**UNIVERSIDAD CÉSAR VALLEJO**

**FACULTAD DE INGENIERÍA**

**Escuela de Ingeniería de Sistemas**

**Práctica MARKETPERU - Gestión de Datos S11**

**AUTOR**

Quispe Olano, Marlon Yber ([0000-0002-3668-207X](https://orcid.org/0000-0002-3668-207X))

**DOCENTE**

Saavedra Jimenez, Robert Roy

**SECCIÓN**

**A1**

**PERÚ - 2023**

**INTRODUCCIÓN**

El presente informe tiene como propósito detallar y analizar una base de datos en SQL Server, así como realizar consultas utilizando el lenguaje de consulta estructurado (SQL) y documentar cada paso a través de capturas y explicaciones de los códigos utilizados.

En el ámbito de la gestión de bases de datos, SQL Server es ampliamente reconocido como un sistema robusto y confiable. Su capacidad para almacenar, administrar y manipular grandes volúmenes de datos lo convierte en una herramienta fundamental en el campo de la informática y la gestión de la información. A través de este informe, explicaremos diferentes aspectos de SQL Server y su aplicación práctica en la realización de consultas.

Durante el desarrollo del informe, se mostrarán ejemplos de consultas utilizando código SQL específico, acompañados de capturas de pantalla que ilustran el resultado obtenido en cada caso. Además, se proporcionarán explicaciones detalladas de cada línea de código, destacando su propósito y función dentro del contexto de la base de datos.

El objetivo principal de este informe es proporcionar una comprensión clara y concisa del proceso de consulta en SQL Server, así como de las capacidades y características que ofrece este sistema de gestión de bases de datos. A través de la ejecución y análisis de las consultas, buscamos fortalecer nuestros conocimientos en la manipulación de datos, la extracción de información relevante y la comprensión de las estructuras y relaciones presentes en la base de datos.

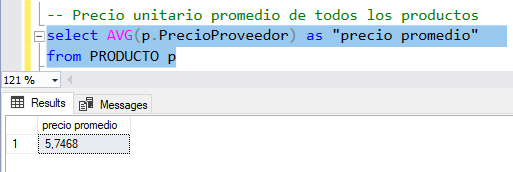
1. **DESARROLLO**

“**Adjuntando Evidencias”**

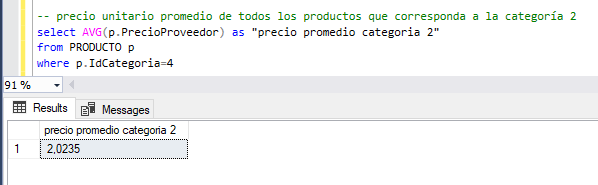
1. **Primer Video**

**1.1 Función AVG**

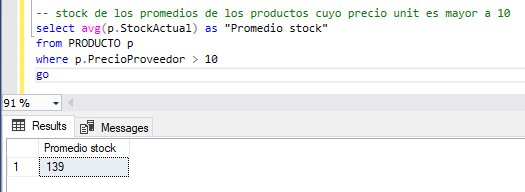
* La consulta selecciona el promedio de los precios de proveedor de la tabla PRODUCTO y lo muestra como "precio promedio".



* La consulta calcula el promedio de los precios de proveedor de los productos de la categoría con el identificador 4 y lo muestra como "precio promedio categoría 2".

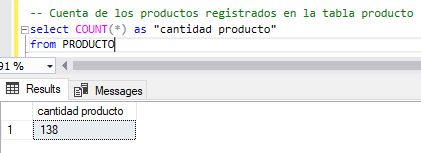


* La consulta calcula el promedio del stock actual de los productos cuyo precio de proveedor es mayor a 10 y lo muestra como "Promedio stock".

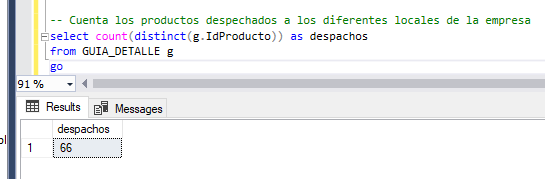
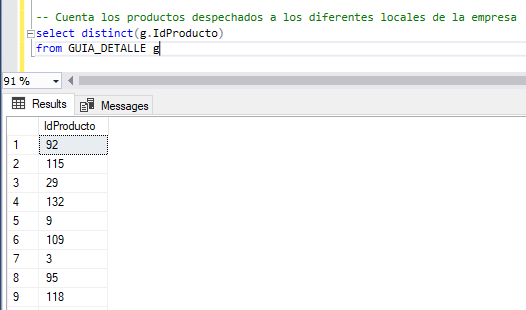
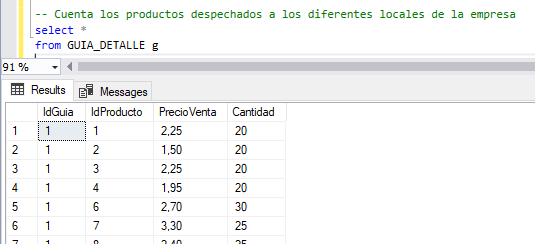


**1.2 Función COUNT y DISTINC**

* La consulta cuenta la cantidad de registros en la tabla PRODUCTO y lo muestra como "cantidad producto".

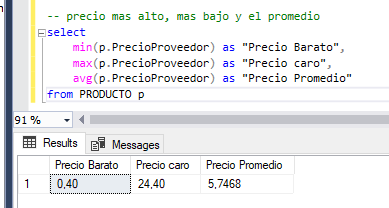


* La consulta selecciona todas las columnas de la tabla GUIA\_DETALLE y muestra todos los registros.

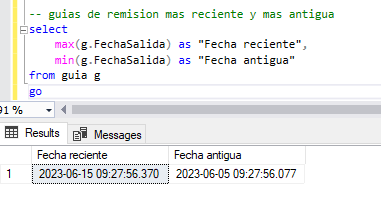


**1.3 Función MIN, MAX y AVG**

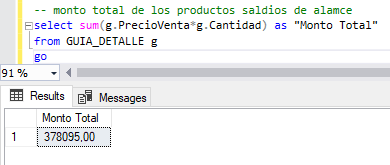
* La consulta calcula el precio más bajo, el precio más alto y el precio promedio de los proveedores de los productos y los muestra con las etiquetas "Precio Barato", "Precio Caro" y "Precio Promedio" respectivamente.



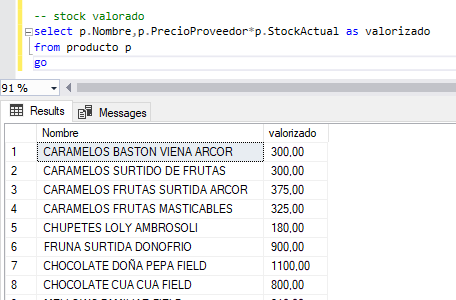
* La consulta selecciona la fecha más reciente y la fecha más antigua de la columna FechaSalida de la tabla guia y las muestra con las etiquetas "Fecha reciente" y "Fecha antigua", respectivamente.



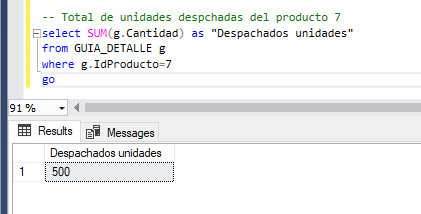
* La consulta calcula el monto total multiplicando el PrecioVenta por la Cantidad en cada registro de la tabla GUIA\_DETALLE y muestra el resultado como "Monto Total".



* La consulta selecciona el nombre del producto (columna Nombre) y calcula el valorizado multiplicando el PrecioProveedor por el StockActual de cada producto en la tabla producto, y lo muestra como "valorizado".

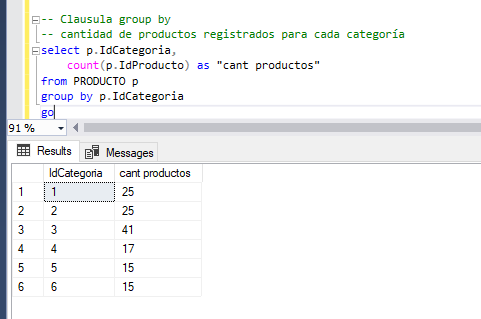


* La consulta calcula la suma total de la cantidad despachada (columna Cantidad) en la tabla GUIA\_DETALLE, donde el IdProducto es igual a 7, y muestra el resultado como "Despachados unidades".

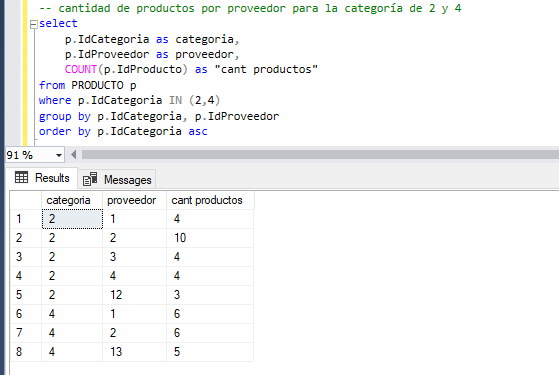


**1.4 Cláusula GROUP BY**

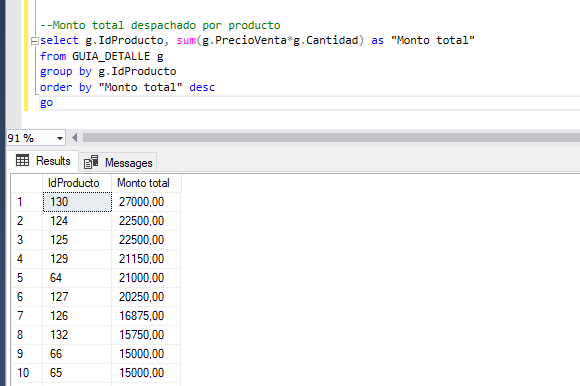
* La consulta proporciona un resumen de la cantidad de productos registrados en cada categoría. Utilizando la cláusula GROUP BY junto con la función COUNT, se agrupan los productos por categoría y se cuenta la cantidad de productos en cada categoría.



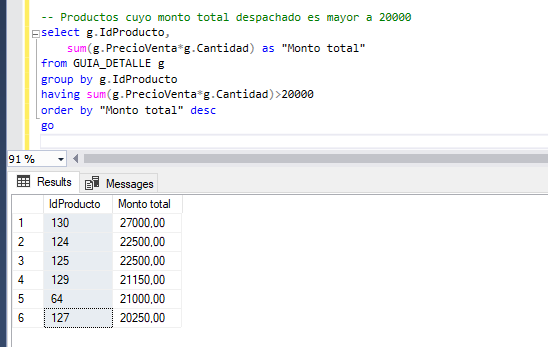
* La consulta selecciona la columna IdCategoria como "categoria", la columna IdProveedor como "proveedor" y utiliza la función de agregación COUNT para contar la cantidad de productos (columna IdProducto) en cada combinación de categoría y proveedor.



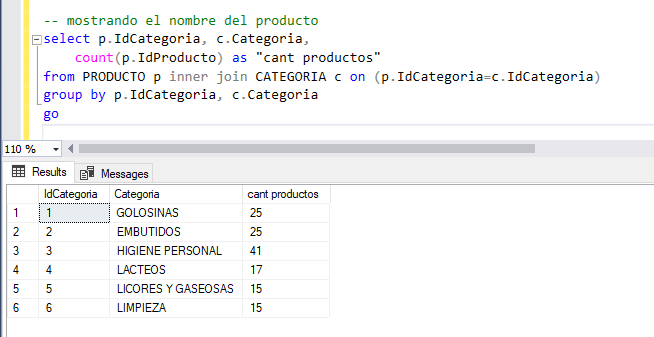
* La consulta selecciona la columna IdProducto y utiliza la función de agregación SUM para calcular el monto total multiplicando el PrecioVenta por la Cantidad en cada registro de la tabla GUIA\_DETALLE. Los resultados se agrupan por el IdProducto.



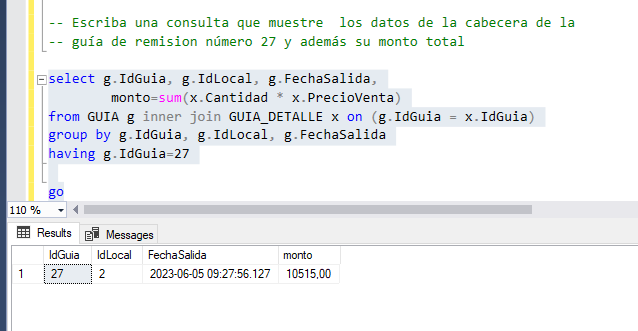
* La consulta selecciona la columna IdProducto y utiliza la función de agregación SUM para calcular el monto total multiplicando el PrecioVenta por la Cantidad en cada registro de la tabla GUIA\_DETALLE. Los resultados se agrupan por el IdProducto. Luego, la cláusula HAVING se utiliza para filtrar los resultados y mostrar solo aquellos registros cuyo monto total despachado es mayor a 20000.



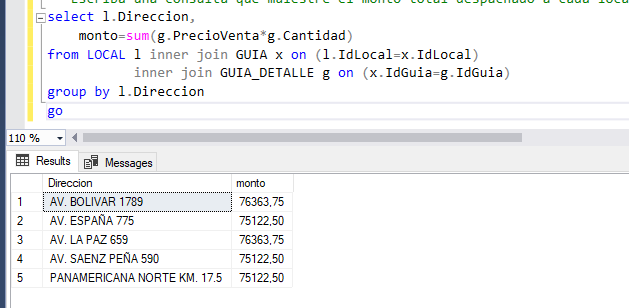
* La consulta realiza una combinación interna (inner join) entre las tablas PRODUCTO y CATEGORIA utilizando la condición de igualdad entre las columnas IdCategoria de ambas tablas. Selecciona la columna IdCategoria de la tabla PRODUCTO, la columna Categoria de la tabla CATEGORIA y utiliza la función de agregación COUNT para contar la cantidad de productos (columna IdProducto) en cada combinación de categoría y categoría. Los resultados se agrupan por el IdCategoria y la Categoria.



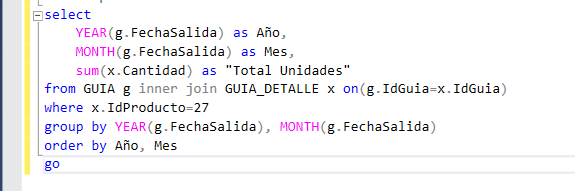
* La consulta combina internamente (inner join) las tablas GUIA y GUIA\_DETALLE utilizando la condición de igualdad entre las columnas IdGuia de ambas tablas. Selecciona las columnas IdGuia, IdLocal y FechaSalida de la tabla GUIA. También calcula la columna monto sumando la multiplicación de la Cantidad por el PrecioVenta en cada registro de la tabla GUIA\_DETALLE. Los resultados se agrupan por IdGuia, IdLocal y FechaSalida.



* La consulta combina internamente (inner join) las tablas LOCAL, GUIA y GUIA\_DETALLE utilizando las condiciones de igualdad entre las columnas IdLocal de LOCAL y GUIA, y entre las columnas IdGuia de GUIA y GUIA\_DETALLE. Selecciona la columna Direccion de la tabla LOCAL. También calcula la columna monto sumando la multiplicación de PrecioVenta por Cantidad en cada registro de la tabla GUIA\_DETALLE. Los resultados se agrupan por la columna Direccion de la tabla LOCAL.



* La consulta combina internamente (inner join) las tablas GUIA y GUIA\_DETALLE utilizando la condición de igualdad entre las columnas IdGuia de ambas tablas. Selecciona el año (YEAR) y mes (MONTH) de la columna FechaSalida de la tabla GUIA. También calcula la columna "Total Unidades" sumando la columna Cantidad en cada registro de la tabla GUIA\_DETALLE. Se aplica una cláusula WHERE para filtrar los resultados y mostrar solo aquellos registros donde el IdProducto sea igual a 27. Los resultados se agrupan por el año y mes de la FechaSalida. Luego, se ordenan por el año y mes en orden ascendente.



* La consulta combina internamente (inner join) las tablas GUIA, GUIA\_DETALLE y PRODUCTO utilizando las condiciones de igualdad entre las columnas IdGuia de GUIA y GUIA\_DETALLE, y entre las columnas IdProducto de GUIA\_DETALLE y PRODUCTO. Selecciona la columna Nombre de la tabla PRODUCTO, y el año (YEAR) y mes (MONTH) de la columna FechaSalida de la tabla GUIA. También calcula la columna "Total Unidades" sumando la columna Cantidad en cada registro de la tabla GUIA\_DETALLE. Los resultados se agrupan por el nombre del producto, año y mes de la FechaSalida. Luego, se ordenan por el nombre del producto, año y mes en orden ascendente.

Texto

Descripción generada automáticamente

**1.5 Sub-Consultas**

* La consulta selecciona el IdProducto, Nombre, PrecioProveedor y la columna "diferencia" de la tabla PRODUCTO. La columna "diferencia" se calcula restando el PrecioProveedor de cada producto al precio promedio de todos los productos. El precio promedio se obtiene utilizando una subconsulta que selecciona la media (AVG) del PrecioProveedor de la tabla PRODUCTO.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

* La primera consulta selecciona la fecha máxima (última fecha) de la columna FechaSalida en la tabla GUIA utilizando la función MAX.

La segunda consulta combina internamente (inner join) las tablas GUIA, GUIA\_DETALLE y PRODUCTO utilizando las condiciones de igualdad entre las columnas IdGuia de GUIA y GUIA\_DETALLE, y entre las columnas IdProducto de GUIA\_DETALLE y PRODUCTO. Selecciona el IdProducto distinto, el Nombre del producto y la FechaSalida formateada en formato dd/mm/aaaa utilizando la función CONVERT y el estilo 103. Se aplica una cláusula WHERE para filtrar los resultados y mostrar solo aquellos registros donde la FechaSalida coincida con la fecha máxima obtenida en la primera consulta.

Texto

Descripción generada automáticamente

1. **Segundo Video**

**2.1 Create database Informe2023**

* Se inicia la creación de una tabla llamada "USUARIO" con tres columnas: "cod\_user" (entero), "nam\_user" (cadena de caracteres) y "status\_user" (carácter). La columna "cod\_user" se establece como clave primaria.

Texto

Descripción generada automáticamente

* Se inserta un registro directamente en la tabla "USUARIO" con los valores 1, 'MYBER' y '1' para las columnas "cod\_user", "nam\_user" y "status\_user" respectivamente.

Texto

Descripción generada automáticamente con confianza media

* Se realiza una consulta a la tabla "USUARIO" para mostrar todos los registros. Se insertan varios registros directamente en la tabla "USUARIO" con diferentes valores para las columnas "cod\_user", "nam\_user" y "status\_user".

Interfaz de usuario gráfica, Texto, Chat o mensaje de texto

Descripción generada automáticamente

* Se utiliza una forma diferente de inserción de registros en la tabla "USUARIO" especificando las columnas a las que se les asignarán los valores.

Texto

Descripción generada automáticamente

* Se crea una nueva tabla llamada "USUARIOUCV" con tres columnas: "cod\_UCVuser" (entero), "nom\_UCVuser" (cadena de caracteres) y "status\_UCVuser" (carácter). La columna "cod\_UCVuser" se establece como clave primaria. Se realiza una inserción múltiple en la tabla "USUARIOUCV" seleccionando los valores de las columnas "cod\_user", "nam\_user" y "status\_user" de la tabla "USUARIO" donde el "status\_user" es igual a '1'.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

* Se eliminan todos los registros de la tabla "USUARIOUCV".
* Se elimina un registro específico de la tabla "USUARIO" utilizando la cláusula WHERE para filtrar por el valor de la columna "cod\_user".

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

* Se actualiza el valor de la columna "status\_user" en la tabla "USUARIO" a '0' (inactivo) para el registro donde el "cod\_user" es igual a 5.
* Se realiza una consulta a la tabla "USUARIO" para mostrar todos los registros después de la actualización.
* Se actualiza el valor de la columna "status\_UCVuser" en la tabla "USUARIOUCV" a '0' (inactivo) para todos los registros.

Texto

Descripción generada automáticamente

* Se realiza una consulta a la tabla "USUARIOUCV" para mostrar todos los registros después de la actualización.

Texto

Descripción generada automáticamente

* Se actualiza el valor de la columna "status\_user" en la tabla "USUARIO" a '0' (inactivo) para el registro donde el "cod\_user" es igual a 9.
* Se realiza una consulta a la tabla "USUARIO" para mostrar solo los registros con "status\_user" igual a '1' (activos).
* Se realiza una consulta a la tabla "USUARIO" para mostrar solo los registros con "status\_user" igual a '0' (inactivos).

Texto

Descripción generada automáticamente

* 1. **Create database bd\_escuelaucv**

• Se utiliza la declaración "if DB\_ID('BD\_ESCUELAUCV') is not null" para verificar la existencia de una base de datos llamada "BD\_ESCUELAUCV". La función DB\_ID() devuelve el identificador de la base de datos si existe y no es NULL. Si la base de datos existe, se ejecuta el comando "drop database bd\_escuelaucv" para eliminarla.

• Una vez eliminada la base de datos existente (si la hay), se crea una nueva base de datos llamada "bd\_escuelaucv" mediante el comando "create database bd\_escuelaucv". Esto crea una base de datos vacía con el nombre especificado.

Captura de pantalla de computadora

Descripción generada automáticamente

• Se procede a crear varias tablas dentro de la base de datos "bd\_escuelaucv" utilizando el comando "create table". Cada tabla tiene un nombre y una lista de columnas con sus respectivos tipos de datos y restricciones.

- La tabla "curso" almacena información sobre los cursos, como su identificador, nombre, horas teóricas, horas prácticas, nivel y grado.

Texto

Descripción generada automáticamente

- La tabla "docente" almacena información sobre los docentes, incluyendo su identificador, nombre, apellidos, dirección, teléfono, DNI, especialidad, correo electrónico y sexo.

Texto

Descripción generada automáticamente

- La tabla "ubigeo" guarda información sobre los lugares geográficos, como identificador, distrito, provincia y departamento.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

- La tabla "alumno" almacena información sobre los alumnos, incluyendo su identificador, nombre, apellidos, fecha de nacimiento, teléfono, sexo, correo electrónico y el identificador del ubigeo al que pertenecen.

Pantalla de computadora con letras

Descripción generada automáticamente

- La tabla "promedio" registra los promedios de los alumnos en los cursos, con su identificador, el identificador del curso y el promedio.

Texto

Descripción generada automáticamente

- La tabla "asignacion" guarda información sobre la asignación de docentes a cursos, con el identificador del docente, el identificador del curso y la sección.

Texto

Descripción generada automáticamente

- La tabla "notas" almacena las notas de los alumnos en los cursos, con el identificador del curso, el identificador del alumno, las notas de cada bimestre y el promedio.

Texto

Descripción generada automáticamente

• Después de crear las tablas, se aplican restricciones adicionales utilizando las declaraciones "ALTER TABLE". Estas restricciones incluyen valores predeterminados (DEFAULT) y restricciones de verificación (CHECK) para garantizar la integridad de los datos.

Captura de pantalla de computadora

Descripción generada automáticamente

• También se aplican restricciones de unicidad (UNIQUE) en ciertos campos para asegurar que los valores sean únicos en las respectivas tablas.

Texto

Descripción generada automáticamente Texto

Descripción generada automáticamente Texto

Descripción generada automáticamente Texto

Descripción generada automáticamente

• Por último, se realizan algunas inserciones de datos de ejemplo en la tabla "curso" utilizando la declaración "insert into". Se agregan dos registros, uno con todos los valores de columna especificados y otro con solo algunos valores. Estos registros representan cursos con identificadores "C0001" y "C0002" respectivamente.

Texto

Descripción generada automáticamente

* 1. **Create database UCVBDNEGOCIO2022**
* Crear la base de datos: La primera línea del script crea una base de datos llamada "UCVBDNEGOCIOS2022".
* Usar la base de datos: La línea "use UCVBDNEGOCIOS2022" establece la base de datos recién creada como la base de datos actual para las operaciones posteriores.
* Establecer el formato de fecha: La línea "set dateformat ymd" establece el formato de fecha para el sistema en "año-mes-día".
* Crear esquemas: Las líneas que comienzan con "Create Schema" crean tres esquemas en la base de datos: Ventas, Compras y RRHH. Los esquemas son contenedores lógicos que se utilizan para organizar y agrupar objetos relacionados en una base de datos.

Captura de pantalla de computadora

Descripción generada automáticamente

* 5. Crear tablas: A continuación, se crean varias tablas en los esquemas creados anteriormente. Cada tabla tiene columnas que definen su estructura y restricciones.

- La tabla "Compras.categorias" tiene columnas para el ID de categoría, el nombre de la categoría y la descripción.

Texto

Descripción generada automáticamente

- La tabla "Ventas.paises" tiene columnas para el ID de país y el nombre del país.

Texto

Descripción generada automáticamente

- La tabla "Ventas.clientes" tiene columnas para el ID de cliente, el nombre del cliente, la dirección, el ID de país y el teléfono del cliente. Texto

Descripción generada automáticamente

- La tabla "Compras.proveedores" tiene columnas para el ID de proveedor, el nombre del proveedor, la dirección, el nombre y cargo del contacto, el ID de país, el teléfono y el fax del proveedor.

Texto

Descripción generada automáticamente

- La tabla "Compras.productos" tiene columnas para el ID de producto, el nombre del producto, el ID de proveedor, el ID de categoría, la cantidad por unidad, el precio por unidad, las unidades en existencia y las unidades en pedido.

Texto

Descripción generada automáticamente

- La tabla "RRHH.Cargos" tiene columnas para el ID de cargo y la descripción del cargo.

- La tabla "RRHH.Distritos" tiene columnas para el ID de distrito y el nombre del distrito.

- La tabla "RRHH.empleados" tiene columnas para el ID de empleado, el apellido, el nombre, la fecha de nacimiento, la dirección, el ID de distrito, el teléfono, el ID de cargo y la fecha de contratación.

Texto

Descripción generada automáticamenteTexto

Descripción generada automáticamente

* Insertar datos en las tablas: Después de crear las tablas, se insertan datos de ejemplo en algunas de ellas utilizando la declaración "INSERT INTO". Se agregan filas a las tablas "Compras.categorias", "Ventas.paises", "Ventas.clientes", "Compras.proveedores", "Compras.productos", "RRHH.Cargos" y "RRHH.empleados".
* Crear la tabla "Ventas.pedidoscabe": Se crea una nueva tabla llamada "Ventas.pedidoscabe" con varias columnas para almacenar información sobre los pedidos, como el ID de pedido, el ID de cliente, el ID de empleado, las fechas de pedido, entrega y envío, la cantidad de pedido, el destinatario, la dirección de entrega y el estado del pedido.

Texto

Descripción generada automáticamentePantalla de computadora con letras

Descripción generada automáticamente con confianza media

* Crear la tabla "Ventas.pedidosdetalle": Se crea otra tabla llamada "Ventas.pedidosdetalle" con columnas para el ID de pedido, el ID de producto, la cantidad de producto y el precio unitario.

Texto

Descripción generada automáticamentePatrón de fondo

Descripción generada automáticamente con confianza baja

Imagen que contiene Texto

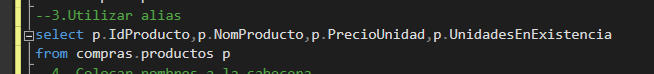
Descripción generada automáticamente

* Utilizando el operador \*: Esta consulta muestra todas las filas y columnas de la tabla "compras.proveedores". El operador \* se utiliza para seleccionar todas las columnas en la salida.
* Especificando columnas: En esta consulta, se seleccionan columnas específicas de la tabla "compras.productos", incluyendo "idproducto", "NomProducto", "CantxUnidad" y "PrecioUnidad". Solo se mostrarán estas columnas en la salida.

Texto

Descripción generada automáticamente

* Utilizar alias: En esta consulta, se utiliza el alias "p" para abreviar el nombre de la tabla "compras.productos" en la consulta. Esto simplifica la escritura y lectura de la consulta.



* Colocar nombres a la cabecera: Esta consulta utiliza alias para asignar nombres personalizados a las columnas en la salida. Por ejemplo, "p.IdProducto as CODIGO" asigna el nombre "CODIGO" a la columna "IdProducto" en la salida resultante.

Texto

Descripción generada automáticamente

* USANDO la función DISTINCT: La cláusula DISTINCT se utiliza para eliminar duplicados en el resultado de la consulta. En estas consultas, se selecciona la columna "IdProveedor" y "IdCategoria" de la tabla "compras.productos" eliminando cualquier duplicado.

Texto

Descripción generada automáticamente

* ORDER BY ASC y DESC: La cláusula ORDER BY se utiliza para ordenar los resultados de la consulta según una columna específica en orden ascendente (ASC) o descendente (DESC). Estas consultas ordenan los resultados de la tabla "compras.productos" por la columna "NomProducto" en orden descendente y ascendente respectivamente.

Texto

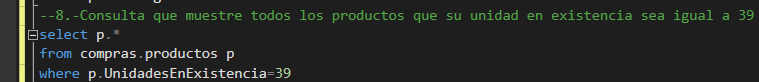
Descripción generada automáticamente

* CLAUSULA WHERE: Esta consulta selecciona todas las filas de la tabla "compras.productos" donde el valor de la columna "IdCategoria" es igual a 3.

Texto

Descripción generada automáticamente

* Consulta que muestra todos los productos cuya unidad en existencia es igual a 39.



* Consulta que muestra todos los productos cuyo precio por unidad es mayor que 100.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

* Consulta que muestra los productos de la categoría 1 y cuya unidad en existencia sea mayor a 50.

Pantalla de un video juego

Descripción generada automáticamente con confianza media

* Consulta que muestra los productos de las categorías "lacteos" o "carnes".

Texto

Descripción generada automáticamente

* Consulta que muestra los productos cuya unidad en existencia está entre 50 y 100, o cuyo precio está entre 5 y 30.

Texto

Descripción generada automáticamente

* Operador LIKE: Esta consulta muestra los productos cuya descripción comienza con la letra "C", seguida de cualquier letra.

Texto

Descripción generada automáticamente

* Esta consulta muestra los productos cuya descripción tiene la letra "e" en la segunda posición.

Texto

Descripción generada automáticamente

* Consulta que muestra los productos cuyo precio está entre 10 y 25 utilizando el operador BETWEEN.

Texto

Descripción generada automáticamente

* Operador IN: Esta consulta muestra los productos cuyo precio por unidad es 30, 35 o 100, utilizando el operador IN para verificar si el precio está en la lista especificada.

Captura de pantalla de un celular

Descripción generada automáticamente

* Función DAY: Esta consulta utiliza la función DAY para obtener el día correspondiente de la fecha de pedido en la tabla "Ventas.pedidoscabe". También se muestran otras partes de la fecha, como el mes, el año y la fecha completa.

Texto

Descripción generada automáticamente

* Función DATEPART: Esta consulta utiliza la función DATEPART para obtener partes específicas de la fecha de pedido, como el día, el mes y el año. Los resultados se formatean para mostrar la fecha de pedido en un formato personalizado.

Texto

Descripción generada automáticamente

# Anexos

* Base de datos “Informe2023“

Texto

Descripción generada automáticamente

* Base de datos “bd\_escualaucv”

Texto

Descripción generada automáticamente

* Base de datos “UCVBDNEGOCIOS2022”

Texto

Descripción generada automáticamente