Bases de Datos

Sprint 2 Lab 2

Felipe Izquierdo Romero

Índice

1. Descripción del problema

Crear en una base de datos relacional las tablas de recomendación y educación, con sus relaciones, para el curriculum de un candidato. De la misma manera en una base de datos no relacional.

Opcional hacer consultas cruzadas en la base de datos relacional como listar a las personas que sepan inglés o que tengan alguna recomendación.

2. Desarrollo

En el paso a paso se referencian las tablas con una tabla intermedia, pero en el siguiente ejercicio para variar se referenciarán con una foreign key o clave foránea que hará referencia al id del candidato. Cada registro de las tablas recomendaciones ye studios estarán relacionados con el candidato a través de esa clave foránea.

Además de este apunte cabe destacar como se guardan las fechas puesto que en educación solo se necesita el año se guarda en una variable de tipo YEAR.

Para recomendaciones se usa un tipo DATE que recogerá la fecha y lógicamente en la parte del Front-End solo se mostrará como en la imagen el mes y el año.

```
35  CREATE TABLE IF NOT EXISTS educacion(
36      id INT AUTO_INCREMENT,
37      id_candidato INT,
38      entidad VARCHAR(128),
39      tipo_formacion VARCHAR(128),
40      anio YEAR,
41      descripcion VARCHAR(512),
42      PRIMARY KEY (id),
43      FOREIGN KEY (id_candidato) REFERENCES candidato(id)
44  );
45
46  CREATE TABLE IF NOT EXISTS recomendaciones(
47      id INT AUTO_INCREMENT,
48      id_candidato INT,
49      recomendador VARCHAR (128),
50      entidad VARCHAR(128),
51      descripcion VARCHAR(128),
52      mes DATE,
53      PRIMARY KEY (id),
54      FOREIGN KEY (id_candidato) REFERENCES candidato(id)
55  );
56
```

Ilustración 1- Tablas recomendaciones y educación

En una base de datos no relacional podrían existir dos maneras de hacerlo o más. En este caso una opción es dividir en dos colecciones diferentes educación y recomendaciones. Donde la referencia seria guardar el identificador único de cada candidato.

Ilustración 2- Colección de relaciones

Ilustración 3- Colección de educación

Otra opción podría ser en la colección de candidato añadir un array para la clave de recomendaciones y otro array para la clave estudios.

Dentro de cada lista objetos diferentes con claves idénticas, pero valores diferentes.

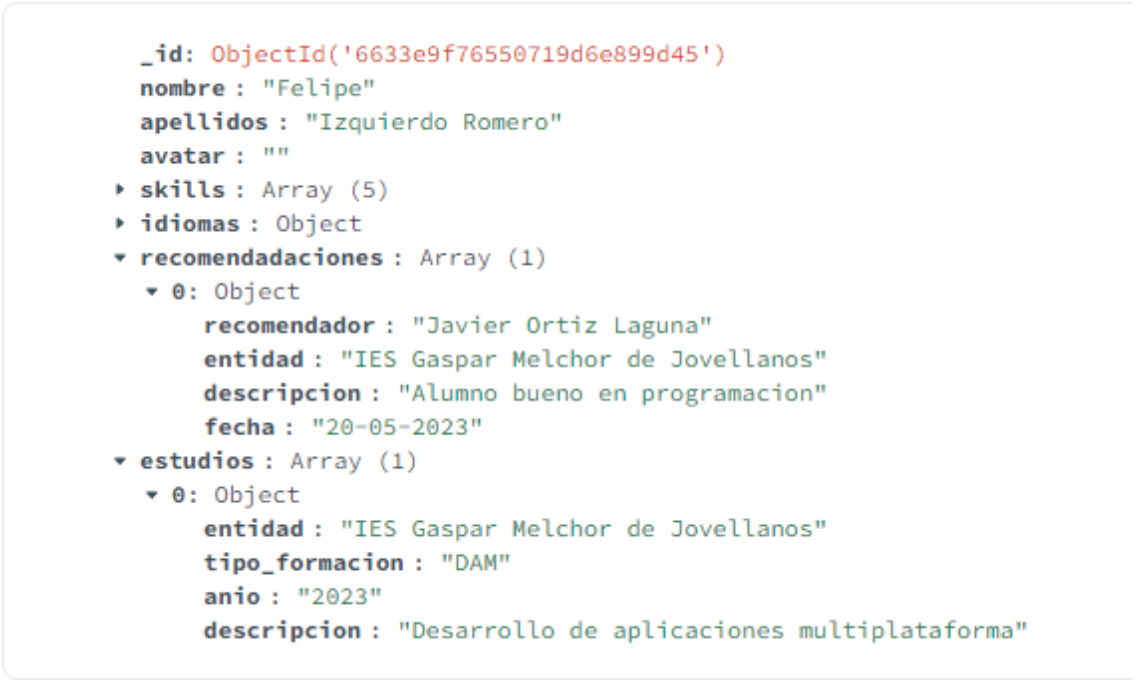

Ilustración 4- Candidato con array de recomendaciones y educación

3. Opcional

Dado como se guardan los tipos de datos estas dos sentencias serían suficiente para mostrar a las personas que saben inglés y que tienen alguna recomendación.

```
30    /*__________SELECCION DE CANDIDATOS QUE SABEN INGLES_________________*/
31 ●  select CONCAT(can.nombre, ' ' ,can.apellido) as Candidato , i.nombre as Idioma , ci.nivel
32    from candidato_idioma ci
33    left join candidato can on ci.id_candidato = can.id
34    left join idioma i on ci.id_idioma = i.id
35    where ci.id_idioma = 1
36    order by Idioma;
37
38    /*__________SELECCION DE CANDIDATO CON RECOMENDACIONES_________________*/
39 ●  select CONCAT(can.nombre, ' ' ,can.apellido) as Candidato, rec.recomendador, rec.entidad, rec.descripcion, rec.mes
40    from recomendaciones rec
41    left join candidato can on can.id = rec.id_candidato
42    where can.id >= 1;
43
```

Ilustración 5- Consultas cruzadas

Además, en el documento .sql habrá más consultas para retornar todos los datos de un usuario filtrando por su identificador.