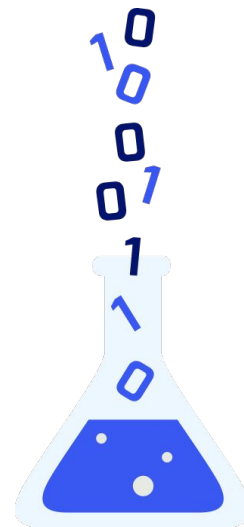


Introducción a Bases de Datos y SQL

Módulo 1 - Desafío

¿Qué son los desafíos?

- Son **ejercicios complementarios** a los realizados en clase.
- Permiten practicar y **consolidar los conocimientos** adquiridos.
- Es recomendable que los realices antes de rendir el examen de la clase.



Ejercicio 1

1. Crear una base de datos con el nombre ***BONUS_TRACK***.
2. Poner en uso la base de datos generada en el paso anterior.
3. Crear una **tabla** con el nombre **“AGENDA”** dentro de la base de datos, con la estructura que se muestra en la [siguiente diapositiva](#).
4. Comentar el **código SQL** generado para la creación de la tabla “AGENDA”.
5. Visualizar las **tablas existentes** en la base de datos para verificar la creación de la tabla “AGENDA”.
6. Visualizar la **estructura** de la tabla “AGENDA”.
7. Agregar el **campo MAIL** a la tabla. Este campo deberá contener como **máximo 50 caracteres** y su **carga** es **obligatoria**.
8. Insertar **4 registros** en la tabla “AGENDA”.

Referencia - Estructura de la tabla a generar:

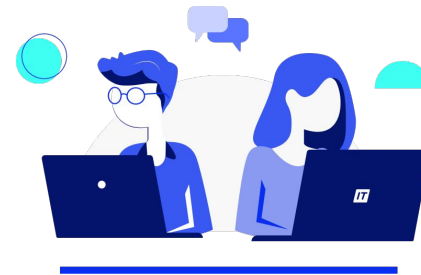
AGENDA		
ID Contacto	integer	PK
Nombre	varchar(20)	NOT NULL
Apellido	varchar(15)	NOT NULL
Domicilio	varchar(50)	
Teléfono	integer	NOT NULL

Ejercicio 2

1. Crear una base de datos con el nombre **“LIBRERÍA”**. Poner en uso la base de datos generada. La misma estará conformada por **8 tablas**, con la estructura que se muestra en la [diapositiva siguiente](#).
2. Una vez creadas todas las tablas, insertar un **conjunto de registros** en cada una de ellas, utilizando el archivo con el nombre **“SCRIPT”**. Este archivo está disponible en la sección *Descargas*.

Consideraciones:

para llevar a cabo este ejercicio, abrir el archivo **“SCRIPT”** provisto, copiar y pegar el código completo en **MySQL Workbench** y ejecutarlo. Una vez ejecutado el código, se debe borrarlo.



Referencia - Estructura de las 8 tablas a generar
(continúa en las siguientes diapositivas):

autores

- autor_id VARCHAR(11)
- apellido VARCHAR(40)
- nombre VARCHAR(20)
- telefono VARCHAR(12)
- direccion VARCHAR(40)
- ciudad VARCHAR(20)
- provincia CHAR(2)
- c_postal CHAR(5)
- estado TINYINT(1)

ventas

- local_id CHAR(4)
- factura_nro VARCHAR(20)
- fecha DATETIME
- cantidad SMALLINT
- forma_pago VARCHAR(12)
- libro_id VARCHAR(6)

locales

- local_id CHAR(4)
- nombre VARCHAR(40)
- direccion VARCHAR(40)
- ciudad VARCHAR(20)
- provincia CHAR(2)
- c_postal CHAR(5)

Referencia - Estructura de las 8 tablas a generar (continuación):

editoriales

- editorial_id CHAR(4)
- nombre VARCHAR(40)
- ciudad VARCHAR(20)
- provincia CHAR(2)
- pais VARCHAR(30)

libros

- libro_id VARCHAR(6)
- titulo VARCHAR(80)
- categoria CHAR(12)
- editorial_id CHAR(4)
- precio DOUBLE
- comentarios VARCHAR(200)
- fecha_publicacion DATETIME

libroautor

- autor_id VARCHAR(11)
- libro_id VARCHAR(6)

Referencia - Estructura de las 8 tablas a generar (continuación):

empleados
<ul style="list-style-type: none">• empleado_id CHAR(9)• nombre VARCHAR(20)• apellido VARCHAR(30)• puesto_id SMALLINT• editorial_id CHAR(4)• fecha_ingreso DATETIME

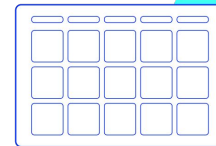
puestos
<ul style="list-style-type: none">• puesto_id SMALLINT• descripcion VARCHAR(50)



Ejercicio 3

1. A partir de lo realizado en el Ejercicio 2, luego de insertar los registros, definir las **claves primarias (Primary Key) en cada tabla**. Tener en cuenta que cada tabla tiene un campo con el sufijo **_id** que identifica al **campo clave**.
2. Examinar los datos ingresados, e **insertar 3 registros** en cada una de las tablas utilizando la sentencia **INSERT** del conjunto de sentencias **DML (Data Manipulation Language)** del lenguaje SQL.

Consideraciones: La lógica del negocio indica que un **mismo número de factura se puede repetir en varios locales y en un mismo local**, pero con **distintos códigos de libros** y la combinación de *autor/libro* es única.

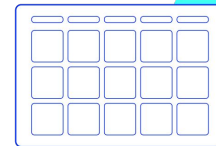


Ejercicio 3

1. A partir de lo realizado en el Ejercicio 2, luego de insertar los registros, definir las **claves primarias (*Primary Key*) en cada tabla**. Tener en cuenta que cada tabla tiene un campo con el sufijo ***_id*** que identifica al **campo clave**.
2. Examinar los datos ingresados, e **insertar 3 registros** en cada una de las tablas utilizando la sentencia **INSERT** del conjunto de sentencias **DML (*Data Manipulation Language*)** del lenguaje SQL.

Consideraciones:

la lógica del negocio indica que un **mismo número de factura se puede repetir en varios locales y en un mismo local**, pero con **distintos códigos de libros** y la combinación de *autor/libro* es única.



En la sección de **Descargas** encontrarás los recursos necesarios para realizar los ejercicios y su resolución para que verifiques cómo te fue.



¡Terminaste el módulo!
Todo listo para rendir el examen