

Reporte Practica 8 Fundamentos de Bases de Datos

Alemán Galicia Diego Leonardo
daleman@ciencias.unam.mx

Ayala Morales Mauricio
ContrarioMotu@ciencias.unam.mx

Díaz Tinoco Gisela Maite
digit@ciencias.unam.mx

Gutiérrez Medina Sebastián Alejandro
sebasguti1511@ciencias.unam.mx

Ramírez Gutiérrez Oscar
rg.oscar17@ciencias.unam.mx

12 de mayo de 2022

Para la población de la base de datos, se utilizó la herramienta *Mockaroo*, la cual nos permite crear nuestros propios schemas y datasets.

Nuestra manera de trabajo implicó el primero crear los datasets de las entidades fuertes, que no dependían de un foreign key, de estas, se crearon 100 elementos de cada una, a excepción de Producto en la cual se crearon 300. Estas entidades son:

- **Supervisor**
- **Veterinario**
- **Estética**
- **Estilista**
- **Dueño**
- **Producto**

Seguido de estas, utilizamos los csv de estas entidades para poder usarlos dentro de las otras tablas, es decir, trabajar con el mismo conjunto de datos.

- **Apartado**

Para este necesitábamos del id de estética, por lo tanto se ocupó el dataset de las estéticas, así cada apartado (de los 100 que generamos) va a estar ligado a una estética mediante su id.

- **Mascota**

Las mascotas están ligadas al CURP de su dueño, por lo que ocupamos el dataset de dueño que se había generado previamente para generar 100 mascotas.

Para el id de la mascota utilizamos un row number que nos permite crear números consecutivos (por lo tanto tendremos ids del 1 al 100), para nombre utilizamos el tipo de dato First name (European) para darle variedad, en fecha de nacimiento utilizamos un tipo date con el rango de Mayo 9 del 2002 a Mayo 9 de 2022, para la especie ocupamos un nombre comun de animal mientras que para la raza utilizamos el nombre científico, y el peso lo manejamos como un numero con dos decimales en el rango de 1 a 90.

■ ConsultaNormal

Aquí necesitabamos el id de la mascota atendida y el curp del veterinario que le atendió, por lo que usamos los datasets de Veterinario y Mascota.

■ ConsultaEmergencia

Aquí necesitabamos el id de la mascota atendida y el curp del veterinario que le atendió, por lo que usamos los datasets de Veterinario y Mascota.

■ Recibo

Usamos el dataset estética ya que debe de estar ligado al id de la estética donde se generó, nombre del dueño del dataset dueño ya que es el que va a pagar el recibo, nombre de la mascota del dataset mascota, ya que es quien recibió un servicio y nombre del trabajador que le atendió lo sacamos del dataset Veterinario. De aquí igual se generan 100 registros.

■ GenerarReciboConsultaNorm

Para idRecibo se trae el id desde el dataset de Recibo, al igual que el id de la estética, el método de pago, el nombre del dueño, el nombre de la mascota y el nombre del trabajador. Del dataset de consulta normal, se trae el id de la consulta normal, el motivo de la consulta, el estado de salud de la mascota y la fecha de la siguiente consulta.

■ GenerarReciboConsultaEm

Para idRecibo se trae el id desde el dataset de Recibo, al igual que el id de la estética, el método de pago, el nombre del dueño, el nombre de la mascota y el nombre del trabajador. Del dataset de consulta de emergencia se trae el id de la consulta de emergencia, el procedimiento y el código asociado a la emergencia. El costo igual viene de recibo pero lo representamos como un número con dos decimales, que va de 1 a 3000.

■ Síntoma

Como las consultas de emergencia tienen un id del 1 al 100, ponemos un número aleatorio entre 1 y 100 para el id.

■ Informar

Para este necesitamos traer del dataset de estética el id, su nombre, dirección que contiene: nombre de la calle, número, estado y código postal, el teléfono de la estética, su horario de inicio y cierre y el número de consultorios. Del dataset supervisor tomamos su CURP, nombre, apellido materno y paterno, fecha de nacimiento, el número de teléfono del supervisor, su género, salario, rfc y su dirección: calle, número, estado y código postal.

■ **Alimento**

Aunque alimento tiene como fk a idProducto, lo que hacemos, ya que generamos 300 productos al inicio, es indicar estos tales que los productos con id entre 201 a 300, son alimentos, por lo que tomamos un row number (que nos generará ids del 1 al 100) y le sumamos 200.

■ **Medicamento**

De una manera algo parecida a medicamento, solo que estos row numbers (del 1 al 100) sólo le sumamos 100, así tendrán un id del 101 al 200 y estos serán los medicamentos.

■ **OfertarProducto**

Aunque tiene como fk idProducto e id de estética, sabemos que los id de las estéticas van de 1 a 100, por lo que elegimos el tipo number que nos dará un número al azar, sin decimales, en el intervalo 1 a 100, igualmente, con idProducto, debido a la construcción que le dimos a los medicamentos y alimentos, sabemos que los productos con id del 1 al 100, son productos sin alguna especialización (como serían medicamento o alimento).

■ **OfertarMedicamento**

Igualmente, estética lo tomamos como un número del 1 al 100, sin decimales y idAlimento, por construcción de Medicamento, sabemos que estos productos tienen ids que van del 101 al 200.

■ **OfertarAlimento**

Igualmente, estética lo tomamos como un número del 1 al 100, sin decimales, y idAlimento, por construcción de Alimento, sabemos que estos productos tienen ids que van del 201 al 300.

■ **Servicio**

Del dataset de recibo traemos el id y para el servicio elegimos un tipo sentencia de sólo una palabra.

■ **MetodoPago**

Del dataset de dueño traemos su curp y su nombre (este se utilizará para potencialmente ligarlo como titular de la tarjeta en caso de que se vaya a pagar con tarjeta), esEfectivo y esTarjeta lo representamos como expresiones regulares que nos arrojan 1 o 0, sólo que en esEfectivo ponemos una condicional ya que si es tarjeta es 1 (o sea que el pago va a ser en tarjeta), automáticamente esEfectivo va a ser 0, ya que no se pueden tener ambos metodos de pago, en cualquier otro caso, es 1 ya que se debe de pagar de forzosamente una forma. para número de tarjeta se usó una secuencia de 16 dígitos, e igual utilizamos una función tal que si esTarjeta es igual a 0, numero de tarjeta es igual a null, esto porque si no hay tarjeta, se paga con efectivo, por lo tanto no necesitamos información de una tarjeta que no va a ser utilizada.

■ **Recetar**

Del dataset de consultaNormal, traemos el id de la consulta y para el medicamento, traemos del dataset de Medicamento uno de los ids.

Procedimiento para poblar tabla sin FK

1. Se creo un esquema con los atributos y sus respectivas restricciones de la tabla que se definieron en el ddl.

The screenshot shows the 'duenio' schema configuration interface. It lists fields with their names, types, and options. The fields are:

- curp**: Character Sequence, Options: #####-####-###, blank: 0%, [Σ] [X]
- fechanacimiento**: Datetime, Options: 05/08/1920 to 05/09/2004, format: yyyy-mm-dd, blank: 0%, [Σ] [X]
- telefono**: Character Sequence, Options: #####, blank: 0%, [Σ] [X]
- calle**: Street Name, Options: blank, 0%, [Σ] [X]
- numcalle**: Number, Options: min: 1, max: 1000, decimals: 0, blank: 0%, [Σ] [X]
- estado**: Custom List, Options: Aguascalientes, Baja California, Baja California Sur, Campeche, Chiapas, Chihuahua, Coahuila, Colima, Ciudad de Mexico, Durango, Guanajuato, random, blank: 0%, [Σ] [X]
- codigopostal**: Character Sequence, Options: ####, blank: 0%, [Σ] [X]
- nombre**: First Name, Options: blank, 0%, [Σ] [X]
- apellidopaterno**: Last Name, Options: blank, 0%, [Σ] [X]
- apellidomaterno**: Last Name, Options: blank, 0%, [Σ] [X]
- email**: Email Address, Options: blank, 0%, [Σ] [X]

2. Se generan 100 tuplas con sus correspondientes valores y se pre visualizan, verificando que los datos tengan sentido. Ademas se crea el dataset, utilizando la opcion *Create Dataset*, para poder referenciarlo en otras tablas y pasarlo a formato SQL.

The screenshot shows the data preview interface for the 'duenio' schema. It displays a table with 10 columns: curp, fechanacimiento, telefono, calle, numcalle, estado, codigopostal, nombre, apellidopaterno, and apellidomaterno. The table contains 100 rows of generated data. A 'Create Dataset' dialog box is visible, prompting the user to generate data and store it as a new dataset for reference in other schemas. The dialog box includes a 'Generate Data' button and a 'Download Data' button.

curp	fechanacimiento	telefono	calle	numcalle	estado	codigopostal	nombre	apellidopaterno	apellidomaterno
BHUM531978SEHPDOR0	1980-01-14	1183727644	Anderson	254	Tlaxcala	03478	Aurore	Winfindale	Radden
YLGJ106995KFPKCD0	2003-03-06	7695477415	Buell	503	Nuevo Leon	54909	Charlotte	Haug	August
UBUE952010SNHKG02	1973-02-09	0244530283	Rockefeller	303	Chiapas	23707	Wina	Cowderay	felip
DBJX805334LND0BMX5	2000-05-29	8902944166	Cambridge	97	Tabasco	12968	Rolfe	Staniland	Whaites
UTFI916766VAISAZF5	1990-09-04	6428382939	Porter	240	Campeche	99084	Rikki	Cawthorn	Saines
KDXS895635COBLATS2	1925-05-15	2808272454	Surrey	385	Queretaro	68496	Geoff	Gamet	Behnecken
TNUZ177694Z5JVASY7	1957-06-25	9627594875	Bluestem	301	Estado de Mexico	88194	Clare	Blague	Mucci
JMBA087226HVQGNKP6	1962-07-09	3757197087	Bunker Hill	455	Baja California Sur	59529	Phyllis	Gaskins	McFadyen
GXP006831FQYFIJ00	1936-06-18	8809983313	Kropf	368	Baja California Sur	68759	Pincus	Fishley	Kort
EJWC855236HRWRGKX7	1969-12-09	9896095513	Manufacturers	90	Tabasco	92993	Eleanore	Asplen	Booler
RFPZ459035FVGIKRX4	1924-09-12	6014259541	Monica	61	Hidalgo	75055	Lexine	Jerome	Hands
EBEA059467ANI8CBA8	1920-12-14	9703512970	O'Neill	346	Sinaloa	54021	Mack	Gillingham	Wellan
MECK893746Z0FXWM7	1961-08-26	2165206681	Village	902	Michoacan	43118	Che	O'Mailey	Millins
OGOP859539TC00EEN7	1928-11-08	7986415504	Laurel	152	Coahuila	58521	Izak		
OTAR6856811OZVNP57	1999-12-03	9757092514	Starling	673	Baja California	87118	Roxi		
THRY633656KAJHFTZ3	1928-03-22	1439910701	Westridge	172	Ciudad de Mexico	61019	Rhanna		

- Se carga el dataset csv en la herramienta *CSV To SQL Converter* y modificamos los tipos para que correspondan a los tipos del ddl, de la misma forma modificamos el nombre de la tabla. Le damos a convertir y copiamos el código SQL generado.

☐ Keep original field names as column names

NOTE - you can change the column names below by overwriting the Field Name value.

☐ Enclose field names below with ☒ Quotes ("name") ☐ Backtick (`name`) ☐ Brackets ([name])

Col #	Field Name	Data Type	Max Size	# Dec	Key	Include	Required	Trim	Upper	Lower	Use NULL for Empty Field	Template ({f}=field) Ex: {f}+100
1	curp	VarChar	18		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
2	fechanacimiento	Date	10		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
3	telefono	VarChar	10		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
4	calle	VarChar	13		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
5	numcalle	Integer	3		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
6	estado	VarChar	19		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
7	codigopostal	VarChar	5		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
8	nombre	VarChar	9		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
9	apellidopaterno	VarChar	11		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
10	apellidomaterno	VarChar	12		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
11	email	VarChar	32		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Schema.Table or View Name:

☐ Drop Table/View ☐ IF EXISTS

☐ Create Table/View ☐ IF NOT EXISTS

Include Auto Increment Clause on single numeric Primary Key using

For INSERT

Procedimiento para poblar tabla con FK

- Hacemos el esquema correspondiente de la tabla, definiendo los atributos llaves foraneas como *Dataset column* y asociando el dataset correspondiente y el atributo al que se hace referencia.

mascota

Field Name	Type	Options
idMascota	Row Number	blank: 0 % <input type="button" value="Σ"/> <input type="button" value="X"/>
curp	Dataset Column	duenio curp random blank: 0 % <input type="button" value="Σ"/> <input type="button" value="X"/>
nombre	First Name (European)	blank: 0 % <input type="button" value="Σ"/> <input type="button" value="X"/>
fechanacimiento	Datetime	05/09/2002 to 05/09/2022 format: yyyy-mm-dd blank: 0 % <input type="button" value="Σ"/> <input type="button" value="X"/>
especie	Animal Common Name	blank: 0 % <input type="button" value="Σ"/> <input type="button" value="X"/>
raza	Animal Scientific Name	blank: 0 % <input type="button" value="Σ"/> <input type="button" value="X"/>
peso	Number	min: 1 max: 90 decimals: 2 blank: 0 % <input type="button" value="Σ"/> <input type="button" value="X"/>

- Hacemos los demás pasos de manera similar al caso anterior.