

Reporte Practica 07

Para la generación de datos para poblar nuestra base de datos utilizamos Mockaroo, que es una herramienta en línea que permite generar datos de prueba de manera rápida y sencilla.

A continuación explicaremos como creamos los datos prueba en Mockaroo para nuestra base de datos:

Tabla Juez:

Decidimos poblar la tabla juez, ya que se nos pedía crear al menos dos tablas con llave primaria y esta tabla lo cumplía.

Para los atributos de la tabla, utilizamos tipos que ya estaban predefinidos en mockaroo.

Y para la fecha de nacimiento buscamos jueces con gran experiencia, por lo que su rango de edad sera de 59 a 44 años

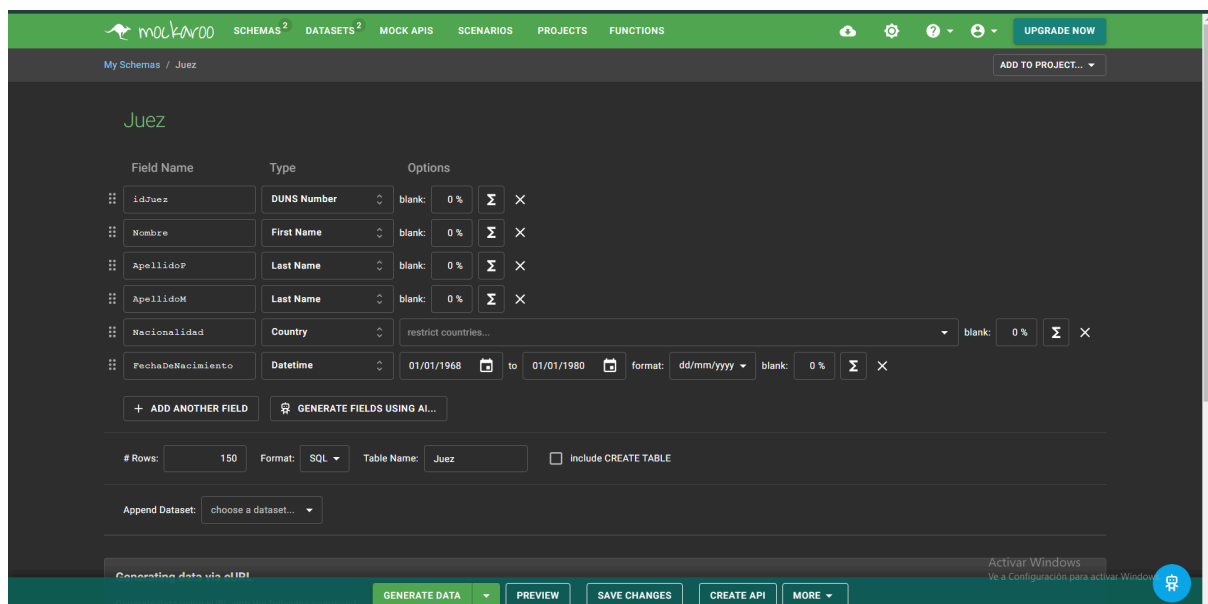


Tabla Atleta

Decidimos poblar la tabla Atleta, que al igual que la tabla juez, cuenta con llave primaria

Para los atributos de la tabla, utilizamos tipos que ya estaban predefinidos en mockaroo, excepto para “Disciplina”, para este investigamos todas las disciplinas que hay en los juegos olímpicos y creamos nuestro propio tipo de valores.

Además para la edad los atletas no pueden ser menores de edad.

Y para el atributo año debe estar en la fecha en la que se desarrollaron los juegos olímpicos

The screenshot shows the Mockaroo website interface for configuring a dataset. The top navigation bar includes links for SCHEMAS, DATASETS, MOCK APIS, SCENARIOS, PROJECTS, and FUNCTIONS. A green banner at the top contains the Mockaroo logo and navigation links. Below the banner, a message states: "Need some mock data to test your app? Mockaroo lets you generate up to 1,000 rows of realistic test data in CSV, JSON, SQL, and Excel formats. Need more data? Plans start at just \$60/year. Mockaroo is also available as a docker image that you can deploy in your own private cloud."

The main configuration area is a table with columns: Field Name, Type, and Options. The fields are configured as follows:

Field Name	Type	Options
idAtleta	MongoDB ObjectID	blank: 0 %
NombrePaís	Country	restrict countries... blank: 0 %
Disciplina	Custom List	natación, maratón, saltos de trampolín, waterpolo, natación artística, tiro con arco, atletismo, bádminton random blank: 0 %
Genero	Gender (Binary)	blank: 0 %
Nombre	First Name	blank: 0 %
ApellidoP	Last Name	blank: 0 %
ApellidoM	Last Name	blank: 0 %
FechaDeNacimiento	Datetime	01/01/1995 to 01/12/2005 format: dd/mm/yyyy blank: 0 %
Año	Datetime	07/26/2024 to 08/11/2024 format: dd-mm-yyyy blank: 0 %

Below the table, there are buttons for "+ ADD ANOTHER FIELD" and "GENERATE FIELDS USING AI...". At the bottom, the configuration is summarized: # Rows: 150, Format: SQL, Table Name: Atleta, and a checkbox for "include CREATE TABLE". The bottom bar contains buttons for "GENERATE DATA", "PREVIEW", "SAVE AS...", "DERIVE FROM EXAMPLE...", and "MORE".

Tabla Competencia:

Poblamos la tabla competencia porque contiene información sobre diferentes competencias en las que pueden participar atletas. Y para vincular estos dos datos tenemos el atributo idAtleta, que funciona como llave foránea.

Para esta vinculación en Mockaroo guardamos la tabla Atleta como Dataset para luego poder hacer referencia en la tabla Competencia

Para los demás atributos idCompetencia y FaseEliminatoria utilizamos datos ya predefinidos.

The screenshot shows the Mockaroo 'New Schema' interface. The top navigation bar includes links for SCHEMAS², DATASETS², MOCK APIS, SCENARIOS, PROJECTS, and FUNCTIONS. The main area is titled 'New Schema' and contains a table with the following fields:

Field Name	Type	Options
idCompetencia	GUID	blank: 0 % Σ ×
idAtleta	Dataset Column	Atleta idAtleta random blank: 0 % Σ ×
FaseEliminatoria	Boolean	blank: 0 % Σ ×

Below the table, there are buttons for '+ ADD ANOTHER FIELD' and 'GENERATE FIELDS USING AI...'. At the bottom of the form, there are settings for '# Rows: 150', 'Format: SQL', 'Table Name: Competencia', and a checkbox for 'Include CREATE TABLE'. An 'Append Dataset:' dropdown menu is also present, set to 'choose a dataset...'. The bottom of the interface features a green bar with buttons for 'GENERATE DATA', 'PREVIEW', 'SAVE AS...', 'DERIVE FROM EXAMPLE...', and 'MORE'. A Windows watermark is visible in the bottom right corner.

Tabla TelefonoJuez:

Poblamos la tabla TelefonoJuez, ya que tiene una llave compuesta porque utiliza dos columnas, idJuez y Telefono, para definir de manera única cada registro en la tabla.

Para hacer referencia del atributo idJuez de nuestra tabla Juez, la guardamos como Dataset, y después seleccionamos el atributo como tipo de valor.

Para el atributo telefono utilizamos el dato ya predefinido de Mockaroo.

The screenshot shows the Mockaroo web interface for configuring a table named 'TelefonoJuez'. The interface is in Spanish and includes a navigation bar with options like SCHEMAS, DATASETS, MOCK APIS, SCENARIOS, PROJECTS, and FUNCTIONS. The main area displays the table configuration with two fields: 'idJuez' and 'Telefono'. The 'idJuez' field is configured as a 'Dataset Column' with the value 'Juez' and the dataset 'idJuez'. The 'Telefono' field is configured as a 'Phone' with a format of '+## ### ####' and a blank percentage of 0%. The table is set to have 150 rows, in SQL format, with the table name 'TelefonoJuez'. There is an option to 'Include CREATE TABLE' which is currently unchecked. Below the configuration, there is a section for 'Generating data via cURL' with a command to generate data using the Mockaroo API. The bottom of the interface has buttons for 'GENERATE DATA', 'PREVIEW', 'SAVE CHANGES', 'CREATE API', and 'MORE'. The Windows taskbar is visible at the bottom, showing the date and time as 01:44 p.m. on 28/10/2024.

Field Name	Type	Options
idJuez	Dataset Column	Juez, idJuez, random, blank: 0%
Telefono	Phone	format: +## ### ####, blank: 0%

Rows: 150 Format: SQL Table Name: TelefonoJuez Include CREATE TABLE

Append Dataset: choose a dataset...

Generating data via cURL

Generate data using cURL with the following command:

```
curl "https://api.mockaroo.com/api/a9c5ac50?count=150&key=c51caf90" > "TelefonoJuez.sql"
```

You can also generate data via the API.

Activar Windows
Ve a Configuración para activar Windows

GENERATE DATA PREVIEW SAVE CHANGES CREATE API MORE