

## EXAMEN TIPO A: 2ª Evaluación Bases de Datos.

Grupo: 1º SI  
Fecha: 15/03/24

Curso: 23/24  
Tiempo: 1h 50'

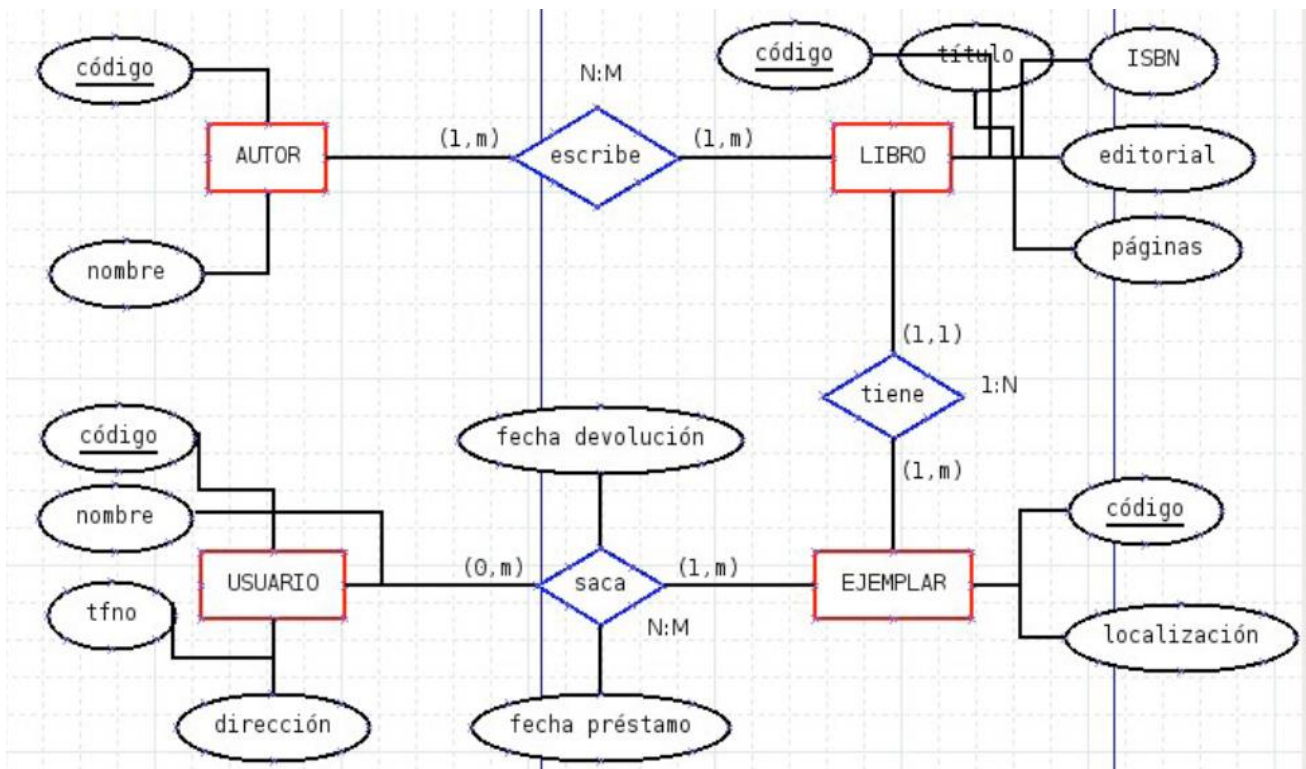
### PARTE 1: Hay que sacar mínimo un 5 en esta primera parte para mediar

#### 1.- Realiza una BD en ORACLE basada en el modelo Entidad-Relación para el caso siguiente : (100 %)

Nuestra BD será la de una biblioteca. Será interesante poblar de datos la BD creada para poder detectar posibles casos de error y la consistencia y seguridad de nuestra BD.

La valoración de los elementos de la BD será la siguiente:

- (60%) Creación de las tablas de la BD
- (40%) Creación de las relaciones de la BD
- Cada fallo en cada una de las partes restará del total del apartado según la gravedad del error:
  - Fallos leves : 5% (por ejemplo tipos de datos mal escogidos)
  - Fallos graves : 10% (por ejemplo una FK olvidada o PK mal)
  - Fallos muy graves : 20% (por ejemplo elementos no hechos)



## **PARTE 2: Hay que sacar mínimo un 5 en esta primera parte para mediar**

### **1.- Realiza las consultas en SQL de ORACLE basada en la BD de la parte anterior: (100 %)**

1. (10%) Libros que tenemos de la editorial “ANAYA” con más de 150 pgs
2. (15% ) Localización de TODOS los ejemplares de los libros de la editorial “ANAYA” con más de 150 pgs
3. (25%) Código de Ejemplares que hayan sido prestados pero todavía NO devueltos
4. (25%) Nombre y teléfono de los usuarios que hayan cogido prestado algún libro en los últimos 30 días.
5. (25%) Número total de autores que tengan los libros de la editorial “ANAYA” con más de 150 pgs

## **CREACION TABLAS TIPO A**

### **TABLAS:**

**AUTOR**

**LIBRO**

**AUTOR-LIBRO (VIENE DE LA RELACION N-M) -> TENDRÁ FKS**  
(codigo\_autor / codigo\_libro)

**EJEMPLAR**

**USUARIO**

**USUARIO - EJEMPLAR (VIENE DE LA RELACION N-M) -> TENDRÁ FKS**  
(codigo\_usuario / codigo\_ejemplar) y ademas fecha\_prestamo fecha\_devolucion

### -----CREACIÓN TABLAS-----

#### **TABLA AUTOR:**

```
CREATE TABLE AUTOR(  
codigo_autor NUMBER(3),  
nombre_autor VARCHAR2(35),  
CONSTRAINT pk_codigo_autor PRIMARY KEY (codigo_autor),  
CONSTRAINT nn_nombre_autor CHECK (nombre_autor IS NOT NULL)  
);
```

#### **TABLA LIBRO:**

```
CREATE TABLE LIBRO(  
codigo_libro NUMBER(3),  
titulo_libro VARCHAR2(30),  
isbn_libro NUMBER(30),  
editorial_libro VARCHAR2(15),  
paginas_libro NUMBER(7),  
CONSTRAINT pk_codigo_libro PRIMARY KEY (codigo_libro),  
CONSTRAINT nn_titulo_libro CHECK (titulo_libro IS NOT NULL),  
CONSTRAINT nn_isbn_libro CHECK (isbn_libro IS NOT NULL),  
CONSTRAINT nn_editorial_libro CHECK (editorial_libro IS NOT NULL),  
CONSTRAINT nn_paginas_libro CHECK (paginas_libro IS NOT NULL),  
CONSTRAINT ck_paginas_libro CHECK (paginas_libro > 0)  
);
```

#### **TABLA AUTOR\_LIBRO:**

```
CREATE TABLE AUTOR_LIBRO(  
codigo_autor NUMBER(3),  
codigo_libro NUMBER(3),  
CONSTRAINT fk_codigo_autor FOREIGN KEY (codigo_autor) REFERENCES AUTOR  
(codigo_autor),  
CONSTRAINT fk_codigo_libro FOREIGN KEY (codigo_libro) REFERENCES LIBRO  
(codigo_libro),  
CONSTRAINT pk2_codigo_autor_codigo_libro PRIMARY KEY (codigo_autor, codigo_libro)  
);
```

# FUNDACIÓN SAN VALERO CES

## TABLA EJEMPLAR

```
CREATE TABLE EJEMPLAR(  
codigo_libro NUMBER(3),  
codigo_ejemplar NUMBER(3),  
localizacion_ejemplar VARCHAR2(50),  
CONSTRAINT pk_codigo_ejemplar PRIMARY KEY (codigo_ejemplar),  
CONSTRAINT fkk_codigo_libro FOREIGN KEY (codigo_libro) REFERENCES LIBRO  
(codigo_libro),  
CONSTRAINT nn_localizacion_ejemplar CHECK (localizacion_ejemplar IS NOT NULL)  
);
```

## TABLA USUARIO:

```
CREATE TABLE USUARIO(  
codigo_usuario NUMBER(3),  
nombre_usuario VARCHAR2(50),  
telefono_usuario NUMBER(9),  
direccion_usuario VARCHAR2(30),  
CONSTRAINT pk_codigo_usuario PRIMARY KEY (codigo_usuario),  
CONSTRAINT nn_nombre_usuario CHECK (nombre_usuario IS NOT NULL),  
CONSTRAINT nn_telefono_usuario CHECK (telefono_usuario IS NOT NULL),  
CONSTRAINT nn_direccion_usuario CHECK (direccion_usuario IS NOT NULL)  
);
```

## TABLA USUARIO\_EJEMPLAR:

```
CREATE TABLE USUARIO_EJEMPLAR(  
codigo_usuario NUMBER(3),  
codigo_ejemplar NUMBER(3),  
fecha_prestamo DATE,  
fecha_devolucion DATE,  
CONSTRAINT fk_codigo_usuario FOREIGN KEY (codigo_usuario) REFERENCES USUARIO  
(codigo_usuario),  
CONSTRAINT fk_codigo_ejemplar FOREIGN KEY (codigo_ejemplar) REFERENCES  
EJEMPLAR (codigo_ejemplar),  
CONSTRAINT nn_fecha_prestamo CHECK (fecha_prestamo IS NOT NULL)  
);
```

-----INSERTS-----

**TABLA LIBRO:**

INSERT INTO LIBRO VALUES (1, 'Don Quijote', 9788424115724, 'ANAYA', 250);  
INSERT INTO LIBRO VALUES (2, 'Cien años de soledad', 9788420471832, 'ANAYA', 320);  
INSERT INTO LIBRO VALUES (3, 'Harry Potter', 9788478884452, 'Salamandra', 400);

**TABLA AUTOR:**

INSERT INTO AUTOR VALUES (1, 'Miguel de Cervantes');  
INSERT INTO AUTOR VALUES (2, 'Gabriel García Márquez');  
INSERT INTO AUTOR VALUES (3, 'J.K. Rowling');

**TABLA AUTOR\_LIBRO:**

INSERT INTO AUTOR\_LIBRO VALUES (1, 1);  
INSERT INTO AUTOR\_LIBRO VALUES (2, 2);  
INSERT INTO AUTOR\_LIBRO VALUES (3, 3);

**TABLA EJEMPLAR:**

INSERT INTO EJEMPLAR VALUES (1, 101, 'Sección A - Estantería 5');  
INSERT INTO EJEMPLAR VALUES (2, 102, 'Sección B - Estantería 2');  
INSERT INTO EJEMPLAR VALUES (3, 103, 'Sección C - Estantería 7');

**TABLA USUARIO:**

INSERT INTO USUARIO VALUES (1, 'Carlos López', 654789321, 'Calle Mayor 45');  
INSERT INTO USUARIO VALUES (2, 'Laura García', 612345678, 'Avenida del Sol 23');  
INSERT INTO USUARIO VALUES (3, 'Pedro Martínez', 698765432, 'Plaza del Pilar 12');

**TABLA USUARIO\_EJEMPLAR:**

INSERT INTO USUARIO\_EJEMPLAR VALUES (1, 101, TO\_DATE('2024-03-01', 'YYYY-MM-DD'), TO\_DATE('2024-03-10', 'YYYY-MM-DD'));  
INSERT INTO USUARIO\_EJEMPLAR VALUES (2, 102, TO\_DATE('2024-04-01', 'YYYY-MM-DD'), NULL);  
INSERT INTO USUARIO\_EJEMPLAR VALUES (3, 103, TO\_DATE('2024-04-15', 'YYYY-MM-DD'), TO\_DATE('2024-04-20', 'YYYY-MM-DD'));

## -----CONSULTAS-----

1. Libros que tenemos de la editorial “ANAYA” con más de 150 pgs

```
SELECT titulo_libro, editorial_libro
FROM LIBRO
WHERE UPPER(editorial_libro) = 'ANAYA'
AND paginas_libro > 150;
```

2. Localización de TODOS los ejemplares de los libros de la editorial “ANAYA” con más de 150 pgs

```
SELECT LIBRO.titulo_libro, LIBRO.editorial_libro, EJEMPLAR.localizacion_ejemplar
FROM LIBRO, EJEMPLAR
WHERE LIBRO.codigo_libro = EJEMPLAR.codigo_libro
AND UPPER(editorial_libro) = 'ANAYA'
AND paginas_libro > 150;
```

3. Código de Ejemplares que hayan sido prestados pero todavía NO devueltos

usaremos 3 tablas: LIBRO, EJEMPLAR, USUARIO, EJEMPLAR (CONTIENE LAS FECHAS)

```
SELECT EJEMPLAR.codigo_ejemplar, USUARIO_EJEMPLAR.fecha_prestamo
FROM LIBRO, EJEMPLAR, USUARIO_EJEMPLAR
WHERE LIBRO.codigo_libro = EJEMPLAR.codigo_libro
AND EJEMPLAR.codigo_ejemplar = USUARIO_EJEMPLAR.codigo_ejemplar
AND USUARIO_EJEMPLAR.fecha_devolucion IS NULL;
```

4. Nombre y teléfono de los usuarios que hayan cogido prestado algún libro en los últimos 30 días.

usaremos 2 tablas: USUARIO, USUARIO\_EJEMPLAR

```
INSERT INTO USUARIO_EJEMPLAR VALUES (2, 102, TO_DATE('2025-03-01', 'YYYY-MM-DD'),
TO_DATE('2025-03-10', 'YYYY-MM-DD'));
```

```
SELECT USUARIO.nombre_usuario, USUARIO.telefono_usuario
FROM USUARIO, USUARIO_EJEMPLAR
WHERE USUARIO.codigo_usuario = USUARIO_EJEMPLAR.codigo_usuario
AND fecha_prestamo >= SYSDATE - 30;
```

5. Número total de autores que tengan los libros de la editorial “ANAYA” con más de 150 pgs

```
SELECT COUNT(DISTINCT AUTOR.codigo_autor) AS TOTAL_AUTORES
FROM AUTOR, LIBRO, AUTOR_LIBRO
WHERE AUTOR.codigo_autor = AUTOR_LIBRO.codigo_autor
AND LIBRO.codigo_libro = AUTOR_LIBRO.codigo_libro
AND UPPER(LIBRO.editorial_libro) = 'ANAYA'
AND LIBRO.paginas_libro > 150;
```