



En el contexto de los programas de mejora docente, una universidad ha creado una base de datos para registrar la información de los proyectos de innovación educativa **concedidos (con o sin financiación)** desarrollados por su personal docente. El objetivo es facilitar su análisis y evaluación posterior.

Cada proyecto está liderado por un responsable, puede tener colaboradores, y si está financiado, debe tener una entidad financiadora obligatoriamente. A continuación, se recoge la información disponible para cada proyecto:

- **Título:** título del proyecto.
- **Colaboradores:** lista de profesores en el proyecto (el responsable en la primera posición). Por cada colaborador, se facilita su nombre y el departamento al que pertenece.
- **Fecha inicio:** fecha en la que comienza el proyecto.
- **Fecha fin:** fecha en la que termina el proyecto.
- **Ámbitos:** conjunto ordenado de los ámbitos de innovación educativa (por ejemplo: “metodologías activas”, “sostenibilidad”, etc.).
- **Presupuesto:** cantidad (en euros) financiada si ha sido financiado por alguna entidad.
- **Entidad financiadora:** entidad que ha aprobado el presupuesto (solo si ha sido financiado).

Dos líneas ejemplo del archivo csv en el que están registrados los datos son las siguientes:

Aprendizaje basado en proyectos en Ingeniería,[Laura Sánchez - Química Inorgánica;Carlos Díaz- Química Inorgánica],2026-01-15,2026-07-15,[metodologías activas; sostenibilidad],3200.0,Vicerrectorado de Profesorado Flip teaching en Java, [Jorge García - Lenguajes y Sistemas Informáticos;David Benavides- Lenguajes y Sistemas Informáticos; José A. Galindo- Lenguajes y Sistemas Informáticos],2025-09-10,2026-07-15,[metodologías activas; programación]

La primera línea presenta el proyecto titulado "Aprendizaje basado en proyectos en Ingeniería", cuyos profesores implicados son Laura Sánchez (Dpto. Química Inorgánica) y Carlos Díaz (Dpto. Química Inorgánica). Laura Sánchez es la responsable. El proyecto abarca el periodo del 15 de enero al 15 de julio de 2026 y tiene como ámbitos las metodologías activas y la sostenibilidad. Además, el Vicerrectorado de Profesorado lo financia con 3200€.

La segunda línea muestra otro proyecto con características similares al anterior, pero sin financiación. La información que se presenta sigue la misma estructura que la primera línea, aunque al final, después de los ámbitos no se incluye presupuesto ni entidad financiadora.

Para dar soporte al análisis de datos, **implemente en Java los tipos** que se especifican en los siguientes ejercicios:



Ejercicio 1: Tipo Colaborador (0,5 puntos)

Implemente en el paquete `fp.innovacion` el tipo **Colaborador** mediante un **record**, de acuerdo con la siguiente descripción:

Propiedades:

- **nombre**, de tipo `String`. Representa el nombre del colaborador. Consultable.
- **departamento**, de tipo `String`, Representa el departamento al que pertenece dentro de la universidad. Consultable.

Constructores:

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

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

Restricciones:

- R1: El nombre y el departamento no pueden ser nulos ni estar vacíos.

Criterio de igualdad: dos responsables son iguales si tienen el mismo nombre y el mismo departamento.

Criterio de ordenación: los colaboradores se ordenan alfabéticamente por nombre y si hay empate, por departamento.

Ejercicio 2: Tipo Proyecto (1,5 puntos)

Implemente en el paquete `fp.innovacion` el tipo **Proyecto** utilizando **una clase**, de acuerdo con la siguiente descripción. **En la implementación de esta clase no se pueden utilizar streams.**

Propiedades:

- **título**, de tipo `String`, consultable.
- **colaboradores**, de tipo `List<Colaborador>`, consultable.
- **fechalnicio**, de tipo `LocalDate`, consultable y modificable.
- **fechaFin**, de tipo `LocalDate`, consultable.
- **ambitos**, de tipo `SortedSet<String>`, consultable.
- **responsable**, de tipo `Colaborador`, consultable.
- **duracion**, de tipo `Long`, número de días entre fecha de inicio y fin, consultable.
- **estado**, de tipo `Estado` (`CONCEDIDO`, `EN_CURSO`, `FINALIZADO`) en función de si aún no ha comenzado, si lo ha hecho y actualmente está en vigor o si ya ha finalizado, respectivamente, consultable
- **numeroColaboradores**, de tipo `Integer`, consultable.

Constructores:

- C1: recibe un parámetro por cada propiedad básica del tipo.

Restricciones:

- R1: La fecha de fin debe ser posterior a la de inicio.
- R2: No puede haber colaboradores duplicados (mismo nombre y departamento).
- R3: El conjunto de ámbitos no puede ser nulo ni vacío.

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

Criterio de igualdad: Dos proyectos son iguales si lo son sus responsables y sus títulos.

Criterio de ordenación: Los proyectos se ordenan por título y en caso de empate, por responsable.

Otras operaciones:



- `esColaborador(String nombre)`: devuelve true si el nombre está en la lista de colaboradores o false, en caso contrario.

Ejercicio 3: Tipo ProyectoFinanciado (1 punto)

Implemente en el paquete `fp.innovacion` el tipo **ProyectoFinanciado** utilizando **una clase** que extienda a **Proyecto**.

Propiedades:

- Todas las propiedades de Proyecto con la misma descripción.
- **presupuesto**, de tipo Double, consultable y modificable.
- **entidad financiadora**, de tipo String, consultable y modificable.

Constructores:

- C1: recibe un parámetro por cada propiedad básica del tipo.

Restricciones:

- R1: El presupuesto tiene que ser mayor que cero.
- R2: La entidad financiadora no puede ser vacía.
- R3: La duración del proyecto debe ser menor de 2 años si el presupuesto es inferior a 10000 €

Representación como cadena: la misma que la de Proyecto, seguida del presupuesto y la entidad financiadora separadas por comas.

Criterio de igualdad: el mismo que el de Proyecto.

Criterio de ordenación: el mismo que el de Proyecto.

Ejercicio 4: Factoría (1,5 puntos)

En la clase **FactoriaProyectos**, que se le da parcialmente implementada en el paquete `fp.innovacion`, implemente el método:

- *parseaProyecto*: crea un objeto Proyecto o ProyectoFinanciado a partir de una línea del archivo CSV (como las mostradas al principio del enunciado). Tenga en cuenta que si la línea tiene un presupuesto, el artículo debe ser de tipo ProyectoFinanciado, mientras que si la línea no tiene precio, debe ser de tipo Proyecto.

Resultado del test:

Se han leído 60 proyectos

Primero:

ProyectoFinanciado [Proyecto=Proyecto [titulo=Aprendizaje basado en retos en Psicología, colaboradores=[Colaborador[nombre=Álvaro Ramírez, departamento=Psicología], Colaborador[nombre=Diego Torres, departamento=Psicología], Colaborador[nombre=Miguel Ruiz, departamento=Psicología], Colaborador[nombre=Sara Sánchez, departamento=Pedagogía]], fechaInicio=2026-03-22, fechaFin=2026-12-24, ambitos=[aprendizaje autónomo, programación], entidadFinanciadora=Ministerio de Educación, presupuesto=3609.62]

Último:

Proyecto [titulo=Gamificación en Matemáticas, colaboradores=[Colaborador[nombre=Carmen Suárez, departamento=Estadística], Colaborador[nombre=Javier Navarro, departamento=Matemáticas], Colaborador[nombre=Sofía Moreno, departamento=Estadística]], fechaInicio=2026-09-26, fechaFin=2027-11-04, ambitos=[TIC, experimentación, programación]]

**Ejercicio 5: Tipo contenedor ProyectoManager (5,5 puntos)**

Implemente en el paquete `fp.innovacion` el tipo **ProyectoManager** según la siguiente descripción:

Propiedades:

- **proyectos:** lista de proyectos a analizar, de tipo `List<Proyecto>`. Consultable.

Constructores:

- **C1:** recibe un parámetro por cada propiedad básica del tipo.

Representación como cadena: la lista de proyectos.

Criterio de igualdad: dos objetos de tipo **ProyectoManager** son iguales si lo son sus proyectos.

Añada a la clase **ProyectoManager** las siguientes operaciones que deben ser implementadas **usando streams**.

1. *responsablesPorAmbito:* *dado un ámbito, devuelve una lista con los nombres de los responsables de los proyectos que contienen dicho ámbito. (0,5 puntos)*

Resultado del test:

```
=== Responsables por ámbito 'TIC' ===
Responsables (21): [Sara Álvarez, Paula Díaz, Raquel Domínguez, Cristina Morales,
Javier Muñoz, Hugo Ramírez, Manuel Rubio, Patricia Castro, José Torres, Isabel
García, Francisco Álvarez, Patricia Gutiérrez, Miguel Torres, José López, Pablo
García, Fernando Ortega, Laura Domínguez, Manuel Hernández, Manuel Moreno, Antonio
Muñoz, Carmen Suárez]
```

2. *presupuestoTotalPorDepartamento:* devuelve un `SortedMap<String, Double>` (orden natural) que asocia a cada departamento del responsable, el presupuesto total de sus proyectos financiados. (0,75 puntos)

Resultado del test:

```
=== Presupuesto total por departamento ===
Administración de Empresas -> 7189.389999999999
Biología -> 4640.51
Ciencias Ambientales -> 18886.21
Ciencias Sociales -> 8258.76
Derecho -> 40135.69
Economía -> 4223.33
Electrónica -> 4779.57
Historia -> 4976.44
Ingeniería del Software -> 2044.95
Lenguajes y Sistemas Informáticos -> 44237.97
Matemáticas -> 26700.6
Pedagogía -> 16329.33
Psicología -> 11978.8
Química Inorgánica -> 16826.89
Química Orgánica -> 14421.81
```



3. *titulosProyectosLargos*¹: devuelve un SortedSet<String> con los títulos de los proyectos ordenados por duración (de mayor a menor) que se prolonguen por encima de un número n de días dado como parámetro (1,75 ptos)

Resultado del test:

```
=== Proyectos con duración > 300 días ===
Aprendizaje-servicio en Biología
Gamificación en Matemáticas
Gamificación en Ingeniería Informática
Realidad aumentada en Ingeniería Informática
Mentoría entre pares en Lenguas Extranjeras
Diseño de materiales abiertos en Ingeniería Informática
Gamificación en Economía
Uso de chatbots en Derecho
Mentoría entre pares en Ingeniería Informática
Gamificación en Biología
Simulación interactiva en Psicología
Diseño de materiales abiertos en Biología
Simulación interactiva en Química
```

4. *proyectoMasColaborativo*: devuelve el título del proyecto con más departamentos colaboradores. En caso de empate, devuelva el que comience antes o null si no hay ningún título que mostrar. (1,5 ptos)

Resultado del test:

```
=== Proyecto más colaborativo ===
Aprendizaje-servicio en Economía
```

Implemente también una clase **TestProyectoManager** en el paquete `fp.proyectos.test`. En la clase se leerán los datos del fichero y se probarán todos los tratamientos secuenciales y la factoría. Defina un método de test con los parámetros adecuados por cada tratamiento secuencial a probar. No se obtendrá la puntuación máxima del ejercicio si no se realiza el test y este se ejecuta correctamente, mostrando los resultados adecuados.

¹ Si hubiera varios proyectos con el mismo título, guarde el de mayor duración.