

Se dispone del siguiente esquema relacional para contener una base de datos de publicaciones científicas (libros, capítulos, artículos de revistas, ponencias de congresos o páginas web):

- Documento(nd, título, fecha, estado, público, resumen, soporte, localización, idioma, idioma-original)
- Autor(código, nombre, organización, dirección, teléfono, fax, email)
- Crear(nd, autor)
- PalabraClave(nd, palabra)
- Libro(nd, editorial, nº-edición, isbn, nº-páginas)
- Artículo(nd, nombre-revista, issn, volumen, número, página-inicial, página-final)
- Ponencia(nd, tipo, nombre-congreso, nº-edición, organizador)
- PáginaWeb(nd, fecha-revisión)
- Congreso(nombre, edición, organizador, tipo-evento, carácter-evento, lugar, fecha-inicio, fecha-fin, libro-actas)

Con las claves ajenas:

Crear.nd -> Documento	Crear.autor -> Autor
PalabraClave.nd -> Documento	Libro.nd -> Documento
Artículo.nd -> Documento	Ponencia.nd -> Documento
Ponencia.{ nombre-congreso, nº-edición, organizador} -> Congreso	
PáginaWeb.nd -> Documento	Congreso.libro-actas -> Libro

### Ejercicio 1 (4'0 puntos)

a) Suponiendo que ya existen las demás tablas con sus correspondientes restricciones PRIMARY KEY y FOREIGN KEY según se indica en el esquema anterior, escriba las sentencias SQL para crear las tablas Congreso y Ponencia. Se debe permitir añadir filas a Congreso aunque no esté añadida en Libro la fila de su libro de actas.

b) Modifique la tabla Congreso para incluir las siguientes restricciones:

- el carácter de un evento es "anual" o "bianual".
- la fecha de inicio y fin deben tener valor y además la segunda debe ser posterior o igual a la primera.

Solución:

A)

Primero se debe crear Congreso para que luego se pueda crear Ponencia con la clave ajena apuntando a Congreso.

Para cumplir el requisito de Congreso basta con no poner NOT NULL en la clave ajena libro-actas.

```
CREATE TABLE Congreso (
    nombre          CHAR(50),
    edicion         INTEGER,
    organizador     CHAR(50),
    tipo-evento     CHAR(1),
    caracter-evento CHAR(10),
    lugar           CHAR(80),
    fecha-inicio    DATE,
```

```
fecha-fin          DATE,  
libro-actas        INTEGER REFERENCES Libro,  
PRIMARY KEY (nombre, edicion, organizador)  
);
```

```
CREATE TABLE Ponencia (  
    nd INTEGER PRIMARY KEY,  
    tipo CHAR(10),  
    nombre_congreso CHAR(50),  
    n_edicion INTEGER,  
    organizador CHAR(50),  
    FOREIGN KEY (nd) References Documento,  
    FOREIGN KEY (nombre_congreso,n_edicion,organizador) References  
Congreso  
);
```

B)

```
ALTER TABLE Congreso  
    ADD CONSTRAINT caracter_evento_chk  
    CHECK (caracter_evento IN ('Annual','Bianual'));
```

```
ALTER TABLE Congreso  
    ADD CONSTRAINT fechas_chk  
    CHECK (fecha_inicio IS NOT NULL AND fecha_fin IS NOT NULL AND  
fecha_inicio<=fecha_fin);
```

## Ejercicio 2 (6'0 puntos)

- Crear una vista (Total\_Publicaciones) que muestre las publicaciones de tipo libro y artículo de cada autor (num\_doc, cod\_autor, nombre\_autor, título, fecha).
- Utilizando la vista anterior crear una consulta que muestre un resumen de las publicaciones de cada autor, indicando el número de publicaciones de cada tipo (libros y artículos) que ha realizado cada autor.

Solución:

A)

```
CREATE VIEW Total_Publicaciones (num_doc, cod_autor, nombre_autor, título,  
fecha, tipo) AS  
(  
    SELECT Doc.nd, Crear.autor, Autor.nombre, Doc.titulo, Doc.fecha, 'Libro' as  
Tipo  
    FROM Documento Doc, Crear, Autor, Libro  
    WHERE Doc.nd=Crear.nd  
    AND Autor.codigo=Crear.autor  
    AND Libro.nd=Doc.nd  
UNION
```

Tipo  
SELECT Doc.nd, Crear.autor, Autor.nombre, Doc.titulo, Doc.fecha, 'Artículo' as  
FROM Documento Doc, Crear, Autor, Articulo  
WHERE Doc.nd=Crear.nd  
AND Autor.codigo=Crear.autor  
AND Articulo.nd=Doc.nd  
);

B)  
SELECT cod\_autor, tipo, Count(num\_doc)  
FROM Total\_Publicaciones  
GROUP BY Cod\_autor, tipo;