

## EXÁMENES PRESENCIALES FP A DISTANCIA JUNIO 2025



CICLO FORMATIVO:  
MÓDULO PROFESIONAL:

FECHA Y HORA:

DURACIÓN:

DAM

ACCESO A DATOS (AD)

Martes, 3 de Junio de 2025  
15:30 -18:15

2 horas y 45 minutos



APELLIDOS, NOMBRE:		DNI:	
I.E.S. DE REFERENCIA:	IES Aguadulce		
I.E.S. DONDE SE REALIZA EL EXAMEN:			

Selecciona tu profesor o profesora asignado:

DAM	[ ] Diana Sola García
-----	-----------------------

### INSTRUCCIONES GENERALES:

- El examen consiste en dos parciales: Febrero y Junio y la duración total será de 2 horas y 45 minutos.
- El parcial de Febrero es obligatorio para el alumnado que no se presentó en Febrero o que suspendió dicho examen.
- El parcial de Junio es obligatorio para todo el alumnado.
- Los RA (Resultados de Aprendizaje) y CE (Criterios de Evaluación) que se evalúan en cada parcial están detallados en la tabla al comienzo del examen.

### PUNTUACIÓN:

- La puntuación de cada parcial será de 10 puntos y constará de dos partes:  
Parte Teórica: 5 puntos + Parte Práctica: 5 puntos
- La nota de cada parcial se calculará como la suma de la parte teórica (5 puntos) y de la parte práctica (5 puntos).
- La parte teórica en cada parcial constará de 20 preguntas tipo test con 4 posibles respuestas, de la que sólo una es correcta. Cada pregunta correcta puntuará con 0,25 puntos y cada incorrecta **NO** resta. Si se deja sin contestar ni suma ni resta.

### REALIZACIÓN DEL EXAMEN:

- El examen se realizará SIN ORDENADOR y NO SE PERMITIRÁ ningún tipo de apuntes.
- Debe realizarse con bolígrafo AZUL o NEGRO. No se podrá utilizar ningún otro color ni tampoco un LÁPIZ.
- Para la parte práctica se podrán utilizar todos los folios que se requieran. NO SE PERMITIRÁ salir del examen con ningún folio.

### ENTREGA:

- Las hojas del examen se entregarán numeradas en la parte inferior, en el orden correcto y serán grapadas al examen.
- Se debe presentar el DNI al entregar el examen y cumplimentar la hoja de firmas.

## PARCIAL FEBRERO

### PARTE TEÓRICA ( 5 PUNTOS)

### PARTE PRÁCTICA (5 PUNTOS)

**EJERCICIO 1.** En un fichero de texto llamado **inventario.txt** se almacenan los datos de los productos de un comercio. De cada producto se guarda la siguiente información: referencia, nombre, cantidad y precio. Cada registro de un producto se almacena en una línea, separando sus campos por el carácter '&'. Por ejemplo, en un momento dado el fichero podría contener la siguiente información:

```
001&Leche&200&2.15
022&Agua&50&5.20
027&Pan&30&1.50
047&Atún&40&2.30
```

Suponiendo que el fichero está ubicado en el directorio actual, debes escribir el código del programa que lea el fichero e imprima por pantalla una lista numerada con el nombre de los productos y su precio. Por ejemplo, para los datos anteriores generaría la siguiente salida **(2,5 puntos)**:

```
1. Leche: 2.15 €
2. Agua: 5.20 €
3. Pan: 1.50 €
4. Atún: 2.30 €
```

```
public static void main(String[] args) {

    try (BufferedReader br = new BufferedReader(new FileReader("inventario.txt"))){
        //Mediante esta variable almacenaremos las lineas
        String linea = null;
        //El while lee linea a linea
        int num=1;
        while ((linea = br.readLine()) != null) {
            String[]campos = linea.split("&");
            System.out.println(num + ". " +campos[1]+": " +campos[3]+ " €");
            num++;
        }
    } catch (IOException e) {
        System.out.println("Error: " + e.getMessage());
    }
}
```

**EJERCICIO 2.** Partiendo de una base de datos MySQL que consta de una única tabla y un procedimiento almacenado que se ha generado con el siguiente script:

```
-- Crear la base de datos
CREATE DATABASE alumnosBD;
USE alumnosBD;

-- Crear tabla Alumno
CREATE TABLE Alumno (
    NIF VARCHAR(9) PRIMARY KEY,
    nombre_completo VARCHAR(60) NOT NULL,
    repetidor BOOLEAN,
    modulos_matriculados INT
);

-- Crear procedimiento almacenado
DELIMITER //
CREATE PROCEDURE nuevoAlumno
(IN dni VARCHAR(9), IN nombre VARCHAR(30), IN repite BOOLEAN, IN total_mod INT)
BEGIN
    INSERT INTO Alumno VALUES (dni, nombre, repite, total_mod);
END//
DELIMITER ;
```

Completa las siguientes líneas del código, correspondientes al método **gestionAlumnos** que deberá establecer la conexión con la base de datos e insertar una nueva alumna haciendo uso del procedimiento almacenado. Ésta nueva alumna nunca ha estado matriculada, se llama "María Pérez", su dni es "22223333F" y se ha matriculado en 3 módulos (2,5 puntos).

```
public static void gestionAlumnos() {
    try {

        // Obtener la conexión a la base de datos con MySQL en localhost
        // con usuario y contraseña del servidor de MySQL: root y root

        Connection conexion =
        DriverManager.getConnection("jdbc:mysql://localhost/alumnosBD", "root", "root");

        // Preparar la llamada al procedimiento almacenado

        CallableStatement prcProcedimientoAlmacenado =
            conexion.prepareCall("{ call nuevoAlumno(?, ?, ?, ?) }");

        // cargar parametros en el procedimiento almacenado

        prcProcedimientoAlmacenado.setString("dni", "22223333F");

        prcProcedimientoAlmacenado.setString("nombre", "María Pérez");
```

## EXÁMENES PRESENCIALES FP A DISTANCIA JUNIO 2025



CICLO FORMATIVO:  
MÓDULO PROFESIONAL:

FECHA Y HORA:

DURACIÓN:

DAM

ACCESO A DATOS (AD)

Martes, 3 de Junio de 2025  
15:30 -18:15

2 horas y 45 minutos



```
prcProcedimientoAlmacenado.setBoolean("repite", false);
```

---

```
prcProcedimientoAlmacenado.setInt("total_mod", 3);
```

---

```
// ejecutar el procedimiento
```

---

```
prcProcedimientoAlmacenado.execute();
```

---

```
} catch (SQLException se) {  
    System.out.println("SQL Exception: "+ se.toString());  
} catch (Exception e) {  
    System.out.println("Exception: "+ e.toString());  
}
```

```
}
```

## PARCIAL JUNIO

### PARTE TEÓRICA (5 PUNTOS)

### PARTE PRÁCTICA (5 PUNTOS)

**EJERCICIO 1. (2.5 puntos)** Se dispone de una base de datos objeto-relacional que utiliza el gestor de objetos Db4o. Esta base de datos se mantiene en un fichero llamados **misEmpleados.db4o** y se utiliza para llevar el control de los empleados de una empresa. Cada objeto Empleado se caracteriza por su **nif, nombre, apellidos y salario**. Además se supone ya creada la clase Empleado con todos los métodos get y set implementados y el método toString() también. Teniendo esto en cuenta, escribir el código Java que realizaría las siguientes acciones (no debes escribir el programa completo sólo las instrucciones o sentencias necesarias para realizar la acción del apartado):

a) Conectar con la base de datos. **(0,5 puntos)**

```
try {
    ObjectContainer bd = Db4oEmbedded.openFile(Db4oEmbedded.newConfiguration(), "misEmpleados.db4o");
    System.out.println("Conexión exitosa con BD: ");
}
catch (Db4oIOException | DatabaseFileLockedException | IncompatibleFileFormatException | OldFormatException
| DatabaseReadOnlyException e){
    System.out.println("Error al conectar con la base de datos: " + e.getMessage());
}
```

b) Añadir a la base de datos un empleado llamado Sergio Gómez Sánchez, cuyo dni es 99999999P y con un salario de 1950,75 euros, suponiendo que "db" es el objeto que representa la conexión ya establecida. **(0,6 puntos)**

```
try{
    Empleado emp = new Empleado ("99999999P", "Sergio", "Gómez Sánchez", 1950.75);
    db.store(emp);
    db.commit();
}
catch (Db4oIOException | DatabaseClosedException | DatabaseReadOnlyException e){
    System.out.println("Error BD: " + e.getMessage());
    db.rollback();
}
```

c) Aumentar el salario en un 20% de un empleado (ya almacenado) cuyo dni es 55577766J, suponiendo que "db" es el objeto que representa la conexión ya establecida. **(0,6 puntos)**

```
try{
    Empleado e = new Empleado("55577766J", null, null, 0);
    res = db.queryByExample(e);
    Empleado e2 = (Empleado) res.next();
}
```

## EXÁMENES PRESENCIALES FP A DISTANCIA JUNIO 2025



CICLO FORMATIVO:  
MÓDULO PROFESIONAL:

FECHA Y HORA:

DURACIÓN:

DAM

ACCESO A DATOS (AD)

Martes, 3 de Junio de 2025  
15:30 -18:15

2 horas y 45 minutos



```
e2.setSalario(e2.getSalario()*1.2);
db.store(e2);
db.commit();
}
catch (Db4oIOException | DatabaseClosedException | DatabaseReadOnlyException e){
    System.out.println("Error BD: "+ e.getMessage());
    db.rollback();
}
```

d) Mostrar por pantalla los empleados que ganan entre 1500 y 1800 euros, suponiendo que “db” es el objeto que representa la conexión ya establecida. **(0,8 puntos)**

```
try{
    Query query = db.query();
    query.constrain(Empleado.class);
    query.descend("salario").constrain(1500).greater();
    query.descend("salario").constrain(1800).smaller();
    res= query.execute();
    if (!res.isEmpty()){
        while (res.hasNext()){
            System.out.println(res.next().toString());
        }
    }
    else {
        System.out.println("no hay ningún empleado con ese salario");
    }
}
catch (DatabaseClosedException e){
    System.out.println("Error BD: "+ e.getMessage());
}
```

**EJERCICIO 2. (2,5 puntos)** El fichero **biblioteca.xml** contiene una colección de libros y tiene una estructura similar al siguiente ejemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<biblioteca>
  <libro id="123">
    <titulo>El Principito</titulo>
    <autor>
      <nombre>Antoine</nombre>
      <apellidos>de Saint-Exupéry</apellidos>
    </autor>
    <fecha_edi>2000</fecha_edi>
    <veces_prestado>55</veces_prestado>
  </libro>
  .
  .
  .
```

# EXÁMENES PRESENCIALES FP A DISTANCIA JUNIO 2025



CICLO FORMATIVO:  
MÓDULO PROFESIONAL:

FECHA Y HORA:

DURACIÓN:

DAM

ACCESO A DATOS (AD)

Martes, 3 de Junio de 2025  
15:30 -18:15

2 horas y 45 minutos



```
<libro>
.
.
.
</libro>
</biblioteca>
```

Suponiendo que el fichero está ubicado en el directorio “src” del siguiente proyecto al que se le ha añadido el driver BaseX.jar, completa las líneas de código que faltarían para que el código realice lo indicado en los comentarios.

```
public class Ejercicio2 {

    private static Context contexto;
    private static String directorioBD = "src/biblioteca.xml";

    public static void main(String[] args) throws QueryException {

        /* Conectar con la base de datos (0,5 puntos) */
        if (contexto == null) {
            contexto = new Context();

            try {
                CreateDB bd = new CreateDB("Biblioteca", directorioBD);

                bd.execute(contexto);

                /* Mostrar el título de los libros cuya fecha de edición
                 sea anterior a 1990 o que haya sido prestado más de 30
                 veces (1 punto) */

                XQuery xq1 = new XQuery ("for $x in //libro where $x/fecha_edi <
1990 or $x/veces_prestado>30 return $x/titulo/text()");

                String resultado = xq1.execute(contexto);
                System.out.println(resultado);

                /* Sustituir el valor del nodo veces_prestado a 100 en el
                 libro cuyo id es '456' (1 punto)*/

                XQuery xq2 = "for $x in //libro[@id =456] return replace value of
node $x/veces_prestado with 100";

                XQuery xq2 = new XQuery(query1);
                xq2.execute(contexto);

                // cerrar base de datos
                contexto.close();

            } catch (BaseXException ex) {
```

## EXÁMENES PRESENCIALES FP A DISTANCIA JUNIO 2025



**CICLO FORMATIVO:**  
**MÓDULO PROFESIONAL:**

**FECHA Y HORA:**

**DURACIÓN:**

**DAM**

**ACCESO A DATOS (AD)**

**Martes, 3 de Junio de 2025**  
**15:30 -18:15**

**2 horas y 45 minutos**



```
System.err.println(ex.getMessage());  
    }  
}  
}
```