null`, se pondrá un mensaje de error en pantalla. El resultado para los datos proporcionados debe ser: Denmark.
- d) Igual que el paso c) pero con 9000 US\$. El resultado para los datos proporcionados debe ser el mensaje de error.

Valoración (sobre 10):

- 1) Encabezamiento de la clase, atributos y constructor: 1 punto
- 2) `gastoMedioFranjaLatitud`: 2 puntos
- 3) `buscaGastoMayor`: 2 puntos
- 4) main: 2 puntos
- 5) captura de pantalla con la gráfica: 2 puntos
- 6) captura de pantalla con los resultados escritos en el terminal por el main: 1 punto

Ejemplo de la gráfica obtenida con los datos proporcionados

