

CICLO FORMATIVO EN DESARROLLO DE APLICACIONES WEB PROGRAMACIÓN - 2ª EVALUACIÓN - EXAMEN B		Fecha ABRIL/2021	Calificación
Nombre	1er Apellido	2º Apellido	

Subir los ficheros del proyecto en la tarea llamada examenEvaluacion2 del aula virtual en un archivo comprimido zip al aulavirtual. El proyecto y el fichero zip deberá llamarse nombreapellido2. El tiempo para realizarlo es de 1:30 minutos. Utilizar para hacerlo el entorno de desarrollo Netbeans 12 y Java jdk 14. Usaremos la base de datos Mysql versión 8. Hay que tener una cámara para que se os vea la pantalla y una parte de vosotros. Identificarse al entrar con el dni. Los primeros 30 minutos son para identificarse y ver que todo funciona bien. El audio debe estar activado y con cascos para no molestar. El proyecto de netbeans podrá ser ant o con maven.

Ejercicio 1: (2 puntos)

Pon la respuesta en el fichero de texto llamado Pregunta1.txt. Consta las siguientes preguntas:

- 1.1 Como debe ser una variable de una clase, cuyo valor sea compartido por todas las clase, y se pueda usar como contador de objetos creados. Pon un ejemplo.
- 1.2 Como debe ser una variable de una clase para que se pueda acceder a ella desde la misma clase y desde las clases que la heredan. Por un ejemplo.

Ejercicio 2: (2 puntos)

Realizar un programa en Java que cree un fichero de texto separado por comas que está ubicado en la carpeta raíz del proyecto usando una dirección relativa.

Ejemplo del fichero de texto alumnos.txt:

```
1;Felipe Garcia;a2
2;Paco Aldarias;b2
```

Además tendremos los siguientes ficheros:

alumnos.txt. Es el fichero de texto separado por punto y coma, con los datos.

Persona.java. Será la clase que describa los datos de las personas. Deberá encapsular los datos. Describirá los objetos personas que tienen un id numérico, el nombre de tipo String. Deberá tener la función public String toString() que deberá ser implementado por la clase herede. Implementará los métodos set y get de las variables de la clase.

Alumno.java. Tendrá como atributo propio el NIA. Heredará de Alumno todos sus atributos y funciones. Implementaremos la función toString que devuelve los valores de todos los atributos de un Alumno. Ejemplo de valor devuelto seria: "1 Felipe Garcia a2".

Almacen.java. Será un interface con la siguiente cabecera public void grabar(Alumno a)

Ficheros.java. Clase que implementará el interface Almacen.java, el cual recibirá un objeto persona que grabará en un fichero. Se definirá una constante String para indicar el fichero que se usa para grabar.

Pregunta2.java. Será una clase que creará dos objetos de la clase Alumno, que llamará a la función grabar de la clase Almacen para almacenarlos en un fichero. Seguidamente se imprimirá en pantalla usando la función toString de la clase Alumno.

Ejercicio 3: (2 puntos)

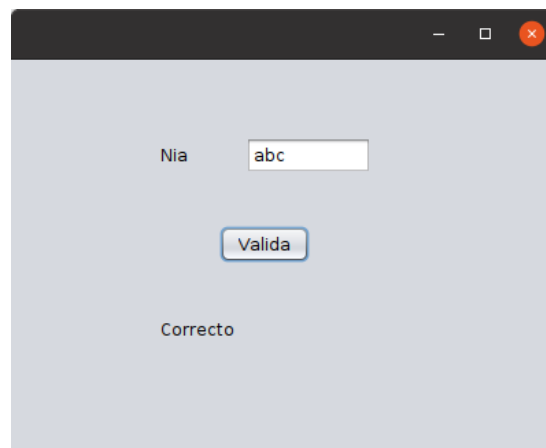
Hacer una aplicación para que pida datos de un alumno por pantalla y si la longitud no es 3, salte la excepción e indique que es incorrecto.

La clase que gestiona la excepción se llamará MiExcepcion.java. El programa donde se encuentra la función main se llamará Ejercicio3.java

Ejercicio 4: (2 puntos)

Realizar un programa que valide el NIA, en Java usando el **entorno gráfico** de Netbeans. La app debe pedir el NIA por teclado. Crear un botón llamado Fichero que al pulsarlo consultará los datos del fichero de alumnos y mostrará un mensaje diciendo Correcto o Incorrecto.

Llamalo Pregunta4.java.



Ejercicio 5: (2 puntos)

Modifica el botón en el ejercicio anterior llamalo Buscar que consulte sobre una **base de datos** la existencia del NIA. Mostrar un mensaje que diga si existe o no. Llamarlo Pregunta5.java.



Crea una clase llamada Mysql.java que contenga todas las operaciones sobre la base de datos mysql. Debemos crear las funciones public void conectar() y boolean existeNIA(Alumno al) que serán usados por la Pregunta5.java.

El fichero personas.sql que crearía los datos en Mysql seria:

```
CREATE TABLE alumnos (  
id int(3) NOT NULL,  
nombre varchar(20),  
nia varchar(15)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
INSERT INTO alumnos (id, nombre,) VALUES  
(1, 'Paco','a12'),  
(2, 'Juan','b34');
```

```
ALTER TABLE alumnos  
ADD PRIMARY KEY (id);
```

Introduce el fichero .sql y la librería jdbc en la carpeta raíz del proyecto. Se puede usar maven introduciendo en el pom:

```
<dependency>  
  <groupId>mysql</groupId>  
  <artifactId>mysql-connector-java</artifactId>  
  <version>8.0.23</version>  
</dependency>
```