

LICORERIA REQUEJO

OBJETIVOS DEL PROYECTO

La finalidad del proyecto es la administración, gestión del personal y la clientela en su total servicio.

Se gestiona la base de datos para realizar un sistema de gestión en general del negocio.

Se puede realizar consultas, cambios y gestiones sobre los proveedores, clientes, productos que conforman la licorería.

DIAGRAMA DE ENTIDAD RELACIÓN - PDF

Diagrama de Entidad - Relación del proyecto:

[DER-SQL.png](#)

Listado de tablas con descripción de su estructura:

[Licoreria.pdf](#)

CREACIÓN DE LA BASE DE DATOS

Creación de la Base de Datos.

En este caso trataremos acerca de una Licorería:

[SCRIPTBD+REQUEJO.SQL](#)

INSERCIÓN DE DATOS

Inserción de datos de la mano de la herramienta llamada MOCKAROO.

Nos permite insertar datos de forma aleatoria:

[ScriptInsertarDatos+Requejo.sql](#)

STORED, FUNCIONES TRIGGERS

Store Procedure:

[ScriptStoreProcedure+Requejo.sql](#)

Funciones:

[ScriptFunciones+Requejo.sql](#)

Triggers:

[ScriptTrigger+Requejo.sql](#)

SENTENCIAS Y TCL

Creaciones de usuarios dando permisos de Lectura, Inserción y Modificación:

[ScriptSentencias+Requejo.sql](#)

Ejecución de algunos comandos breves:

[ScriptTCL+Requejo.sql](#)

BACKUP

Generamos una copia de seguridad de toda la Base de Datos de Licorería, tomando en cuenta que también se añadiran las Funciones, Store Procedure, Triggers y Vistas:

[ScriptBackup+Requejo.sql](#)

HERRAMIENTAS

Utilizamos como gestor de Base de Datos (MySQL Workbench).

Generador de Datos Aleatorios:

<https://www.mockaroo.com/>

LICORERIA REQUEJO

STORE PROCEDURE, FUNCIONES, TRIGGERS, VISTAS

STORE PROCEDURES

- `sp_insertardatos` = Nos permitira Insertar Datos en la tabla Cliente.
- `sp_orden_cliente` = Nos permitira ordenar(ascendente o descendente) la tabla según el parametro colocado.
- `sp_eliminar_registro` = Nos permitira eliminar un registro(`id_cliente`).

FUNCIONES

- `numero_letra` = Me mostrara la cantidad nombres que conllevan dicha letra seleccionada.
- `o_datos` = nos permite concatenar diferentes tablas(`SELECT,INTO,FROM,WHERE`).

TRIGGERS

- `registrar_Acciones` = Trigger para visualizar el registro de un cliente(`accion,fecha`) - Se utilizo AFTER.
- `tr_actualiza_datos` = Trigger para Actualizar Datos de un Tabla - Se utilizo BEFORE.

VISTAS

- `vista_cliente` = VISTA CLIENTE QUE NOS PERMITE VISUALIZAR NOMBRE,DISTRITO,EDAD ESPECIFICANDO SOLO EL DISTRITO.
- `orden_cliente` = VISTA CLIENTE QUE NOS PERMITE IDENTIFICAR A LOS CLIENTES EN ORDEN ASCENDENTE MEDIANTE SU APELLIDO.
- `listado_precios` = VISTA PRODUCTO QUE NOS MUESTRA LOS PRECIOS SUPERIORES A 300.
- `orden_envio` = VISTA ENVIO QUE NOS PERMITE IDENTIFICAR EL ENVIO EN ORDEN DESCENDENTE MEDIANTE LA CIUDAD DE ENVIO.
- `vista_marca` = VISTA MARCA QUE NOS MUESTRA LOS DATOS DE LA TABLA.
- `lista_pedido` = VISTA PEDIDO QUE NOS MUESTRA EL MONTO TOTAL DE LOS PEDIDOS MENORES A 60.

NOTA: EN ESTE LINK
PODREMOS
VISUALIZAR TODOS
LOS ARCHIVOS
SOLICITADOS PARA EL
PROYECTO FINAL:
[ARCHIVOS COMPLETOS](#)