



FUNDAMENTOS DE PROGRAMACIÓN

SEGUNDO PARCIAL. 10 de junio de 2019. Sesión 1

EJERCICIO 1 – TIPO BASE (1 pto)

Se quiere fiscalizar la oferta de empleo público del Servicio Andaluz de Salud, accediendo a los datos de las ofertas de empleo publicados en el portal de datos abiertos de la Junta de Andalucía, de donde se ha obtenido un subconjunto de los datos publicados, que se han guardado en el fichero `oep_sas_2018_utf8.csv`, situado en la carpeta `data` del proyecto que se le proporciona. El fichero tiene la siguiente estructura:

```
ESPECIALIDAD;PLAZAS;RESERVA DISCAPACIDAD;FECHA PUBLICACIÓN BOJA;ACCESO_LIBRE/PROMOCION_INTERNA  
ADMINISTRATIVO/A;135;15;23/11/2018;ACCESO_LIBRE  
AUXILIAR ADMINISTRATIVO/A;114;13;23/11/2018;ACCESO_LIBRE  
FEA ENDOCRINOLOGÍA Y NUTRICIÓN;1;0;23/11/2018;PROMOCION_INTERNA  
FEA FARMACIA HOSPITALARIA;1;0;23/11/2018;PROMOCION_INTERNA  
FEA HEMATOLOGÍA Y HEMOTERAPIA;2;0;23/11/2018;PROMOCION_INTERNA
```

Implemente la clase **OfertaEmpleo** con las siguientes operaciones (puede ayudarse del diagrama de clases de la Figura 1).

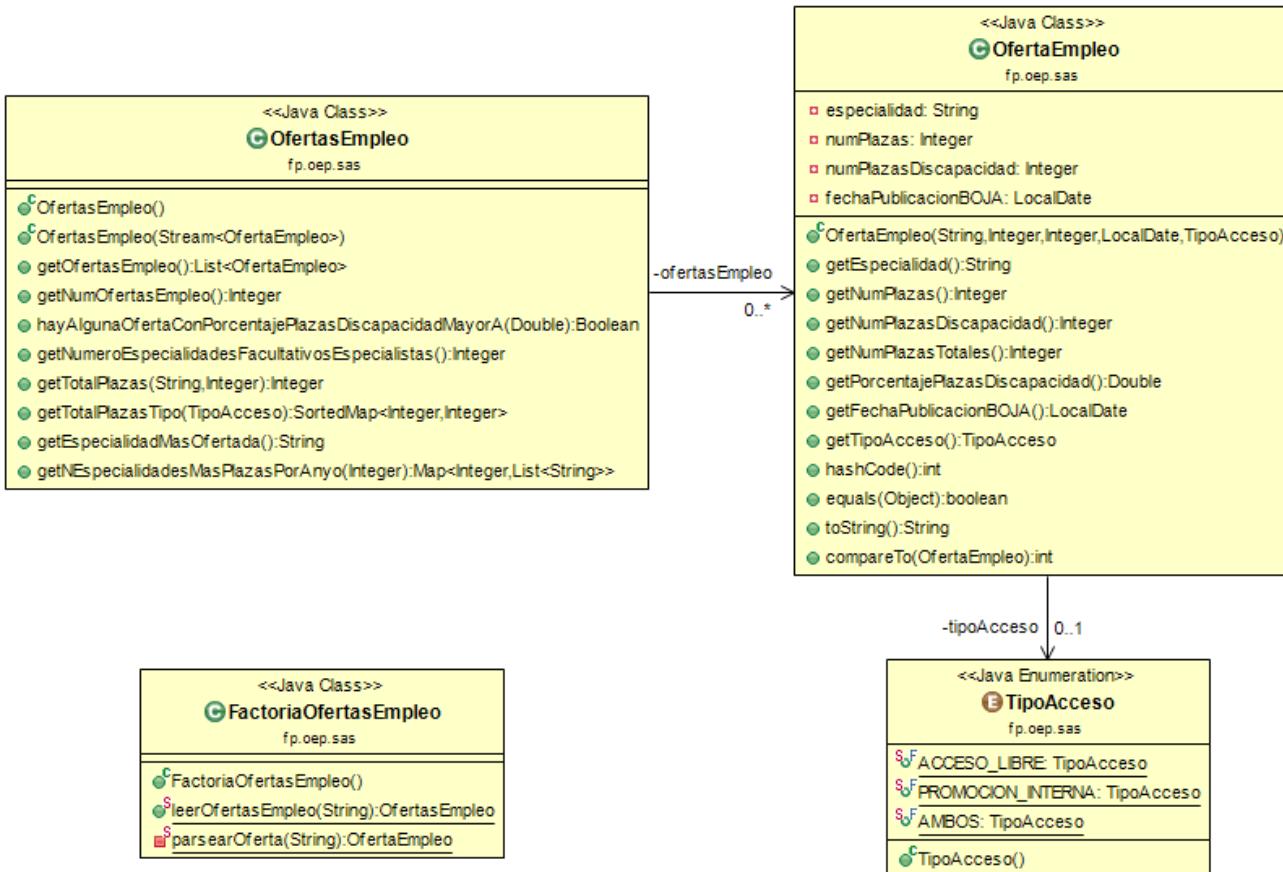


Figura 1. Diagrama de clases

Propiedades:

- `especialidad`, de tipo `String`. Consultable. Representa la especialidad para la que se realiza la oferta.
- `número de plazas`, de tipo `Integer`. Consultable. Número de plazas ordinarias ofertadas para la especialidad.

- número de plazas discapacidad, de tipo Integer. Consultable. Número de plazas reservadas para personas con discapacidad.
- número de plazas totales, de tipo Integer. Consultable. Número total de plazas ofertadas. Se obtiene a partir del número de plazas y el número de plazas con discapacidad.
- porcentaje Plazas Discapacidad, de tipo Double. Consultable. Representa la relación entre el número de plazas reservadas para discapacidad y el número total de plazas ofertadas.
- fecha de publicación BOJA, de tipo LocalDate. Recoge la fecha en la que salió publicada en BOJA la oferta de empleo.
- tipo de acceso, de tipo TipoAcceso. Puede tomar los valores: ACCESO_LIBRE, PROMOCION_INTERNA o AMBOS.

Constructores: un solo constructor que tiene como parámetros las propiedades básicas del tipo, y en el mismo orden en el que se han definido en el apartado anterior.

Criterio de igualdad: dos ofertas son iguales si tienen la misma fecha de publicación en BOJA, la misma especialidad y el mismo tipo de acceso.

Criterio de ordenación: por fecha de publicación en BOJA, luego por especialidad, luego por tipo de acceso, y finalmente por número de plazas totales.

Representación como cadena: el valor de todas las propiedades, incluidas las derivadas, separados por coma.

Restricciones:

- El número de plazas no puede ser negativo.
- El número de plazas reservadas para discapacitados no puede ser negativo.
- El año de publicación debe ser posterior o igual a 1990 y la fecha de publicación debe ser anterior o igual a la fecha de hoy.

EJERCICIO 2 – TIPO CONTENEDOR (8 puntos, 7,5 de los cuales son para las operaciones de tratamientos secuenciales)

Implemente el tipo contenedor **OfertasEmpleo**, con la siguiente descripción:

Propiedades:

- ofertas de empleo, de tipo List<OfertaEmpleo>. Consultable. Lista con ofertas de empleo.
- número de ofertas de empleo, de tipo Integer. Consultable. Número de ofertas de empleo registradas.

Constructores: un solo constructor que tiene como parámetro un Stream<OfertaEmpleo> y crea un objeto de tipo OfertasEmpleo a partir del Stream.

Criterio de igualdad: dos objetos de tipo OfertasEmpleo son iguales si sus listas de ofertas son iguales.

Representación como cadena: la representación como cadena de la lista de ofertas de empleo.

Otras operaciones:

- a) **Boolean hayAlgunaOfertaConPorcentajePlazasDiscapacidadMayorA(Double umbral)**. Devuelve cierto si hay alguna oferta de empleo con un porcentaje de plazas reservadas para discapacidad superior o igual al umbral dado como parámetro. Se eleva IllegalArgumentException si el umbral no está en el intervalo [0,100]. **(1 Pto)**
- b) **Integer getNumeroEspecialidadesFacultativosEspecialistas()**. Devuelve el número de especialidades de Facultativo/a Especialista de Área distintas para las que se ofertan plazas de cualquier tipo. Las especialidades de facultativos especialistas son aquellas cuyo nombre comienza por FEA, por ejemplo, FEA ALERGOLOGÍA. **(1 Pto)**

- c) **Integer getTotalPlazas(String especialidad, Integer anyo).** Devuelve el número de plazas totales de la especialidad dada como parámetro ofertadas en el año dado como parámetro. Si no se ofertó ninguna plaza, el método debe devolver 0. Se considera que el año en el que se ofertan las plazas es el año de su publicación en BOJA. **(1 Pto)**
- d) **SortedMap<Integer, Integer> getTotalPlazasTipo(TipoAcceso tipo).** Devuelve un SortedMap que asocia un año y el total de plazas del tipo que se pasa como parámetro que se ofertaron ese año. **(1 Pto)**
- e) **String getEspecialidadMasOfertada().** Devuelve el nombre de la especialidad para la que se han ofertado más plazas. Si no se puede calcular se elevará la excepción NoSuchElementException. **(1,5 Ptos)**
- f) **Map<Integer, List<String>> getNEspecialidadesMasPlazasPorAnyo(Integer n).** Dado un número n, proporcionado como parámetro, devuelve un Map en el que las claves son los años de publicación de la oferta y los valores son las especialidades de las N ofertas de empleo en las que se ofertaron un mayor número de plazas totales. Use dos métodos auxiliares: uno que devuelva un Map auxiliar para organizar las ofertas por año; y otro que tome una lista de OfertaEmpleo y un número n, y devuelva otra lista con los nombres de las n especialidades con más plazas, ordenadas de mayor a menor número de plazas. **(2 Ptos)**

EJERCICIO 3 – FACTORÍA (1 pto)

Implemente el tipo **FactoriaOfertasEmpleo** para poder cargar la información del fichero de datos proporcionado y añádale el siguiente método:

- a) **OfertaEmpleo parsearOferta(String lineaCSV).** Crea un objeto de tipo OfertaEmpleo a partir de una cadena de caracteres. La cadena de caracteres debe tener el mismo formato que una de las líneas del fichero especificado anteriormente. Si la línea no tiene el formato esperado, se elevará IllegalArgumentException. Use el formato d/M/yyyy para las fechas.