

Para la realización del examen se entrega un proyecto Java de Netbeans, en el que encontraréis todo lo necesario.

Los recursos necesarios para la realización de cada ejercicio se encuentran en “**Examen DAW\data\ejercicio\_n\**” (donde n es el número del ejercicio)

El examen consta de 4 ejercicios, que deben realizarse íntegramente en lenguaje Java y con las herramientas explicadas en clase.

1. **(1,5 puntos)** Emulando el comando dir de windows o ls de linux, crea una clase en Java que permita listar los archivos y directorios de la carpeta **ejercicio\_1**, indicando en cada caso si se trata de un archivo (F) o un directorio (D).  
Los archivos y directorios existentes sirven como prueba, durante la corrección, el número de archivos y directorios puede variar.  
No se pedirá ningún dato al usuario por teclado.
2. **(2,5 puntos)** Crea una clase Java que solicite al usuario una palabra y nos muestre en pantalla el número de veces que aparece dicha palabra en el archivo **ejercicio\_2.txt**. Se contarán como validas todas las apariciones de la palabra buscada, sin diferenciar mayúsculas o minúsculas, tanto si aparece de forma individual o forma parte de una palabra compuesta.  
Ejemplo:      palabra              Total: 4 veces  
Texto: **palabra palaBra** pa-la-bra pasapalabra para todo **palabra**.
3. **(1,5 puntos)** Crea una clase Java que permita realizar una copia del archivo de video **ejercicio\_3.mkv**. El nombre del nuevo archivo será **copia\_ejercicio\_3.mkv**. El nuevo archivo creado, debe poder abrirse con el programa “Películas y Tv” de Windows y mostrar el mismo contenido que el original.
4. **(4,5 puntos)** En este ejercicio simularemos aspectos básicos de un ORM. Para ello debéis completar la clase **Ejercicio4** de tal forma que realice las siguientes tareas:
  - I. Al iniciar, rellenar el ArrayList **actores**, con los actores creados a partir de los datos obtenidos de la tabla **actor** de la base de datos **DAW\_codigoAlumno**.
    - i. Para la creación de los actores desde la base de datos debéis utilizar el constructor **public Actor(int id, String nombre, String apellidos, LocalDate fechaNacimiento)** de la clase Actor.
    - ii. Para la realización de la consulta a la base de datos no se pueden utilizar **procedimientos almacenados** (prepared).
  - II. Al finalizar, sincronizar los valores de nuestro ArrayList **actores** con la base de datos.
    - i. Para saber si un objeto actor se ha creado nuevo, ha sido modificado o se debe eliminar, la clase **Actor** cuenta con los métodos:
      1. **public boolean isNuevo()**
      2. **public boolean isModificado()**
      3. **public boolean isEliminado()**
    - ii. Debido a que se pueden realizar multiples sentencias similares contra la base de datos, debéis usar **procedimientos almacenados** (prepared) para la creación de las sentencias SQL.

Para facilitar el trabajo de completar la clase Ejercicio4, se han colocado comentarios en los lugares donde he borrado código de la solución.

### Datos a utilizar:

Base de datos	sakila
Tabla	actor_codigoAlumno
Servidor	172.21.29.50
Puerto	3306
Usuario	userDAW
Contraseña	Java
SQL	SELECT * FROM actor INSERT INTO actor (first_name, last_name, last_update) VALUES(?,?,?) UPDATE actor SET first_name = ?, last_name = ?, last_update = ? WHERE actor_id = ? DELETE FROM actor WHERE actor_id = ?
Relación de campos:	first_name = <b>nombre</b> -----String last_name = <b>apellidos</b> -----String last_update = <b>fecha de cumpleaños</b> ---LocalDate actor_id = <b>id del actor</b> -----int

Condiciones para la evaluación y calificación del examen:

- Debéis entregar el proyecto de Netbeans con el código que hayáis generado.
- Para la comprobación de los ejercicios se ejecutará el proyecto de Netbeans entregado por el alumno.
- Las clases que no se puedan ejecutar o no funcionen en el ordenador del profesor se considerarán como erróneas y no se valorarán.
- No usar rutas absolutas para la realización de los ejercicios con archivos.
- En el caso de usar herramientas o tecnologías (clases Java) que no hayamos explicado en clase, el alumno deberá justificar razonadamente el motivo y ventajas de la utilización de la nueva clase frente a las impartidas en clase.