

TECNOLOGICO NACIONAL DE MEXICO  
INSTITUTO TECNOLÓGICO DE MORELIA  
“José María Morelos y Pavón”

Taller de Base de Datos  
Amaury Aurelio Coria Ramírez

# Base de Datos Red Social FACEBOOK

Carlos Jahir Castro Cázares  
Jaime Isaí Velázquez Aguilar  
Ernesto Vieyra Melo  
Víctor Hugo Ávila Pérez

Ingeniería en Sistemas Computacionales  
Grupo: A

Semestre Ene-Jun 2019  
04 de marzo del 2019

## ***Tablas que poseen relaciones***

- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
- Lugares\_de\_trabajo (ID\_TRABAJO, TRABAJO)
- Relación: Perfil\_trabajo(ID\_PERFIL, ID\_TRABAJO, FECH\_INI, FECH\_FIN)
  - Entidad Fuerte: Perfil
  - Entidad Débil: Lugares\_de\_trabajo
  - Cardinalidad: 1 a N, 1 Perfil N lugares
  - Cardinalidad mínima: 1 perfil no requiere ningún lugar
  - Participación total: No existe, 1 perfil no se relaciona con todos los lugares de trabajo
  
- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
- Albumes (ID\_ALBUM, ID\_PERFIL, NOMBRE)
  - Entidad Fuerte: Perfil
  - Entidad Débil: Albumes
  - Cardinalidad: 1 a N, 1 Perfil N albumes
  - Cardinalidad mínima: 1 perfil no requiere ningún álbum
  - Participación total: No existe, 1 perfil no se relaciona con todos los álbumes
  
- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
- Titulos\_academicos (ID\_TITULO, TITULO)
- Relación: Perfil\_titulo(ID\_PERFIL, ID\_TITULO)
  - Entidad Fuerte: Perfil
  - Entidad Débil: Titulos\_academicos
  - Cardinalidad: 1 a N, 1 Perfil N títulos
  - Cardinalidad mínima: 1 perfil no requiere ningún titulo
  - Participación total: No existe, 1 perfil no se relaciona con todos los títulos

- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
  - Certificación (ID\_CERTIFICACION, CERTIFICACION)
  - Relación: Perfil\_certifi (ID\_PERFIL, ID\_CERTIFICACION)
    - Entidad Fuerte: Perfil
    - Entidad Débil: Certificación
    - Cardinalidad: 1 a N, 1 Perfil N certificados
    - Cardinalidad mínima: 1 perfil no requiere ningún certificado
    - Participación total: No existe, 1 perfil no se relaciona con todos los certificados
- 
- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
  - Películas (ID\_PELICULAS, PELICULA)
  - Relación: Peliculas\_fav (ID\_PERFIL, ID\_PELICULA)
    - Entidad Fuerte: Perfil
    - Entidad Débil: Películas
    - Cardinalidad: 1 a N, 1 Perfil N películas
    - Cardinalidad mínima: 1 perfil no requiere ninguna película
    - Participación total: No existe, 1 perfil no se relaciona con todas las películas
- 
- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
  - Cantantes (ID\_CANTANTE, CANTANTE)
  - Relación: Cantantes\_fav (ID\_PERFIL, ID\_CANTANTE)
    - Entidad Fuerte: Perfil
    - Entidad Débil: Cantantes
    - Cardinalidad: 1 a N, 1 Perfil N cantantes
    - Cardinalidad mínima: 1 perfil no requiere ningún cantante
    - Participación total: No existe, 1 perfil no se relaciona con todos los cantantes

- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
  - Libros (ID\_LIBROS, LIBRO)
  - Relación: Libros\_fav (ID\_PERFIL, ID\_LIBROS)
    - Entidad Fuerte: Perfil
    - Entidad Débil: Libros
    - Cardinalidad: 1 a N, 1 Perfil N libros
    - Cardinalidad mínima: 1 perfil no requiere ningún libro
    - Participación total: No existe, 1 perfil no se relaciona con todos los libros
- 
- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
  - Amigos (ID\_PERFIL, ID\_AMIGO)
    - Entidad Fuerte: Perfil
    - Entidad Débil: Perfil
    - Cardinalidad: N a N, Perfil N amigos
    - Cardinalidad mínima: 1 perfil no requiere ningún amigo
    - Participación total: No existe, 1 perfil no se relaciona con todos los perfiles (Amigo)
- 
- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
  - Historial (ID\_HISTORIAL, ID\_PERFIL, TEXT, IMAGEN, LIGA)
    - Entidad Fuerte: Perfil
    - Entidad Débil: Historial
    - Cardinalidad: 1 a 1, Perfil 1 Historial
    - Cardinalidad mínima: 1 perfil requiere 1 historial
    - Participación total: No existe, 1 perfil solo se relaciona con un historial

- Historial (ID\_HISTORIAL, ID\_PERFIL, TEXT, IMAGEN, LIGA)
- Publicacion (ID\_PUBLICACION, ID\_HISTORIAL, TEXTO, DIA-HORA, FOTO, ID\_UBICACION, ID\_ESTADO)
  - Entidad Fuerte: Historial
  - Entidad Débil: Publicacion
  - Cardinalidad: 1 a N, 1 Historial N Publicacion
  - Cardinalidad mínima: 1 Historial no requiere ninguna publicacion
  - Participación total: No existe, un historial no se relaciona con todas las publicaciones y una publicación no se relaciona con todos los historiales
- Publicacion (ID\_PUBLICACION, ID\_HISTORIAL, TEXTO, DIA-HORA, FOTO, ID\_UBICACION, ID\_ESTADO)
- Ubicación (ID\_UBICACION, UBICACION)
  - Entidad Fuerte: Ubicacion
  - Entidad Débil: Publicacion
  - Cardinalidad: N a 1, N Publicaciones tienen 1 Publicacion
  - Cardinalidad mínima: 1 Publicacion requiere 1 Ubicación
  - Participación total: No existe, una publicación solo requiere una ubicación
- Publicacion (ID\_PUBLICACION, ID\_HISTORIAL, TEXTO, DIA-HORA, FOTO, ID\_UBICACION, ID\_ESTADO)
- Estado (ID\_ESTADO, ESTADO)
  - Entidad Fuerte: Estado
  - Entidad Débil: Publicacion
  - Cardinalidad: N a 1, N Publicaciones tienen 1 Estado
  - Cardinalidad mínima: 1 Publicacion requiere un Estado
  - Participación total: No existe, una publicación solo requiere un Estado

- Publicacion (ID\_PUBLICACION, ID\_HISTORIAL, TEXTO, DIA-HORA, FOTO, ID\_UBICACION, ID\_ESTADO)
- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
- Relación: Me\_Gusta (ID\_PERFIL, ID\_PUBLICACION)
  - Entidad Fuerte: Publicacion
  - Entidad Débil: Perfil
  - Cardinalidad: N a N, N publicaciones pueden tener N perfiles
  - Cardinalidad mínima: 1 Publicacion no requiere ningún perfil (Me gusta)
  - Participación total: Si ya que una publicación puede tener todos los perfiles con me gusta
  
- Publicacion (ID\_PUBLICACION, ID\_HISTORIAL, TEXTO, DIA-HORA, FOTO, ID\_UBICACION, ID\_ESTADO)
- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
- Relación: Etiquetado(ID\_PERFIL, ID\_PUBLICACION)
  - Entidad Fuerte: Publicacion
  - Entidad Débil: Perfil
  - Cardinalidad: N a N, N publicaciones pueden tener N perfiles
  - Cardinalidad mínima: 1 Publicacion no requiere ningún perfil (Etiquetado)
  - Participación total: Si ya que una publicación puede tener todos los perfiles etiquetados
  
- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
- Amigos (ID\_PERFIL, ID\_AMIGO)
  - Entidad Fuerte: Perfil
  - Entidad Débil: Perfil
  - Cardinalidad: N a N, Perfil N amigos
  - Cardinalidad mínima: 1 perfil no requiere ningún amigo
  - Participación total: No existe, 1 perfil no se relaciona con todos los perfiles (Amigo)

- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
- Historial (ID\_HISTORIAL, ID\_PERFIL, TEXT, IMAGEN, LIGA)
  - Entidad Fuerte: Perfil
  - Entidad Débil: Historial
  - Cardinalidad: 1 a 1, Perfil 1 Historial
  - Cardinalidad mínima: 1 perfil requiere 1 historial
  - Participación total: No existe, 1 perfil solo se relaciona con un historial
  
- Historial (ID\_HISTORIAL, ID\_PERFIL, TEXT, IMAGEN, LIGA)
- Publicacion (ID\_PUBLICACION, ID\_HISTORIAL, TEXTO, DIA-HORA, FOTO, ID\_UBICACION, ID\_ESTADO)
  - Entidad Fuerte: Historial
  - Entidad Débil: Publicacion
  - Cardinalidad: 1 a N, 1 Historial N Publicacion
  - Cardinalidad mínima: 1 Historial no requiere ninguna publicacion
  - Participación total: No existe, un historial no se relaciona con todas las publicaciones y una publicación no se relaciona con todos los historiales
  
- Publicacion (ID\_PUBLICACION, ID\_HISTORIAL, TEXTO, DIA-HORA, FOTO, ID\_UBICACION, ID\_ESTADO)
- Ubicación (ID\_UBICACION, UBICACION)
  - Entidad Fuerte: Ubicacion
  - Entidad Débil: Publicacion
  - Cardinalidad: N a 1, N Publicaciones tienen 1 Publicacion
  - Cardinalidad mínima: 1 Publicacion requiere 1 Ubicación
  - Participación total: No existe, una publicación solo requiere una ubicación
  
- Publicacion (ID\_PUBLICACION, ID\_HISTORIAL, TEXTO, DIA-HORA, FOTO, ID\_UBICACION, ID\_ESTADO)
- Estado (ID\_ESTADO, ESTADO)
  - Entidad Fuerte: Estado
  - Entidad Débil: Publicacion
  - Cardinalidad: N a 1, N Publicaciones tienen 1 Estado
  - Cardinalidad mínima: 1 Publicacion requiere un Estado
  - Participación total: No existe, una publicación solo requiere un Estado

- Publicacion (ID\_PUBLICACION, ID\_HISTORIAL, TEXTO, DIA-HORA, FOTO, ID\_UBICACION, ID\_ESTADO)
- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
- Relación: Me\_Gusta (ID\_PERFIL, ID\_PUBLICACION)
  - Entidad Fuerte: Publicacion
  - Entidad Débil: Perfil
  - Cardinalidad: N a N, N publicaciones pueden tener N perfiles
  - Cardinalidad mínima: 1 Publicacion no requiere ningún perfil (Me gusta)
  - Participación total: Si ya que una publicación puede tener todos los perfiles con me gusta

- Publicacion (ID\_PUBLICACION, ID\_HISTORIAL, TEXTO, DIA-HORA, FOTO, ID\_UBICACION, ID\_ESTADO)
- Perfil (ID\_PERFIL, NOMBRE, CORREO, FECH\_NACI)
- Relación: Etiquetado(ID\_PERFIL, ID\_PUBLICACION)
  - Entidad Fuerte: Publicacion
  - Entidad Débil: Perfil
  - Cardinalidad: N a N, N publicaciones pueden tener N perfiles
  - Cardinalidad mínima: 1 Publicacion no requiere ningún perfil (Etiquetado)
  - Participación total: Si ya que una publicación puede tener todos los perfiles etiquetados



# Código LDD

-- MySQL Workbench Forward Engineering

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';
```

```
-----
-- Schema BD_Facebook
-----
```

```
-----
-- Schema BD_Facebook
-----
```

```
CREATE SCHEMA IF NOT EXISTS `BD_Facebook` DEFAULT CHARACTER SET utf8mb4 ;
USE `BD_Facebook` ;
```

```
-----
-- Table `BD_Facebook`.`Titulos_Academicos`
-----
```

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Titulos_Academicos` (
  `ID_TITULO` INT NOT NULL AUTO_INCREMENT,
  `TITULO` VARCHAR(50) NOT NULL,
  PRIMARY KEY (`ID_TITULO`))
ENGINE = InnoDB;
```

```
-----
-- Table `BD_Facebook`.`Lugar_Trabajo`
-----
```

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Lugar_Trabajo` (
  `ID_LUGAR-TRABAJO` INT NOT NULL AUTO_INCREMENT,
  `TRABAJO` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`ID_LUGAR-TRABAJO`))
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Certificacion`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Certificacion` (  
  `ID_CERTIFICACION` INT NOT NULL AUTO_INCREMENT,  
  `CERTIFICACION` VARCHAR(50) NOT NULL,  
  PRIMARY KEY (`ID_CERTIFICACION`))  
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Peliculas`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Peliculas` (  
  `ID-PELICULA` INT NOT NULL AUTO_INCREMENT,  
  `PELICULA` VARCHAR(100) NOT NULL,  
  PRIMARY KEY (`ID-PELICULA`))  
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Cantante`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Cantante` (  
  `ID_CANTANTE` INT NOT NULL AUTO_INCREMENT,  
  `CANTANTE` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`ID_CANTANTE`))  
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Libro`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Libro` (  
  `ID_LIBRO` INT NOT NULL AUTO_INCREMENT,  
  `LIBRO` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`ID_LIBRO`))  
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Perfil`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Perfil` (  
  `ID_PERFIL` INT NOT NULL AUTO_INCREMENT,  
  `NOMBRE` VARCHAR(45) NOT NULL,  
  `CORREO` VARCHAR(45) NOT NULL,  
  `FECH_NACI` DATE NOT NULL,  
  PRIMARY KEY (`ID_PERFIL`))  
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Estado`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Estado` (  
  `ID_ESTADO` INT NOT NULL AUTO_INCREMENT,  
  `ESTADO` VARCHAR(40) NOT NULL,  
  PRIMARY KEY (`ID_ESTADO`))  
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Ubicacion`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Ubicacion` (  
  `ID_UBICACION` INT NOT NULL AUTO_INCREMENT,  
  `UBICACION` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`ID_UBICACION`))  
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Perfil-Titulos\_Academicos`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Perfil-Titulos_Academicos` (  
  `ID_PERFIL` INT NOT NULL,  
  `ID_TITULO` INT NOT NULL,  
  INDEX `fk_Perfil_has_Titulos_Academicos_Titulos_Academicos1_idx` (`ID_TITULO`  
ASC),  
  INDEX `fk_Perfil_has_Titulos_Academicos_Perfil_idx` (`ID_PERFIL` ASC),  
  CONSTRAINT `fk_Perfil_has_Titulos_Academicos_Perfil`  
    FOREIGN KEY (`ID_PERFIL`)  
    REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Perfil_has_Titulos_Academicos_Titulos_Academicos1`
```

```

FOREIGN KEY (`ID_TITULO`)
REFERENCES `BD_Facebook`.`Titulos_Academicos` (`ID_TITULO`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `BD_Facebook`.`Perfil-Lugar_Trabajo`
-----

```

```

CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Perfil-Lugar_Trabajo` (
  `ID_PERFIL` INT NOT NULL,
  `ID_LUGAR-TRABAJO` INT NOT NULL,
  `FECHA-FIN` DATE NOT NULL,
  `FECHA-INI` DATE NOT NULL,
  INDEX `fk_Perfil_has_Lugar_Trabajo_Lugar_Trabajo1_idx` (`ID_LUGAR-TRABAJO`
ASC),
  INDEX `fk_Perfil_has_Lugar_Trabajo_Perfil1_idx` (`ID_PERFIL` ASC),
  CONSTRAINT `fk_Perfil_has_Lugar_Trabajo_Perfil1`
    FOREIGN KEY (`ID_PERFIL`)
    REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Perfil_has_Lugar_Trabajo_Lugar_Trabajo1`
    FOREIGN KEY (`ID_LUGAR-TRABAJO`)
    REFERENCES `BD_Facebook`.`Lugar_Trabajo` (`ID_LUGAR-TRABAJO`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `BD_Facebook`.`Perfil-Certificacion`
-----

```

```

CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Perfil-Certificacion` (
  `ID_PERFIL` INT NOT NULL,
  `ID_CERTIFICACION` INT NOT NULL,
  INDEX `fk_Perfil_has_Certificacion_Certificacion1_idx` (`ID_CERTIFICACION` ASC),
  INDEX `fk_Perfil_has_Certificacion_Perfil1_idx` (`ID_PERFIL` ASC),
  CONSTRAINT `fk_Perfil_has_Certificacion_Perfil1`
    FOREIGN KEY (`ID_PERFIL`)
    REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Perfil_has_Certificacion_Certificacion1`
    FOREIGN KEY (`ID_CERTIFICACION`)
    REFERENCES `BD_Facebook`.`Certificacion` (`ID_CERTIFICACION`)

```

```
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-----
-- Table `BD_Facebook`.`Fav-Peliculas`
-----
```

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Fav-Peliculas` (
  `ID_PERFIL` INT NOT NULL,
  `ID-PELICULA` INT NOT NULL,
  INDEX `fk_Perfil_has_Peliculas_Peliculas1_idx` (`ID-PELICULA` ASC),
  INDEX `fk_Perfil_has_Peliculas_Perfil1_idx` (`ID_PERFIL` ASC),
  CONSTRAINT `fk_Perfil_has_Peliculas_Perfil1`
    FOREIGN KEY (`ID_PERFIL`)
      REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  CONSTRAINT `fk_Perfil_has_Peliculas_Peliculas1`
    FOREIGN KEY (`ID-PELICULA`)
      REFERENCES `BD_Facebook`.`Peliculas` (`ID-PELICULA`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-----
-- Table `BD_Facebook`.`Fav-Cantante`
-----
```

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Fav-Cantante` (
  `ID_PERFIL` INT NOT NULL,
  `ID_CANTANTE` INT NOT NULL,
  INDEX `fk_Perfil_has_Cantante_Cantante1_idx` (`ID_CANTANTE` ASC),
  INDEX `fk_Perfil_has_Cantante_Perfil1_idx` (`ID_PERFIL` ASC),
  CONSTRAINT `fk_Perfil_has_Cantante_Perfil1`
    FOREIGN KEY (`ID_PERFIL`)
      REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  CONSTRAINT `fk_Perfil_has_Cantante_Cantante1`
    FOREIGN KEY (`ID_CANTANTE`)
      REFERENCES `BD_Facebook`.`Cantante` (`ID_CANTANTE`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Fav-Libro`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Fav-Libro` (  
  `ID_PERFIL` INT NOT NULL,  
  `ID_LIBRO` INT NOT NULL,  
  INDEX `fk_Perfil_has_Libro_Libro1_idx` (`ID_LIBRO` ASC),  
  INDEX `fk_Perfil_has_Libro_Perfil1_idx` (`ID_PERFIL` ASC),  
  CONSTRAINT `fk_Perfil_has_Libro_Perfil1`  
    FOREIGN KEY (`ID_PERFIL`)  
      REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Perfil_has_Libro_Libro1`  
    FOREIGN KEY (`ID_LIBRO`)  
      REFERENCES `BD_Facebook`.`Libro` (`ID_LIBRO`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Album`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Album` (  
  `ID_ALBUM` INT NOT NULL AUTO_INCREMENT,  
  `NOMBRE` VARCHAR(45) NOT NULL,  
  `Perfil_ID_PERFIL` INT NOT NULL,  
  PRIMARY KEY (`ID_ALBUM`),  
  INDEX `fk_Album_Perfil1_idx` (`Perfil_ID_PERFIL` ASC),  
  CONSTRAINT `fk_Album_Perfil1`  
    FOREIGN KEY (`Perfil_ID_PERFIL`)  
      REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Hitorial`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Hitorial` (  
  `ID_HISTORIAL` INT NOT NULL AUTO_INCREMENT,  
  `ID_PERFIL` INT NOT NULL,  
  `TEXT` TEXT NOT NULL,  
  `IMAGEN` LONGBLOB NOT NULL,  
  `LIGA` VARCHAR(45) NOT NULL,
```

```

INDEX `fk_Perfil_has_Publicacion_Perfil1_idx` (`ID_PERFIL` ASC),
PRIMARY KEY (`ID_HISTORIAL`),
CONSTRAINT `fk_Perfil_has_Publicacion_Perfil1`
  FOREIGN KEY (`ID_PERFIL`)
  REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `BD_Facebook`.`Publicacion`
-----

```

```

CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Publicacion` (
  `ID_PUBLICACION` INT NOT NULL AUTO_INCREMENT,
  `ID_HISTORIAL` INT NOT NULL,
  `TEXTO` LONGTEXT NOT NULL,
  `DIA-HORA` DATETIME NOT NULL,
  `FOTO` BLOB NOT NULL,
  `ID_UBICACION` INT NOT NULL,
  `ID_ESTADO` INT NOT NULL,
  PRIMARY KEY (`ID_PUBLICACION`),
  INDEX `fk_Publicacion_Hitorial1_idx` (`ID_HISTORIAL` ASC),
  INDEX `fk_Publicacion_Ubicacion1_idx` (`ID_UBICACION` ASC),
  INDEX `fk_Publicacion_Estado1_idx` (`ID_ESTADO` ASC),
  CONSTRAINT `fk_Publicacion_Hitorial1`
    FOREIGN KEY (`ID_HISTORIAL`)
    REFERENCES `BD_Facebook`.`Hitorial` (`ID_HISTORIAL`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Publicacion_Ubicacion1`
    FOREIGN KEY (`ID_UBICACION`)
    REFERENCES `BD_Facebook`.`Ubicacion` (`ID_UBICACION`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Publicacion_Estado1`
    FOREIGN KEY (`ID_ESTADO`)
    REFERENCES `BD_Facebook`.`Estado` (`ID_ESTADO`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

-----  
-- Table `BD\_Facebook`.`Amigo`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Amigo` (  
  `ID_PERFIL` INT NOT NULL,  
  `ID_AMIGO` INT NOT NULL,  
  INDEX `fk_Perfil_has_Perfil_Perfil2_idx` (`ID_AMIGO` ASC),  
  INDEX `fk_Perfil_has_Perfil_Perfil1_idx` (`ID_PERFIL` ASC),  
  CONSTRAINT `fk_Perfil_has_Perfil_Perfil1`  
    FOREIGN KEY (`ID_PERFIL`)  
      REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)  
      ON DELETE NO ACTION  
      ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Perfil_has_Perfil_Perfil2`  
    FOREIGN KEY (`ID_AMIGO`)  
      REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)  
      ON DELETE NO ACTION  
      ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

-----  
-- Table `BD\_Facebook`.`Etiquetado`  
-----

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Etiquetado` (  
  `ID_PERFIL` INT NOT NULL,  
  `ID_PUBLICACION` INT NOT NULL,  
  INDEX `fk_Perfil_has_Publicacion_Publicacion2_idx` (`ID_PUBLICACION` ASC),  
  INDEX `fk_Perfil_has_Publicacion_Perfil2_idx` (`ID_PERFIL` ASC),  
  CONSTRAINT `fk_Perfil_has_Publicacion_Perfil2`  
    FOREIGN KEY (`ID_PERFIL`)  
      REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)  
      ON DELETE NO ACTION  
      ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Perfil_has_Publicacion_Publicacion2`  
    FOREIGN KEY (`ID_PUBLICACION`)  
      REFERENCES `BD_Facebook`.`Publicacion` (`ID_PUBLICACION`)  
      ON DELETE NO ACTION  
      ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```



```
-----  
-- Table `BD_Facebook`.`Me_Gusta`  
-----
```

```
CREATE TABLE IF NOT EXISTS `BD_Facebook`.`Me_Gusta` (  
  `ID_PERFIL` INT NOT NULL,  
  `ID_PUBLICACION` INT NOT NULL,  
  INDEX `fk_Perfil_has_Publicacion1_Publicacion1_idx` (`ID_PUBLICACION` ASC),  
  INDEX `fk_Perfil_has_Publicacion1_Perfil1_idx` (`ID_PERFIL` ASC),  
  CONSTRAINT `fk_Perfil_has_Publicacion1_Perfil1`  
    FOREIGN KEY (`ID_PERFIL`)  
      REFERENCES `BD_Facebook`.`Perfil` (`ID_PERFIL`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Perfil_has_Publicacion1_Publicacion1`  
    FOREIGN KEY (`ID_PUBLICACION`)  
      REFERENCES `BD_Facebook`.`Publicacion` (`ID_PUBLICACION`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
SET SQL_MODE=@OLD_SQL_MODE;  
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;  
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

# Índices

Los campos que pensamos que serían candidatos a crear un índice son:

- ID\_PERFIL además de ID\_AMIGO de la tabla Amigo, ya que por la estructura de la base de datos al momento de hacer consultas sobre la tabla Amigo para ver los amigos de cada perfil, por lo tanto, la creación de un índice nos ayudaría mejorar la velocidad de consultas.
- ID\_Publicacion y ID\_Perfil de la tabla Me\_Gusta, porque para el propósito de la base de datos esta tabla será muy usada y ayudaría a mejorar la velocidad de consulta.
- ID\_Perfil y ID\_Publicacion de la tabla Etiquetado, ya que el uso que se le dará a la base de datos, crear un índice en estos campos nos ayudaría a mejorar la velocidad de consulta.
- TITULO de la tabla Titulos\_Academicos, TRABAJO de la Tabla Lugar\_Trabajo, CERTIFICACION de la tabla Certificacion, PELICULA de la tabla Peliculas, CANTANTE de la tabla Cantante y LIBRO de la tabla Libro , todos estos campos tiene algo en común, eso es que son totalmente o parcialmente directorios, ya que los nuevos registros que tendremos en estas tablas, se agregaran raramente, por lo que al ser casi estáticos un índice nos ayudaría a tener una mejor optimización de la base de datos.

Por último, no descartamos la creación de índices en las llaves primarias y foráneas, pero ya que los índices son creados automáticamente al momento de crear la base de datos no lo mencionamos a fondo.

# Sql Server

## Base de datos creada

The screenshot displays the Microsoft SQL Server Management Studio interface. The title bar indicates the file is 'BD\_Facebook (Comandos SQLServer).sql' located at 'DESKTOP-NISOP1P\SQLSERVER\BD\_Facebook (DESKTOP-NISOP1P\Zoaria1 (53))'.

**Object Explorer:** The left pane shows the 'BD\_Facebook' database selected. Under the 'Tables' folder, several tables are listed, including 'dbo.Album', 'dbo.Amigo', 'dbo.Cantante', 'dbo.Certificacion', 'dbo.Estado', 'dbo.Etiquetado', 'dbo.Fav\_Cantante', 'dbo.Fav\_Libro', 'dbo.Fav\_Peliculas', 'dbo.Hitorial', 'dbo.Libro', 'dbo.Lugar\_Trabajo', 'dbo.Me\_Gusta', 'dbo.Peliculas', 'dbo.Perfil', 'dbo.Perfil\_Certificacion', 'dbo.Perfil\_Lugar\_Trabajo', 'dbo.Perfil\_Titulos\_Academicos', 'dbo.Publicacion', 'dbo.Titulos\_Academicos', and 'dbo.Ubicacion'. These tables are highlighted with a red circle.

**Query Editor:** The central pane shows the SQL script used to create the database and tables. The script starts with 'USE BD\_Facebook;' and includes comments for each table. The tables created are:

- Table Titulos\_Academicos:** ID\_TITULO INT NOT NULL IDENTITY(1,1), TITULO VARCHAR(50) NOT NULL, PRIMARY KEY (ID\_TITULO);
- Table Lugar\_Trabajo:** ID\_LUGAR\_TRABAJO INT NOT NULL IDENTITY(1,1), TRABAJO VARCHAR(45) NOT NULL, PRIMARY KEY (ID\_LUGAR\_TRABAJO);
- Table Certificacion:** ID\_CERTIFICACION INT NOT NULL IDENTITY(1,1), CERTIFICACION VARCHAR(50) NOT NULL, PRIMARY KEY (ID\_CERTIFICACION);
- Table Peliculas:** ID\_PELICULA INT NOT NULL IDENTITY(1,1), PELICULA VARCHAR(100) NOT NULL, PRIMARY KEY (ID\_PELICULA);

**Properties:** The right pane shows the 'Current connection parameters' for the connection. The connection name is 'DESKTOP-NISOP1P\SQLSERVER'. The connection state is 'Open'. The server version is '12.0.2000'.

**Status Bar:** The bottom status bar shows 'Connected. (1/1)' and 'DESKTOP-NISOP1P\SQLSERVER ... | DESKTOP-NISOP1P\Zoaria1 (53) | BD\_Facebook | 00:00:00 | 0 rows'.

# Diccionario de datos

DICCIONARIO DE DATOS SQLSERVER.sql - DESKTOP-NISOP1P\SQLEXPRESS.BD\_Facebook (DESKTOP-NISOP1P\Zoarial (52))\*

	object_id	table	column	type	Precision	max_length	Permite Nuls	Es Autonumerico	Descripcion	ForeignKey	ReferenceTableName	ReferenceColumnName
1	821577965	Album	ID_ALBUM	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
2	821577965	Album	NOMBRE	varchar	NULL	45	NO	NO	NULL	NULL	NULL	NULL
3	821577965	Album	Perfil_ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Album_Perfil1	Perfil	ID_PERFIL
4	997578592	Amigo	ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Perfil_Perfil1	Perfil	ID_PERFIL
5	997578592	Amigo	ID_AMIGO	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Perfil_Perfil2	Perfil	ID_PERFIL
6	373576369	Cantante	ID_CANTANTE	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
7	373576369	Cantante	CANTANTE	varchar	NULL	45	NO	NO	NULL	NULL	NULL	NULL
8	309576141	Certificacion	ID_CERTIFICACION	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
9	309576141	Certificacion	CERTIFICACION	varchar	NULL	50	NO	NO	NULL	NULL	NULL	NULL
10	469576711	Estado	ID_ESTADO	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
11	469576711	Estado	ESTADO	varchar	NULL	40	NO	NO	NULL	NULL	NULL	NULL
12	1045578...	Etiquetado	ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Publicacio...	Perfil	ID_PERFIL
13	1045578...	Etiquetado	ID_PUBLICACION	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Publicacio...	Publicacion	ID_PUBLICACION
14	725577623	Fav_Cant...	ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Cantante...	Perfil	ID_PERFIL
15	725577623	Fav_Cant...	ID_CANTANTE	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Cantante...	Cantante	ID_CANTANTE
16	773577794	Fav_Libro	ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Libro_Perfil1	Perfil	ID_PERFIL
17	773577794	Fav_Libro	ID_LIBRO	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Libro_Libro1	Libro	ID_LIBRO
18	677577452	Fav_Pelic...	ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Peliculas...	Perfil	ID_PERFIL
19	677577452	Fav_Pelic...	ID_PELICULA	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Peliculas...	Peliculas	ID_PELICULA
20	869578136	Hitorial	ID_HISTORIAL	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
21	869578136	Hitorial	ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Publicacio...	Perfil	ID_PERFIL
22	869578136	Hitorial	TEXT	text	NULL	16	NO	NO	NULL	NULL	NULL	NULL
23	869578136	Hitorial	IMAGEN	image	NULL	16	NO	NO	NULL	NULL	NULL	NULL
24	869578136	Hitorial	LIGA	varchar	NULL	45	NO	NO	NULL	NULL	NULL	NULL
25	405576483	Libro	ID_LIBRO	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
26	405576483	Libro	LIBRO	varchar	NULL	45	NO	NO	NULL	NULL	NULL	NULL
27	277576027	Lugar_Tra...	ID_LUGAR TRAB...	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
28	277576027	Lugar_Tra...	TRABAJO	varchar	NULL	45	NO	NO	NULL	NULL	NULL	NULL
29	1093578...	Me_Gusta	ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Publicacio...	Perfil	ID_PERFIL
30	1093578...	Me_Gusta	ID_PUBLICACION	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Publicacio...	Publicacion	ID_PUBLICACION
31	341576255	Peliculas	ID_PELICULA	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
32	341576255	Peliculas	PELICULA	varchar	NULL	100	NO	NO	NULL	NULL	NULL	NULL

Query executed successfully.

DESKTOP-NISOP1P\SQLEXPRESS ... DESKTOP-NISOP1P\Zoaria... BD\_Facebook 00:00:00 55 rows

DICCIONARIO DE DATOS SQLSERVER.sql - DESKTOP-NISOP1P\SQLEXPRESS.BD\_Facebook (DESKTOP-NISOP1P\Zoarial (52))\*

	object_id	table	column	type	Precision	max_length	Permite Nuls	Es Autonumerico	Descripcion	ForeignKey	ReferenceTableName	ReferenceColumnName
25	405576483	Libro	ID_LIBRO	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
26	405576483	Libro	LIBRO	varchar	NULL	45	NO	NO	NULL	NULL	NULL	NULL
27	277576027	Lugar_Tra...	ID_LUGAR TRAB...	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
28	277576027	Lugar_Tra...	TRABAJO	varchar	NULL	45	NO	NO	NULL	NULL	NULL	NULL
29	1093578...	Me_Gusta	ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Publicacio...	Perfil	ID_PERFIL
30	1093578...	Me_Gusta	ID_PUBLICACION	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Publicacio...	Publicacion	ID_PUBLICACION
31	341576255	Peliculas	ID_PELICULA	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
32	341576255	Peliculas	PELICULA	varchar	NULL	100	NO	NO	NULL	NULL	NULL	NULL
33	437576597	Perfil	ID_PERFIL	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
34	437576597	Perfil	NOMBRE	varchar	NULL	45	NO	NO	NULL	NULL	NULL	NULL
35	437576597	Perfil	CORREO	varchar	NULL	45	NO	NO	NULL	NULL	NULL	NULL
36	437576597	Perfil	FECH_NACI	date	NULL	3	NO	NO	NULL	NULL	NULL	NULL
37	629577281	Perfil_Certi...	ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Certificaci...	Perfil	ID_PERFIL
38	629577281	Perfil_Certi...	ID_CERTIFICACION	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Certificaci...	Certificacion	ID_CERTIFICACION
39	581577110	Perfil_Lug...	ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Lugar_Tra...	Perfil	ID_PERFIL
40	581577110	Perfil_Lug...	ID_LUGAR TRAB...	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Lugar_Tra...	Lugar_Trabajo	ID_LUGAR_TRABAJO
41	581577110	Perfil_Lug...	FECHA_FIN	date	NULL	3	NO	NO	NULL	NULL	NULL	NULL
42	581577110	Perfil_Lug...	FECHA_INI	date	NULL	3	NO	NO	NULL	NULL	NULL	NULL
43	533576939	Perfil_Titul...	ID_PERFIL	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Titulos_Ac...	Perfil	ID_PERFIL
44	533576939	Perfil_Titul...	ID_TITULO	int	NULL	4	NO	NO	NULL	fk_Perfil_has_Titulos_Ac...	Titulos_Academicos	ID_TITULO
45	917578307	Publicacion	ID_PUBLICACION	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
46	917578307	Publicacion	ID_HISTORIAL	int	NULL	4	NO	NO	NULL	fk_Publicacion_Hitorial1	Hitorial	ID_HISTORIAL
47	917578307	Publicacion	TEXTO	text	NULL	16	NO	NO	NULL	NULL	NULL	NULL
48	917578307	Publicacion	DIA_HORA	dateti...	NULL	8	NO	NO	NULL	NULL	NULL	NULL
49	917578307	Publicacion	FOTO	image	NULL	16	NO	NO	NULL	NULL	NULL	NULL
50	917578307	Publicacion	ID_UBICACION	int	NULL	4	NO	NO	NULL	fk_Publicacion_Ubicaci...	Ubicacion	ID_UBICACION
51	917578307	Publicacion	ID_ESTADO	int	NULL	4	NO	NO	NULL	fk_Publicacion_Estado1	Estado	ID_ESTADO
52	245575913	Titulos_Ac...	ID_TITULO	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
53	245575913	Titulos_Ac...	TITULO	varchar	NULL	50	NO	NO	NULL	NULL	NULL	NULL
54	501576825	Ubicacion	ID_UBICACION	int	NULL	4	NO	SI	NULL	NULL	NULL	NULL
55	501576825	Ubicacion	UBICACION	varchar	NULL	45	NO	NO	NULL	NULL	NULL	NULL

Query executed successfully.

DESKTOP-NISOP1P\SQLEXPRESS ... DESKTOP-NISOP1P\Zoaria... BD\_Facebook 00:00:00 55 rows

