

Actividad 6: Funciones SQL

NOMBRE DEL ALUMNO: GARCÍA QUIROZ GUSTAVO IVAN

GRUPO: 3CV2

BASE DE DATOS

05/12/2023

Actividad 6: Funciones SQL

Instrucciones

Realizar los 9 ejercicios que vienen en el script, en equipo de 3 personas, elaborar su reporte en formato .PDF colocar:

--1.Crear una funcion que reciba como parametro la fecha de nacimiento y devuelva la edad al dia de hoy*/

```
CREATE FUNCTION dbo.CalcularEdad(@FechaNacimiento datetime)
RETURNS INT
AS
BEGIN
    DECLARE @Edad INT
    SET @Edad = DATEDIFF(YEAR, @FechaNacimiento, GETDATE())
    IF (MONTH(@FechaNacimiento) > MONTH(GETDATE())) OR (MONTH(@FechaNacimiento) =
MONTH(GETDATE()) AND DAY(@FechaNacimiento) > DAY(GETDATE()))
        SET @Edad = @Edad - 1
    RETURN @Edad
END
```

```
SELECT dbo.CalcularEdad('1980-11-01') AS Edad
```

	Edad
1	43

--2.crear una funcion que reciba la matricula de un avion y devuelva la siguiente fecha de mantenimiento */

```
CREATE FUNCTION dbo.ProximaFechaMantenimiento(@Matricula INT)
RETURNS DATETIME
AS
BEGIN
    DECLARE @ProximaFecha DATETIME
    SELECT @ProximaFecha = DATEADD(MONTH, mesesMantenimiento, UltimoMantenimiento)
    FROM dbo.Avion
    WHERE matricula = @Matricula
    RETURN @ProximaFecha
END
```

```
SELECT dbo.ProximaFechaMantenimiento(234312) AS ProximaFecha
```

	ProximaFecha
1	2014-04-01 00:00:00.000

/*3.crear una funcion que reciba una fecha y muestre la siguiente informacion del vuelo mas cercano:

matricula del avion, nombre del vendedor, nombre del cliente, fecha de vuelo y precio del asiento*/

```
CREATE FUNCTION dbo.InformacionVueloCercano(@Fecha DATETIME)
RETURNS TABLE
AS
RETURN
(
```

```

SELECT TOP 1
    V.matriculaAvion,
    P.nombre AS nombreVendedor,
    C.nombre AS nombreCliente,
    V.fechaVuelo,
    V.precioPorAsiento
FROM Vuelo V
JOIN Venta VE ON V.idVuelo = VE.idVuelo
JOIN Vendedor VN ON VE.idVendedor = VN.vendedorId
JOIN Persona P ON VN.personaId = P.IdPersona
JOIN Cliente CL ON VE.idCliente = CL.clienteId
JOIN Persona C ON CL.personaId = C.IdPersona
WHERE V.fechaVuelo > @Fecha
ORDER BY V.fechaVuelo ASC
)

```

```
SELECT * FROM dbo.InformacionVueloCercano('2014-01-02')
```

	matriculaAvion	nombreVendedor	nombreCliente	fechaVuelo	precioPorAsiento
1	432734	Edgardo	Monica	2014-01-15 00:00:00.000	4730.00

/*4.crear una funcion que reciba el id de un cliente y el id de un vuelo y regrese el precio del vuelo considerando el descuento por tipo de cliente*/

```

CREATE FUNCTION dbo.PrecioConDescuento(@idCliente INT, @idVuelo INT)
RETURNS MONEY
AS
BEGIN
    DECLARE @Precio MONEY
    DECLARE @Descuento INT
    DECLARE @PrecioFinal MONEY

    SELECT @Precio = precioPorAsiento FROM Vuelo WHERE idVuelo = @idVuelo
    SELECT @Descuento = descuento FROM TipoCliente WHERE idTipoCliente = (SELECT idTipoCliente FROM Cliente WHERE clienteId = @idCliente)

    SET @PrecioFinal = @Precio - (@Precio * @Descuento / 100)

    RETURN @PrecioFinal
END
SELECT dbo.PrecioConDescuento(1, 2) AS PrecioFinal

```

	PrecioFinal
1	3000.00

/*5.Crear una funcion que regrese el RFC de todas las personas de la base de datos o se especifique algun IdPersona para solo obtener el de una*/

```

CREATE FUNCTION dbo.ObtenerRFC(@IdPersona INT = NULL)
RETURNS TABLE
AS
RETURN
(
    SELECT
        P.IdPersona,
        LEFT(P.paterno, 2) + LEFT(P.materno, 1) + LEFT(P.nombre, 1) +

```

```

        CONVERT(VARCHAR(2), DAY(P.nacimiento)) +
        CONVERT(VARCHAR(2), MONTH(P.nacimiento)) +
        RIGHT(CONVERT(VARCHAR(4), YEAR(P.nacimiento)), 2) +
        'XXX' AS RFC
    FROM Persona P
    WHERE (@IdPersona IS NULL OR P.IdPersona = @IdPersona)
)
SELECT * FROM dbo.ObtenerRFC()
SELECT * FROM dbo.ObtenerRFC(1)

```

	IdPersona	RFC
1	1	HeME30880XXX

/*6.crear una funcion que reciba como parametros el Id de un cliente, y una fecha inicio y fin. Como salida debe indicar cuantas veces ha viajado*/

```

CREATE FUNCTION dbo.NumeroViajes(@idCliente INT, @FechaInicio DATETIME, @FechaFin DATETIME)
RETURNS INT
AS
BEGIN
    DECLARE @NumeroViajes INT

    SELECT @NumeroViajes = COUNT(*)
    FROM Venta
    WHERE idCliente = @idCliente AND fechaCompra BETWEEN @FechaInicio AND @FechaFin

    RETURN @NumeroViajes
END
SELECT dbo.NumeroViajes(1, '2011-01-01', '2022-12-31') AS NumeroViajes

```

	NumeroViajes
1	3

/*7.Crear una funcion que reciba como parametro un sueldo base y que muestre el nombre completo de los empleados que tengan un salario mayor o igual al parametro de entrada de mostrar la cantidad de años que lleva el empleado desde su fecha de ingreso.*/

```

CREATE FUNCTION fn_empleados_por_sueldo(@sueldo_base money)
RETURNS TABLE
AS
RETURN
(
    SELECT
        P.nombre + ' ' + P.paterno + ' ' + ISNULL(P.materno, '') AS NombreCompleto,
        E.sueldo,
        DATEDIFF(YEAR, E.fechaIngreso, GETDATE()) AS AnosDeServicio
    FROM
        Empleado E
    INNER JOIN
        Persona P ON E.idPersona = P.IdPersona
    WHERE
        E.sueldo >= @sueldo_base
)
SELECT * FROM fn_empleados_por_sueldo(50000)

```

	NombreCompleto	sueldo	AnosDeServicio
1	Elisa Romero Ayala	50000.00	1

```

/*8.modificar la funcion ObtenerDireccionEmpleado, en caso de que el Id
proporcionado no tenga jefe, mostrar en la columna
direccion completa la siguiente leyenda "informacion confidencial"*/
CREATE FUNCTION ObtenerDireccionEmpleado(@idEmpleado int)
RETURNS TABLE
AS
RETURN
(
    SELECT
        CASE
            WHEN E.idJefe IS NULL THEN 'Informacion confidencial'
            ELSE D.calle + ' ' + D.numero + ', ' + D.colonia + ', ' + D.codigoPostal
        END AS DireccionCompleta
    FROM
        Empleado E
    INNER JOIN
        Persona P ON E.idPersona = P.IdPersona
    INNER JOIN
        Direccion D ON P.direccionId = D.direccionId
    WHERE
        E.idEmpleado = @idEmpleado
)
SELECT * FROM ObtenerDireccionEmpleado(1)
SELECT * FROM ObtenerDireccionEmpleado(5)

```

	DireccionCompleta
1	Av. 20 de Noviembre 2, Centro, 06060

	DireccionCompleta
1	Informacion confidencial

/*9.crear una funcion que devuelva el precio por asiento de todos los vuelos y que reciba un parametro opcional que indique si es temporada alta, en su caso aumentar 20% el precio por asiento*/

```

CREATE FUNCTION fnPrecioAsiento(@temporadaAlta bit = 0)
RETURNS TABLE
AS
RETURN
(
    SELECT
        idVuelo,
        CASE
            WHEN @temporadaAlta = 1 THEN precioPorAsiento * 1.2
            ELSE precioPorAsiento
        END AS PrecioPorAsiento
    FROM
        Vuelo
)
SELECT * FROM fnPrecioAsiento(1)

```

	idVuelo	PrecioPorAsier
1	2	3600.00000
2	3	7200.00000
3	4	4440.00000
4	6	2400.00000
5	7	4404.00000
6	8	3108.00000
7	9	7200.00000
8	10	3000.00000
9	12	3600.00000
10	13	6000.00000
11	14	5676.00000
12	15	4838.40000
13	16	2808.00000
14	17	8400.00000
15	18	8762.40000
16	19	7608.00000
17	20	1588.80000