

# Dependencias funcionales y ER/ Relacional/SQL

**Alex Di Genova**

**13/04/2022**

# Outline

- Repaso
- Dependencias funcionales
- ER -> Relacional -> SQL

# Directrices de diseños informales para los esquemas de relación

## Resumen de directrices

- Diseñar un esquema de relación que sea fácil explicar su significado.
- Evitar problemas de inserción, borrado y actualización en relaciones.
- Evite al máximo situar en las relaciones atributos que sean NULL frecuentemente.
- Diseñar relaciones puedan concatenarse con condiciones de igualdad en los atributos que son parejas de clave primaria y foranea de forma que se garantice la no generación de tuplas falsas.

# Dependencias funcionales

# Dependencias funcionales

## Definición

- La dependencia funcional es una relación que existe cuando un atributo determina de manera única a otro atributo.
- Generalmente relación de clave primaria y otros atributos dentro de una tabla.
- $X \rightarrow Y$
- Ejemplos:
  - $Rut \rightarrow \{\text{Nombre, apellido, edad}\}$
  - $Patente \rightarrow \{\text{Marca, color, año}\}$

# Dependencias funcionales

## Reglas de Dependencias Funcionales

Tabla X

C	B	A	D	E
c1	b1	a1	d1	e1
c2	b1	a2	d2	e1
c1	b2	a3	d1	e1
c2	b2	a4	d2	e1
c3	b3	a5	d1	e1

- ¿Qué tipo de dependencias podemos observar entre los atributos de la Tabla X?
- Como los valores de A son unicos entonces
  - $A \rightarrow B$ ;  $A \rightarrow C$ ,  $A \rightarrow D$ ,  $A \rightarrow E$
  - $A \rightarrow BCDE$  (A es ?)
- $BC \rightarrow ADE$
- $BD \rightarrow ACE$
- Mirar tuplas puede ayudar a aclarar qué atributos son dependientes y cuáles son determinantes.
  - determinantes  $\rightarrow$  dependientes.

# Dependencias funcionales

## Reglas de inferencia

- Los axiomas de Armstrong son un conjunto de reglas de utilizadas para inferir todas las dependencias funcionales en una base de datos relacional. Fueron creadas por William W. Armstrong (matemático, 1974).
- **Dependencia trivial (RI1)**
  - Si  $Y$  es subconjunto de  $X$  entonces  $X \rightarrow Y$
- **Aumentación (RI2)**
  - Si  $X \rightarrow Y \Rightarrow (X \cup Z) \rightarrow (Y \cup Z) \Rightarrow XZ \rightarrow YZ$
  - Cada atributo no clave debe depender completamente del PK.
  - Ejemplo:  $\text{id\_estudiante, curse} \rightarrow \text{Nombre\_e, Ciudad, Provincia, Nota, fecha\_completado}$ 
    - Tabla1 :  $\text{id\_estudiante, curso, nota, fecha\_realizado}$
    - Tabla 2:  $\text{id\_estudiante, nombre\_e, ciudad, provincia.}$
- **Transitividad (RI3)**
  - Si  $X \rightarrow Y$  and  $Y \rightarrow Z \Rightarrow X \rightarrow Z$ 
    - Ejemplo  $\text{id\_estudiante} \rightarrow \text{Nombre\_e, Ciudad, Provincia, Nombre\_programa, id\_programa}$
    - Tabla 1:  $\text{id\_estudiante} \rightarrow \text{Nombre\_e, Ciudad, Provincia, id\_programa}$
    - Tabla 2:  $\text{id\_programa} \rightarrow \text{Nombre\_programa}$

# Dependencias funcionales

## Reglas de inferencia

- Problema:
- Sea  $R(A,B,C,D,E,F)$  y  $DF = \{BD \rightarrow E, CD \rightarrow A, E \rightarrow C, B \rightarrow D\}$ , Cuáles son las claves minimales?
- Observación: Todo atributo que no es determinado por otro es parte de la clave minimal.
- Ni B ni F son determinados por otros atributos.
- $DF = \{BD \rightarrow E (1), CD \rightarrow A (2), E \rightarrow C(3), B \rightarrow D(4)\}$
- $BF \rightarrow BDF(4)$
- $\rightarrow BDEF(1)$
- $\rightarrow BCDEF(3)$
- $\rightarrow ABCDEF(2)$
- $BF \rightarrow BDF(4) \rightarrow BDEF(1) \rightarrow BCDEF(3) \rightarrow ABCDEF(2) \Rightarrow BF$  es clave minimal.
- Como B y F son atributos que deben ser parte de toda clave minimal, BF es la unica clave minima de R.



**Tour completo**

# Organizaciones no Gubernamentales (ONG)

## Enunciado

- La coordinadora de ONG desea implementar un BD de las asociaciones existentes en nuestro país. Para lo cual requiere almacenar información sobre cada ONG, los socios, los proyectos que realizan y los trabajadores que las componen.
- De las ONGs se desea almacenar su ID, denominación, dirección, provincia, region, su tipo (ecologista, social, tecnológica...), así como si está declarada de utilidad pública por el ministerio del interior.
- Cada ONG está formada por socios de los que se requiere conocer su Rut, nombre, dirección, provincia, region, fecha de ingreso, cuota mensual de colaboración y el aporte anual que realizan (derivado de cuota x 12).
- Los trabajadores de las ONGs pueden ser voluntarios o contratados.
- Los contratados cobran un sueldo y ocupan cierto cargo en la ONG. Se desea registrar cuanto pagan de salud y AFP.
- Los voluntarios no reciben sueldo y es necesario registrar su edad, profesión y horas destinadas a la ONG.
- Cada trabajador se identifica por su Rut, nombre y fecha de ingreso.
- Un socio no puede ser trabajador de la ONG.
- Las ONGs realizan proyectos a los que están asignados sus trabajadores. Un trabajador puede involucrarse en varios proyectos de una misma region. De cada proyecto se desea registrar su identificación dentro de la ONG, la region y provincia en la que se ejecuta, así como el objetivo que desarrolla y el número de beneficiados. Un proyecto se compone de subproyectos (que son proyectos).

# Organizaciones no Gubernamentales (ONG)

## Entidades E/R

- ONG
- TRABAJADOR
- VOLUNTARIO
- CONTRATADO
- SOCIO
- PROYECTO
- MIEMBRO

# Organizaciones no Gubernamentales (ONG)

## Relaciones

- En una **ONG** trabaja 1 o más **TRABAJADORes**
  - 1:N
- La **ONG** desarrolla uno o más **PROYECTO**s.
  - 1:N
- La **ONG** esta formada por **SOCIOS**
  - N:M
- Un **TRABAJADOR** es **CONTRATADO**
  - 1:1
- Un **TRABAJADOR** es **VOLUNTARIO**
  - 1:1
- Un **PROYECTO** esta compuesto por Sub**PROYECTO**s.
  - 1:N
- Los **MIEMBROS** son **SOCIOS**
  - 1:1
- Los **MIEMBROS** son **TRABAJADORes**
  - 1:1

# Organizaciones no Gubernamentales (ONG)

## Atributos Entidades E/R

- ONG(ID, denominación, dirección, provincia, región, tipo, utilidad\_publica)
- SOCIO(dirección, provincia, región)
- TRABAJADOR(fecha\_ingreso)
- VOLUNTARIO(edad, profesión, horas)
- CONTRATADO(sueldo, rango, p\_salud, p\_afp)
- PROYECTO(id, región, provincia, obj, numero\_beneficiados)
- MIEMBRO(rut, nombre)

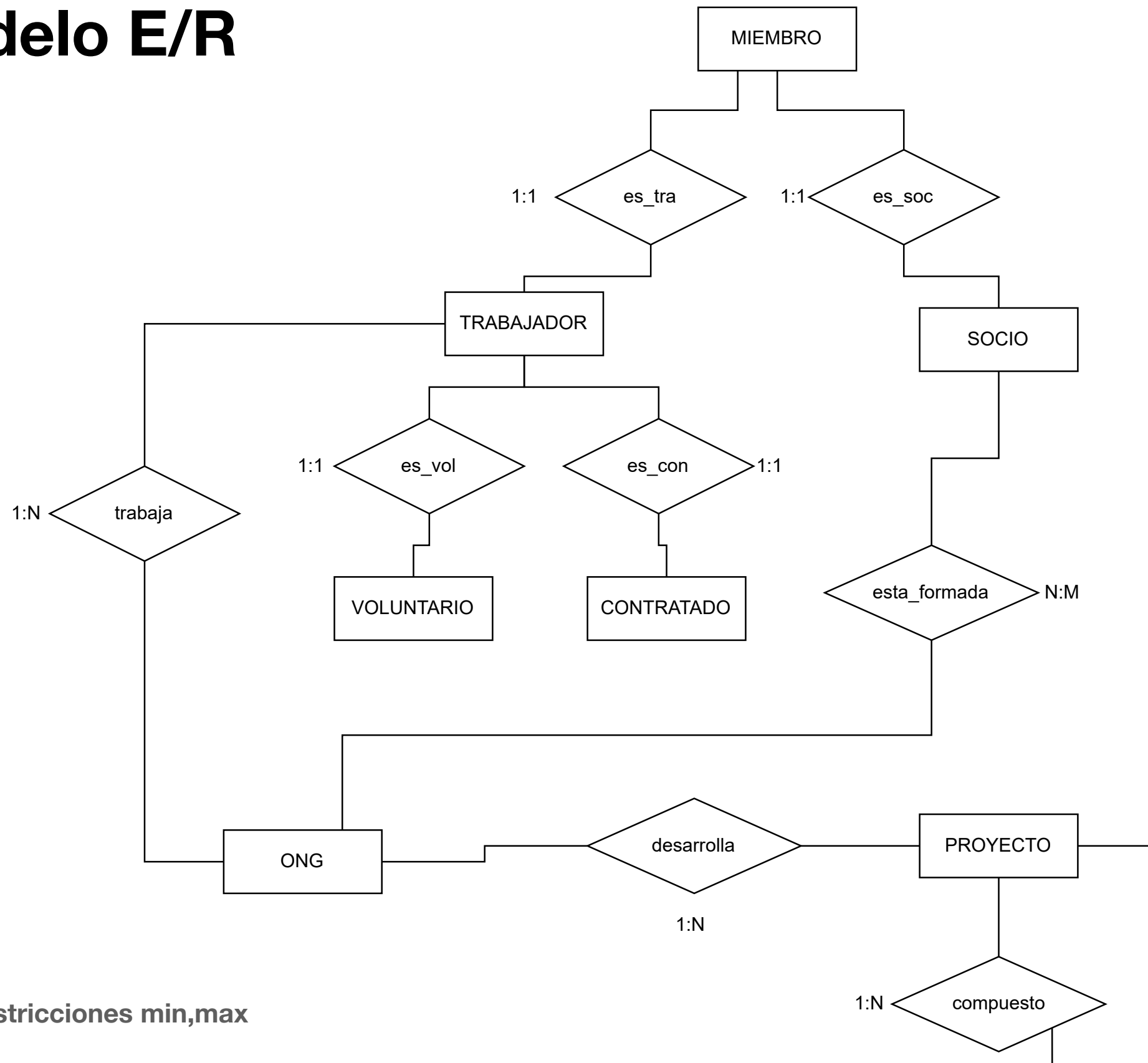
# Organizaciones no Gubernamentales (ONG)

## Atributos en relaciones

- En una **ONG** trabaja 1 o más **TRABAJADORes**
  - 1:N
- La **ONG** desarrolla uno o más **PROYECTO**s.
  - 1:N
- La **ONG** esta formada por **SOCIOS** (*cuota\_mensual, fecha\_ingreso, aporte\_anual*)
  - N:M
- Un **TRABAJADOR** es **CONTRATADO** (*tipo*)
  - 1:1
- Un **TRABAJADOR** es **VOLUNTARIO** (*tipo*)
  - 1:1
- Un **PROYECTO** esta compuesto por Sub**PROYECTO**s.
  - 1:N
- Los **MIEMBROS** son **SOCIOS** (*tipo*)
  - 1:1
- Los **MIEMBROS** son **TRABAJADORes** (*tipo*)
  - 1:1

# Organizaciones no Gubernamentales (ONG)

## Modelo E/R



Complete restricciones min,max

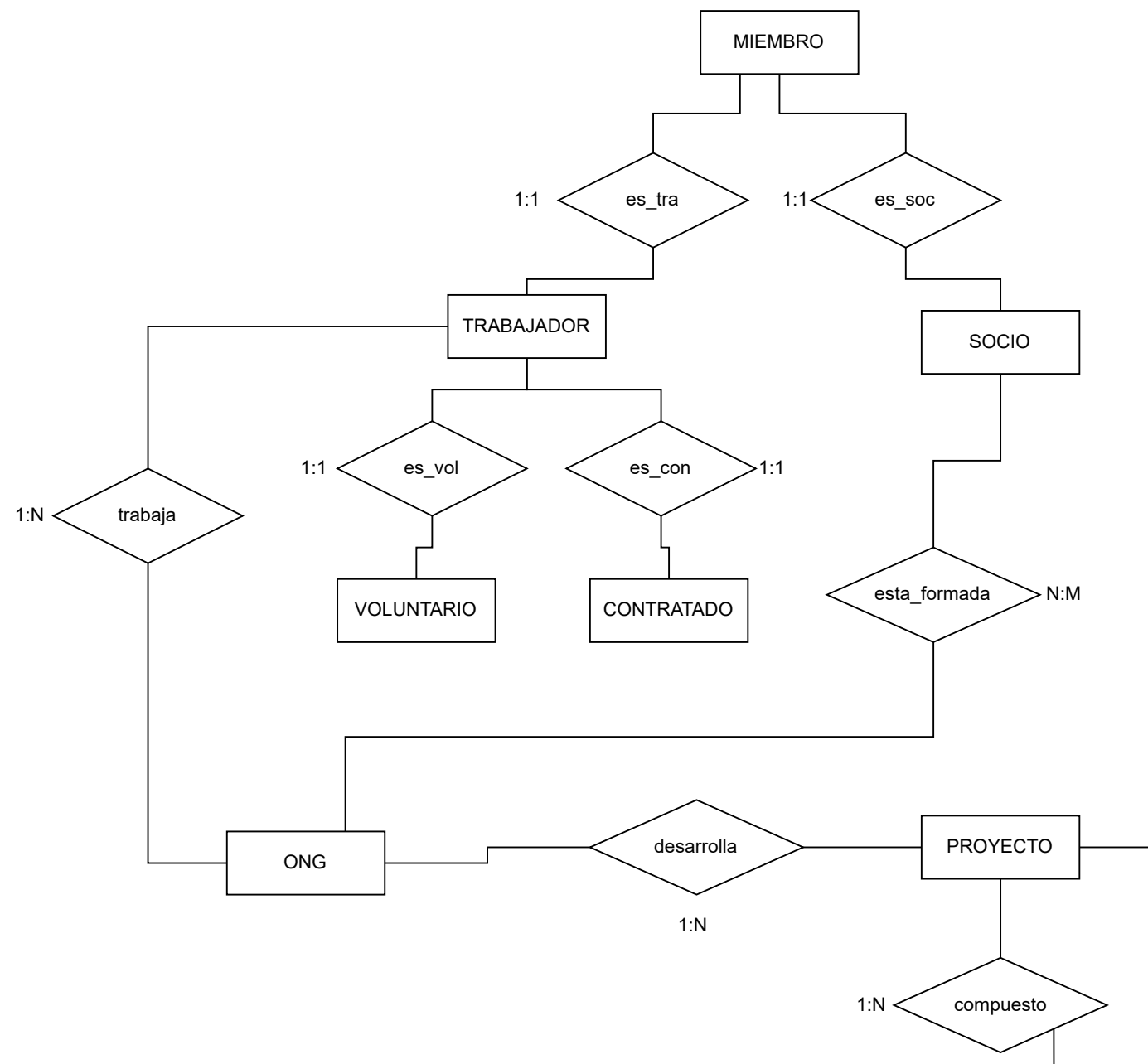
**ER a relacional**



# ONG a relacional

E/R	Relacional
Entidad	Relación
Relación 1:1 o 1:N	Clave foránea
Relación N:M	Genera relación con dos claves foráneas
Relación n-ary	Genera relacion con n claves foraneas
Atributo simple	Atributo
Atributo compuesto	Conjunto de atributos simples
Atributo multivalor	Genera relación y clave foranea
Atributo clave	Clave primaria o secundaria

# ONG a relacional



- MIEMBRO(Rut, nombre)
- SOCIO(dirección, provincia, region)
- TRABAJADOR(fecha\_ingreso)
- VOLUNTARIO(Edad, profesión, horas)
- CONTRATADO(sueldo, rango, p\_salud, p\_afp)

# ONG a relacional

1:1

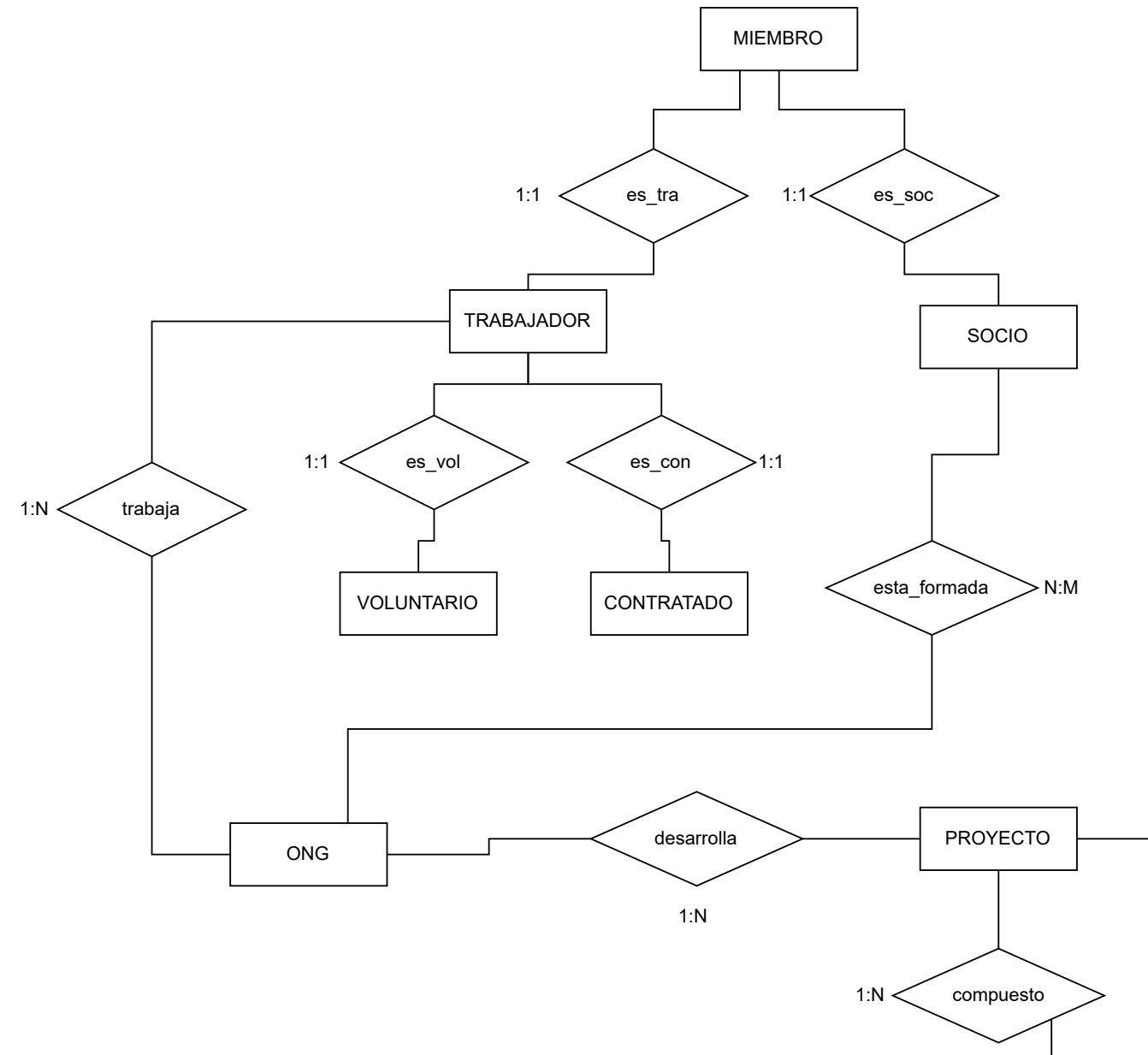
→ **MIEMBRO**(Rut, nombre, Tipo)

→ **SOCIO**(Rut, dirección, provincia, region)

→ **TRABAJADOR**(Rut, fecha\_ingreso, Tipo)

→ **VOLUNTARIO**(Rut, edad, profesión, horas)

→ **CONTRATADO**(Rut, sueldo, rango, p\_salud, p\_afp)



# ONG a relacional

1:N

**PROYECTO**(idp, región, provincia, obj, numero\_beneficiados, *IDO*)

**ONG**(IDO, denominación, dirección, provincia, region, tipo, utilidad\_publica)

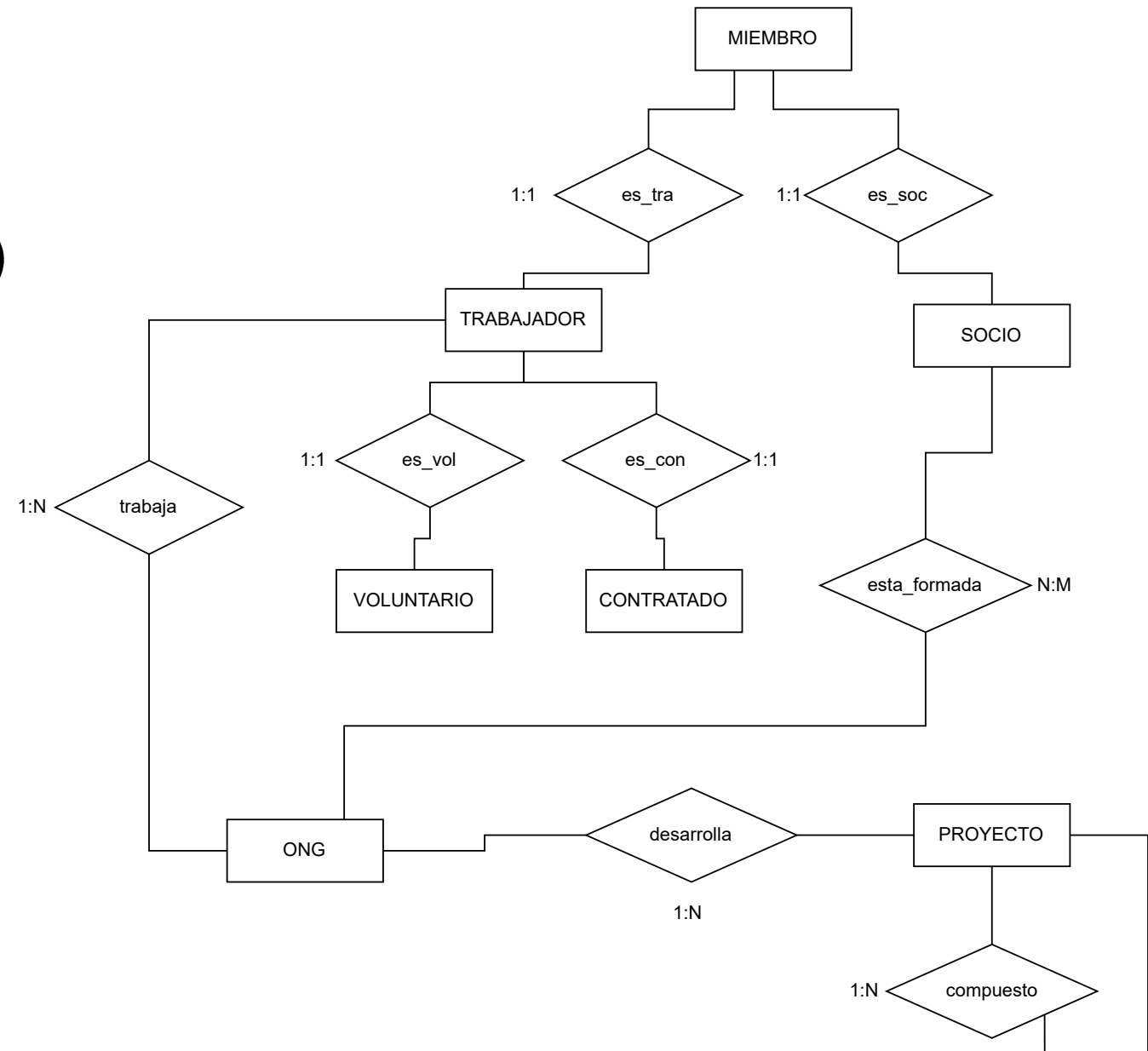
**MIEMBRO**(Rut, nombre, Tipo)

**SOCIO**(Rut, dirección, provincia, región)

**TRABAJADOR**(Rut, fecha\_ingreso, Tipo, *IDO*)

**VOLUNTARIO**(Rut, edad, profesión, horas)

**CONTRATADO**(Rut, sueldo, rango, p\_salud, p\_afp)



# ONG a relacional

## 1:N reflexivo

**PROYECTO**(idp, región,  
provincia,obj,numero\_beneficiados,*IDO,A\_padre*)

**ONG**(IDO, denominación, dirección,  
provincia, region, tipo,utilidad\_publica)

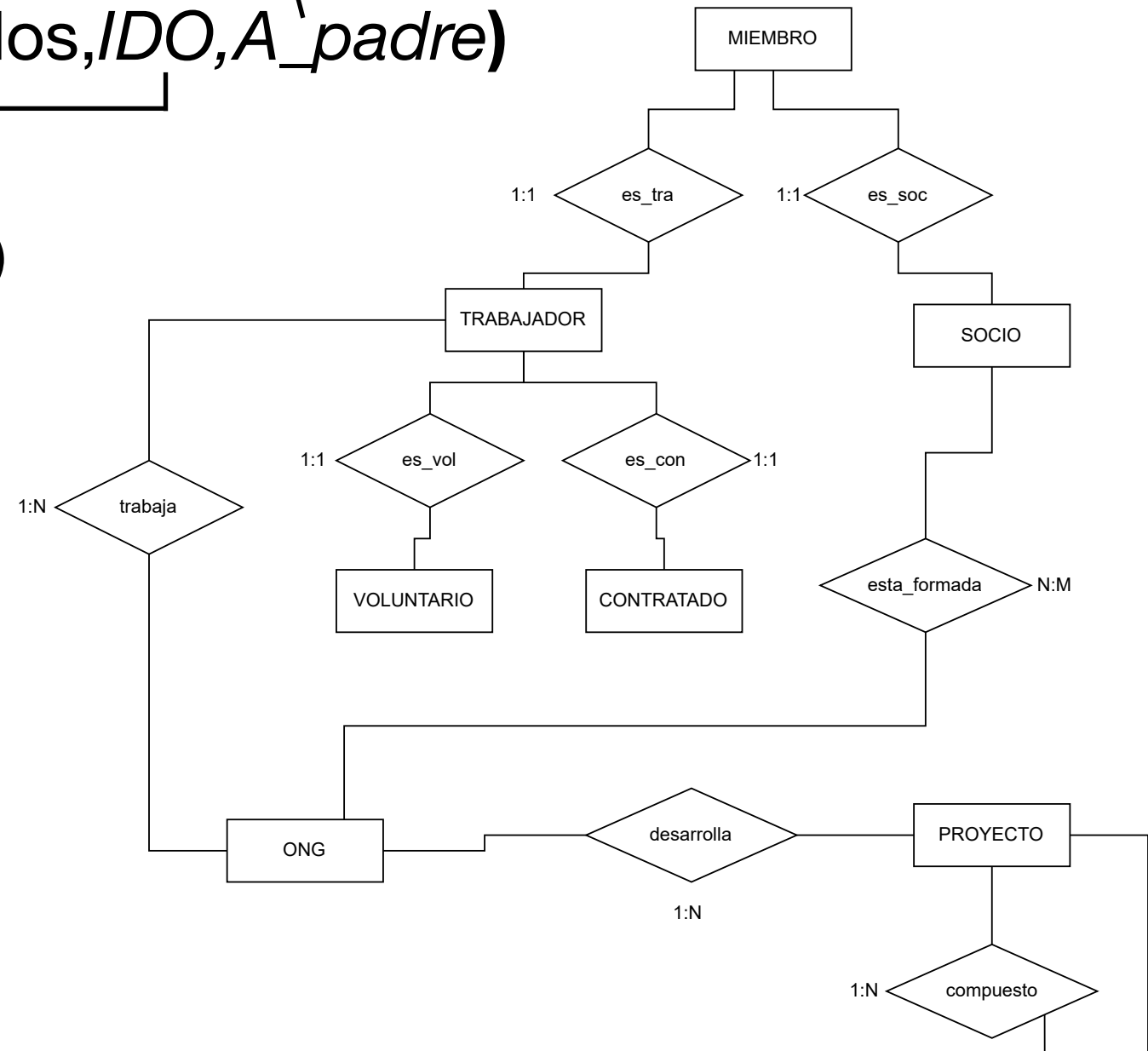
**MIEMBRO**(Rut, nombre, Tipo)

**SOCIO**(Rut,dirección, provincia,  
región)

**TRABAJADOR**(Rut,fecha\_ingreso,Tipo,  
*IDO*)

**VOLUNTARIO**(Rut,Edad, profesión,  
horas)

**CONTRATADO**(Rut,sueldo,  
rango,p\_salud,p\_afp)



# ONG a relacional

N:M

**PROYECTO**(idp, region, provincia, obj, numero\_beneficiados, *IDO*, *A\_padre*)

**ONG**(IDO, denominacion, direccion, provincia, region, tipo, utilidad\_publica)

**MIEMBRO**(Rut, nombre, Tipo)

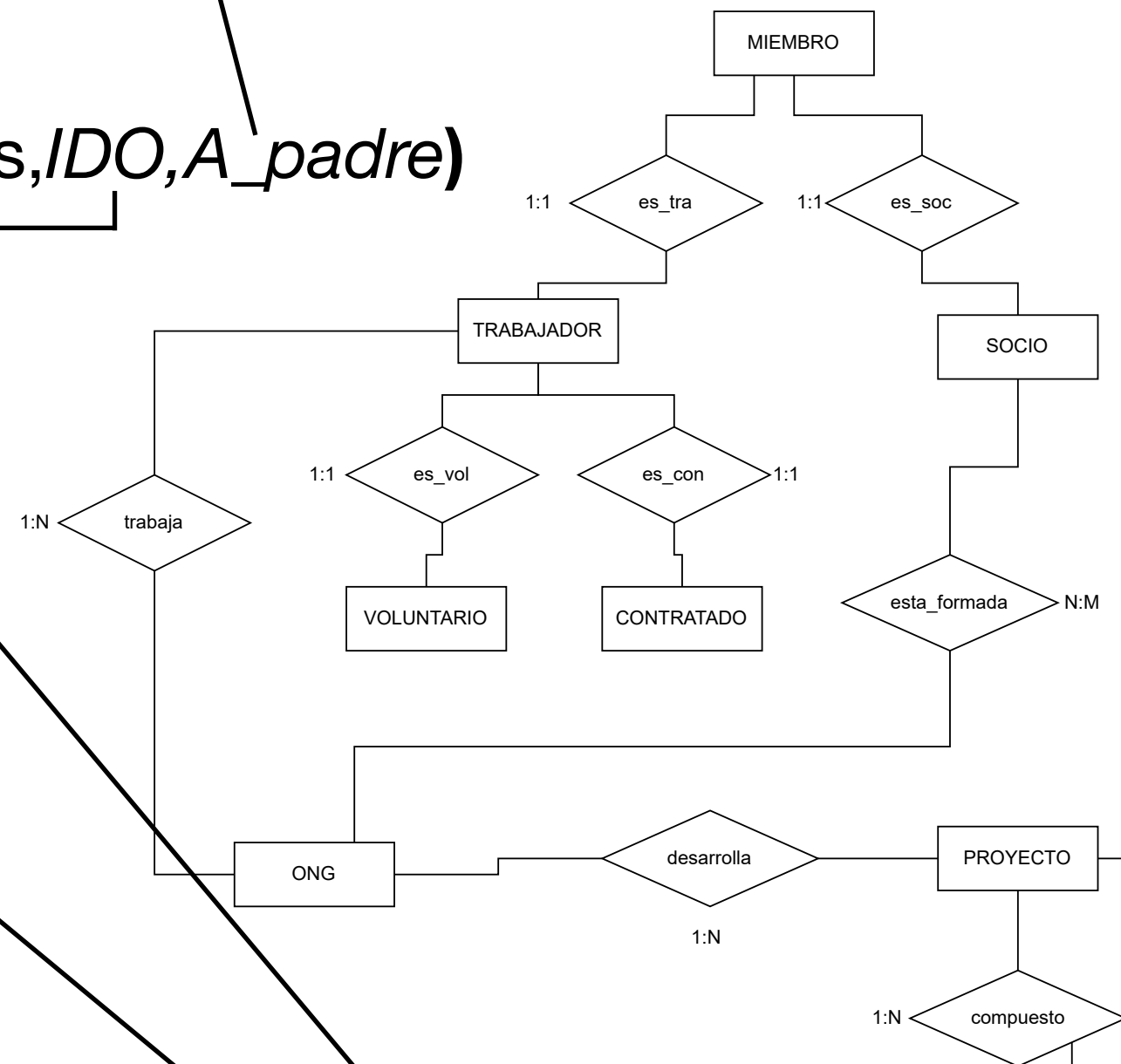
**SOCIO**(Rut, direccion, provincia, region)

**TRABAJADOR**(Rut, fecha\_ingreso, Tipo, *IDO*)

**VOLUNTARIO**(Rut, Edad, profesion, horas)

**CONTRATADO**(Rut, sueldo, rango, p\_salud, p\_afp)

**FORMADA**(Rut, *IDO*, cuota\_mensual, fecha\_ingreso, aporte\_anual)



# Relacional a SQL

# ONG a relacional a SQL

**PROYECTO**(idp, region, provincia, obj, numero\_beneficiados, *IDO*, *A\_padre*)

**ONG**(IDO, denominacion, direccion, provincia, region, tipo, utilidad\_publica)

**MIEMBRO**(Rut, nombre, Tipo)

**SOCIO**(Rut, direccion, provincia, region)

**TRABAJADOR**(Rut, fecha\_ingreso, Tipo, *IDO*)

**VOLUNTARIO**(Rut, Edad, profesion, horas)

**CONTRATADO**(Rut, sueldo, rango, p\_salud, p\_afp)

**FORMADA**(Rut, IDO, cuota\_mensual, fecha\_ingreso, aporte\_anual)

```
CREATE TABLE ONG (  
    IDO INTEGER PRIMARY KEY,  
    denominación INTEGER NOT NULL,  
    dirección TEXT NOT NULL  
    provincia TEXT NOT NULL,  
    región TEXT NOT NULL,  
    tipo TEXT NOT NULL,  
    utilidad_publica INTEGER  
);
```



# GoogleColab

ONG.ipynb - Colaboratory

Take a screenshot on your Mac

colab.research.google.com/drive/1Qs10Da8ExMpuocAYMyf2HP8CbGn4kBo\_#scrollTo=0\_SubMMmWLhV

Apps Mac OS X 10.6 Sn... curso bioperl SVG Crowbar 2 NGS Alignment Pr... SRA-toolkit GAGE PAGIT - Post Asse... Contacto - Cerraj... Microsoft Exchan... MactelSupportTea... mkvds2ac3/REA... SPSmart-SNP

ONG.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

RAM Disk Editing

+ Code + Text

ONG relacional a SQL

[1]

1 # cargamos sqlite3  
2 # https://www.sqlite.org/docs.html  
3 %reload\_ext sql  
4 #nos conectamos o creamos una base de datos  
5 %sql sqlite:///ONG.db

'Connected: @ONG.db'

0s

1 #activamos claves foráneas  
2 #activate foreign key support on SQL lite  
3 %%sql  
4 PRAGMA foreign\_keys;  
5 PRAGMA foreign\_keys = ON;  
6 PRAGMA foreign\_keys;

\* sqlite:///ONG.db  
Done.  
Done.  
Done.  
foreign\_keys  
1

Creamos base de datos ONG

[3]

1 %%sql  
2 drop table if exists PROYECTO;  
3 drop table if exists ONG;  
4 /\*drop table if exists MIEMBRO;  
5 drop table if exists TRABAJADOR;  
6 drop table if exists VOLUNTARIO;  
7 drop table if exists CONTRATADO;  
8 drop table if exists FORMADA;\*/  
9  
10 -- creamos las relaciones --  
11  
12 -- Relación ONG --  
13 CREATE TABLE ONG (  
14 ido INTEGER PRIMARY KEY,  
15 denominacion INTEGER NOT NULL,  
16 direccion TEXT NOT NULL,  
17 provincia TEXT NOT NULL,  
18 region TEXT NOT NULL,  
19 tipo TEXT NOT NULL,

completed at 4:38 AM

https://github.com/adigenova/uohdb/blob/main/code/ONG.ipynb

# Consultas?

Consultas o comentarios?

Muchas gracias