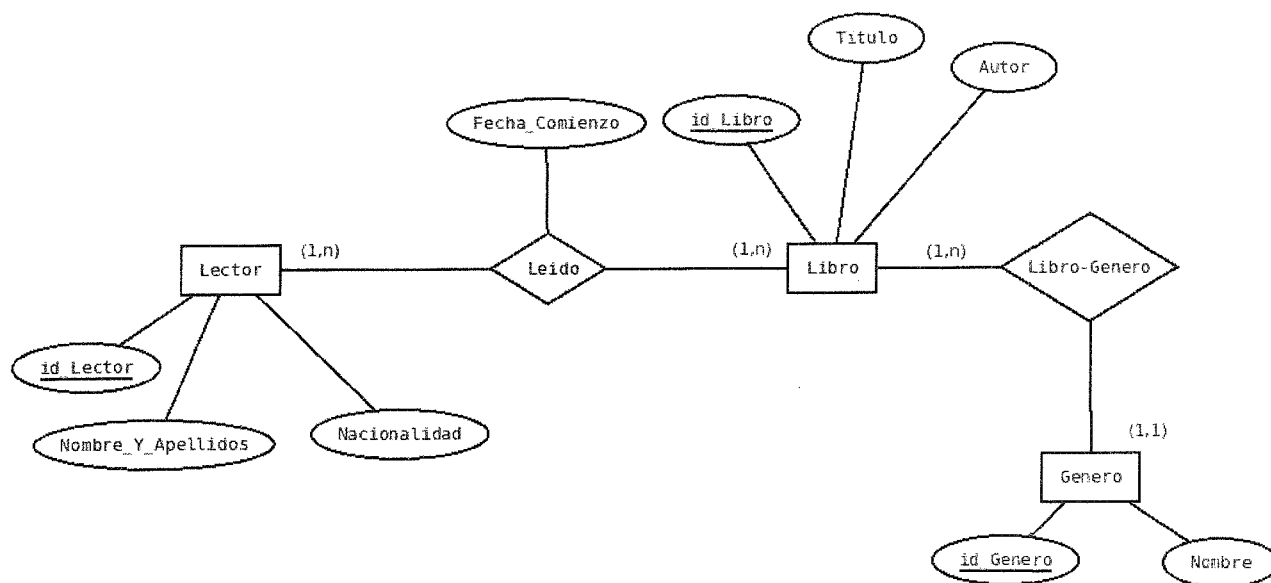


Orden de 21 de febrero de 2025, por la que se efectúa convocatoria de procedimiento selectivo para el ingreso en los Cuerpos de Profesores de Enseñanza Secundaria, Profesores de Escuelas Oficiales de Idiomas, Profesores de Música y Artes Escénicas y de Maestros.

### EJERCICIO 1: BASES DE DATOS (Total: 2,5 puntos).

Observe el siguiente diagrama entidad/relación:



Tenga en cuenta las siguientes definiciones y restricciones:

- id\_Lector: Entero.
- Nombre\_Y\_Apellidos: Cadena de 50 caracteres. No puede estar vacío.
- Nacionalidad: Cadena de 20 caracteres. No puede estar vacío.
- Fecha\_Comienzo: Fecha. No puede estar vacío. Representa la fecha en que un lector comienza a leer un determinado libro.
- id\_Libro: Entero.
- Título: Cadena de 40 caracteres. No puede estar vacío.
- Autor: Cadena de 40 caracteres. No puede estar vacío.
- id\_Genero: Entero.
- Nombre: Cadena de 40 caracteres. No puede estar vacío.

#### EJERCICIO 1.1. CREACIÓN DE TABLAS (1,25 puntos)

Detalle las sentencias SQL necesarias para crear las tablas de la base de datos que implemente el modelo anterior en tercera forma normal.



## EJERCICIO 1.2. CONSULTA (1,25 puntos)

Dadas las tablas que ha realizado en el apartado anterior, detalle las sentencias SQL necesarias para realizar la siguiente consulta:

Todos los libros del género 'Terror' que se leyeron en 2024 por lectores italianos (Nacionalidad = 'Italia') con las cantidades de lecturas acumuladas en dicho periodo. Se considerarán leídos en 2024 todos los libros que se comenzaron a leer entre el 1 de Enero y el 31 de diciembre de 2024.

Debe aparecer una línea por cada libro que cumpla dicho requisito, sin repeticiones. Cada línea contendrá los siguientes campos: *Título*, *Autor* y *Total\_Lecturas*.

## Ejercicio 1: Bases de datos (Total: 2,5 puntos)

### Ejercicio 1.1 Creación de tablas (1,25 puntos)

Tabla lector:

```
CREATE TABLE Lector (  
    id_Lector INT PRIMARY KEY,  
    Nombre_Y_Apellidos VARCHAR(50) NOT NULL,  
    Nacionalidad VARCHAR(20) NOT NULL  
);
```

Tabla género:

```
CREATE TABLE Genero (  
    id_Genero INT PRIMARY KEY,  
    Nombre VARCHAR(40) NOT NULL  
);
```

Tabla libro:

```
CREATE TABLE Libro (  
    id_Libro INT PRIMARY KEY,  
    Titulo VARCHAR(40) NOT NULL,  
    Autor VARCHAR(40) NOT NULL,  
    id_Genero INT NOT NULL,  
    FOREIGN KEY (id_Genero) REFERENCES genero(id_Genero)  
);
```

Tabla leído:

```
CREATE TABLE Leido (  
    id_Libro INT NOT NULL,  
    id_Lector INT NOT NULL,  
    Fecha_Comienzo DATE NOT NULL,  
    PRIMARY KEY (id_Libro, id_Lector, Fecha_Comienzo),  
    FOREIGN KEY (id_Libro) REFERENCES Libro(id_Libro),  
    FOREIGN KEY (id_Lector) REFERENCES Lector(id_Lector)  
);
```

Email: info@informaticaoposicioneducacion.com WhatsApp y teléfono: 624409705

Página web: <https://informaticaoposicioneducacion.com>

Comentario sobre "FECHA\_COMIENZO" en tabla "Leído":

La solución dada considera el atributo "Fecha\_Comienzo" como el tercer atributo de la clave primaria.

A la hora de resolver el ejercicio hay dos opciones, añadir "Fecha\_Comienzo" a la clave primaria o no añadirlo, veamos que implica cada opción. Antes de ello, nótese que en la rúbrica proporcionada sí se dice que Fecha\_Comienzo es parte de la clave primaria.

|                                                                                                                                                                                                                                                                                                   |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| EJERCICIO 1.1 - ÍTEM 2: Todas las tablas poseen clave primaria (primary key) que será id_XXX, salvo en la tabla Leídos, cuya clave primaria estará formada por Lector*, Libro* y Fecha_Comienzo. (*) Los nombres de las claves ajenas pueden cambiar, ya que no están especificadas. (Peso: 2.5%) |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

**Añadir "Fecha\_Comienzo" a la clave primaria:**

Al añadir "Fecha\_Comienzo" a la clave primaria implica que un lector puede leer el mismo libro más de una vez siempre que no se empiece en el mismo día. Algo que tiene sentido y que deberíamos darnos cuenta.

**No añadir "Fecha\_Comienzo" a la clave primaria:**

Al no añadir "Fecha\_Comienzo" a la clave primaria implica que un lector solo puede leer un libro en particular una única vez, algo que no acaba de encajar mucho. Por otra parte, el diagrama entidad-relación no tiene subrayado el atributo "Fecha\_Comienzo" por lo que siguiendo estrictamente el diagrama no debería formar parte de la clave primaria.

**Conclusión propia:**

Personalmente prefiero la primera opción, no solo porque en la rúbrica lo dice por correcto, sino porque a pesar de no estar representado en el diagrama (atributo subrayado) es la que más sentido tiene.

Dicho esto, siempre recomiendo al alumnado poner una explicación en caso de ambigüedad, ejemplo para cada caso:

- 1) **Sí añadir:** El atributo "Fecha\_Comienzo" no está subrayado y por lo tanto no formaría parte de la clave primaria, aun así he decidido hacerlo parte de ella ya que tiene sentido que un libro se puede leer más de una vez por la misma persona. Si no, simplemente se quitaría "Fecha\_Comienzo" y cada libro solo podría ser leído una única vez por cada persona.
- 2) **No añadir:** El atributo "Fecha\_Comienzo" no está subrayado y por lo tanto no forma parte de la clave primaria, suponemos que se ha querido representar que un lector solo puede leer una vez un libro (o solo se registra la primera vez), en caso de querer que pudiera leer el mismo libro más de una vez en diferentes días, añadiría "Fecha\_Comienzo" a la clave primaria.

De esta forma queda claro que se controla el tema ante posibles ambigüedades.

## Ejercicio 1.2 Consulta (1,25 puntos)

Se dan varias opciones como respuesta, en la que lo único que cambia es el group by y se explica la diferencia entre ellas, mostrando el típico error que se comete.

### Opción 1, correcta

```
SELECT Libro.Titulo, Libro.Autor, COUNT(*) AS Total_Lecturas
FROM Leido
JOIN Lector ON Leido.id_Lector = Lector.id_Lector
JOIN Libro ON Leido.id_Libro = Libro.id_Libro
JOIN Genero ON Libro.id_Genero = Genero.id_Genero
WHERE
  Lector.Nacionalidad = 'Italia'
  AND Genero.Nombre = 'Terror'
  AND Leido.Fecha_Comienzo BETWEEN '2024-01-01' AND '2024-12-31'
GROUP BY Libro.id_Libro, Libro.Titulo, Libro.Autor;
```

|   | Titulo       | Autor          | Total_Lecturas |
|---|--------------|----------------|----------------|
| ▶ | La Maldición | Giovanni Verdi | 2              |
|   | La Maldición | Giovanni Verdi | 1              |
|   | El Susurro   | Andrea Neri    | 1              |

Nótese que se ha añadido “Libro.id\_Libro”, ¿por qué? Ya que Titulo y Autor no tiene una UNIQUE nadie nos dice que no se puede repetir el nombre y el titulo del autor, siendo libros diferentes, gracias a poner el “id\_Libro” evitamos tal error. En nuestro ejemplo, podemos imaginar que el primer libro “la Maldición” es del año 1800 y el segundo de 2010.

Si ejecutáramos la misma consulta añadiendo en el select “Libro.id\_Libro” veríamos la diferencia.

|   | id_Libro | Titulo       | Autor          | Total_Lecturas |
|---|----------|--------------|----------------|----------------|
| ▶ | 101      | La Maldición | Giovanni Verdi | 2              |
|   | 102      | El Susurro   | Andrea Neri    | 1              |
|   | 104      | La Maldición | Giovanni Verdi | 1              |

### Opción 2, incorrecta, en algunas SGDB funciona:

Consulta igual que la Opción1, pero cambiamos el GroupBy

```
Group by Libro.id_Libro
```

En algunos SGDB esto funciona y es correcto porque tiene en cuenta las dependencias funcionales que se derivan del atributo “Libro.id\_Libro”, pero en otras no, se recuerda que esto no cumple el estándar de SQL. Por ejemplo, es correcto en MySQL pero no en Oracle.

No se recomienda usar esta forma ni aunque se diga que la base de datos es MySQL.

**Email:** [info@informaticaoposicioneducacion.com](mailto:info@informaticaoposicioneducacion.com) **WhatsApp y teléfono:** 624409705

**Página web:** <https://informaticaoposicioneducacion.com>

Opción 3, incorrecta:

Consulta igual que la Opción1, pero cambiamos el GroupBy

```
Group by Libro.Titulo, Libro.Autor;
```

|   | Título       | Autor          | Total_Lecturas |
|---|--------------|----------------|----------------|
| ▶ | La Maldición | Giovanni Verdi | 3              |
|   | El Susurro   | Andrea Neri    | 1              |

Al contrario que en la Opción 1 no tiene en cuenta el "Id\_Libro", por lo tanto, nos da un resultado incorrecto. En nuestro ejemplo, a unificado el primer libro "la Maldición" del año 1800 y el segundo de 2010 como si fueran el mismo libro, obteniendo un resultado incorrecto.

<https://informaticaoposicioneducacion.com> todos los derechos reservados