

BASE DE DATOS

PROCESUAL HITO 4

ESTUDIANTE:

QUIROGA HUARISTE
ANDRÉS VLADIMIR

DOCENTE:

LIC. BARRA PAREDES WILLIAM

CARRERA:

INGENIERIA DE SISTEMAS

GESTIÓN:

2022

MANEJO DE CONSULTAS

2.1 Muestra un ejemplo de DDL

```
--Creación de base de datos
CREATE DATABASE BaseDeDatos1;

--Cambio de nombre a una base de datos
ALTER DATABASE BaseDeDatos1
MODIFY NAME=bbdd01

--Eliminación de una base de datos
DROP DATABASE bbdd01;
```

2.2 Muestra un ejemplo de DML

```
INSERT INTO Vehiculos
(codVehiculo, Descripcion)
VALUES (1, 'Furgoneta')

INSERT INTO Vehiculos
VALUES (2, 'Coche')
```

2.3 Para que sirve **INNER JOIN**?

R. Combina los registros de dos tablas si hay valores coincidentes en un campo común.

2.4 Defina que es una función de **Agregación**.

R. Las funciones de agregación en SQL nos permiten efectuar operaciones sobre un conjunto de resultados

2.5 Liste funciones de agregación que conozca.

- ❖ AVG. – Utilizada para calcular el promedio de los valores de un campo determinado.
- ❖ COUNT. – Utilizada para devolver el numero de registros de la selección.
- ❖ SUM. – Utilizada para devolver la suma de todos los valores de un campo determinado.
- ❖ MAX. –Utilizada para devolver el valor mas alto de un campo especificado.
- ❖ MIN. –Utilizada para devolver el valor mas bajo de un campo especificado.

2.6 Mencione algunas funciones propias de **SQL-Server**

- GROUP BY. – Es una sentencia que va ligada a las funciones agregadas.

2.7 Para que sirve la función **CONCAT** en **SQL-server**.

R. Para unir dos o mas cadenas en una.

2.8 Muestre un ejemplo del uso de COUNT.

Query 15 x

```
1 * SELECT * FROM eli.authors;
```

Filter

	idauthors	name	birthday
▶	1	Edu	1971-02-02
	2	Eli	1985-01-01
	3	Pedro	1940-03-03
+

2.9 Muestre un ejemplo del uso de AVG.

SQLQuery1.sql - NC8430\...\Fran...2))*

```
= select AVG(Calificacion)  
L from AlumnosBD
```

Resultados Mensajes

	(Sin nombre de columna)
1	87

SQL Server Enterprise Edition (10.50 RTM) | nc8430\Francisco (52) | MiPrimeraBD | 00:00:00 | 1 filas

2.9 Muestra un ejemplo del uso de MIN-MAX

SQLQuery1.sql - NC8430\...\Fran...2))*

```
select min(Sueldo)
from Usuarios
```

Resultados Mensajes

	(Sin nombre de columna)
1	800

RTM) | nc8430\Francisco (52) | MiPrimeraBD | 00:00:00 | 1 filas

Query 1

```
having max(precio)>=30;
select editorial, max(precio) as 'mayor'
from libros
group by editorial
having mayor>=30;
```

Result Grid

	editorial	mayor
▶	Planeta	40.00
	Paidós	46.00

Result 1 Result 2 Result 3 Result 4 Result 5 Result 6 Result 7 Result 8 Result 9 Result 10

- Manejo de Consultas.

3.1 Mostrar que jugadores que formen parte del equipo equ-333

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'Database Explorer' for 'Microsoft SQL Server (JTds) - @localhost'. The 'Defensa_Hito_4_Grupo1' database is expanded, showing tables like 'campeonato', 'equipo', and 'jugador'. The 'Database Objects' section is also visible.

The central pane shows a SQL query in the 'console_1' window. The query is as follows:

```
--mostrar el nombre del campeonato, el nombre del equipo, nombres y apellidos de los jugadores que pertenezcan al equipo cuyo id
select cam.nombre_campeonato , equi.nombre_equipo, jug.nombres, jug.apellidos
from jugador as jug
inner join equipo as equi 1.n<->1: on jug.id_equipo = equi.id_equipo
inner join campeonato as cam 1.n<->1: on equi.id_campeonato = cam.id_campeonato
where jug.id_equipo = 'equ-222';

--cuantos equipos inscritos hay
select Count (equi.id_equipo)
from equipo as equi;

--mostrar que jugadores que formen parte del equipo equ-333
select jug.nombres as jugadores
from jugador as jug
inner join equipo as equi 1.n<->1: on jug.id_equipo = equi.id_equipo
WHERE jug.id_equipo = 'equ-333';

--crear una funcion que permita saber cuantos jugadores estan inscritos.
--la funcion debe llamarse Crear una funcionn que permita saber cuantos jugadores estan inscritos.
--La funcion debe llamarse f1_CantidadJugadores()
```

The bottom pane shows the 'Services' section with a list of console windows. The 'console_1' window is selected, showing the results of the query. The results are displayed in a table with 2 rows:

jugadores
1 Sandra
2 Ana

The status bar at the bottom indicates '2 rows retrieved starting from 1 in 126 ms (execution: 27 ms, fetching: 99 ms)'. The system tray shows the date and time as 15/6/2022 19:18.

- 3.2. Crear una función que permita saber cuántos jugadores están inscritos.
- ■ La función debe llamarse Crear una función que permita saber cuántos jugadores están inscritos.
- ■ La función debe llamarse F1_CantidadJugadores()

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the database structure for 'Defensa_Hito_4_Grupo1' (1 of 13), including tables like 'campeonato', 'columns', 'keys', 'indexes', 'equipo', and 'jugador'. The main editor shows the following SQL script:

```

87 WHERE jug.id_equipo = 'equ-333';
88
89 --crear una funcion que permita saber cuantos jugadores estan inscritos.
90 --la funcion debe llamarse Crear una funcion que permita saber cuantos jugadores estan inscritos.
91 --La funcion debe llamarse f1_CantidadJugadores()
92
93 create or alter function F1_CantidadJugadores()
94 returns integer as
95 begin
96     declare @Cantidad_Jugadores integer;
97
98     select @Cantidad_Jugadores = Count(jug.nombres)
99     from jugador as jug
100     return @Cantidad_Jugadores
101 end;
102
103 select dbo.F1_CantidadJugadores() as jugadores_inscritos
104
105 --Crear una funcion que permita saber cuantos jugadores estan inscritos y que sean de la categoria varones o mujeres.
106 create or alter function F2_CantidadJugadoresParam()
107 returns integer as

```

The bottom pane shows the execution results for the query 'jugadores_inscritos:int'. The results table has one row with the value 5.

jugadores_inscritos
5

The status bar at the bottom indicates '1 row retrieved starting from 1 in 532 ms (execution: 26 ms, fetching: 506 ms)'.

- 3.3. Crear una función que permita saber cuántos jugadores están inscritos y que sean de la categoría varones o mujeres.
- ■ La función debe llamarse F2_CantidadJugadoresParam()
- ■ La función debe recibir un parámetro “Varones” o “Mujeres”

The screenshot displays the SQL Server Enterprise Manager interface. The left pane shows the 'Database Explorer' with the 'Defensa_Hito_4_Grupo1' database selected. The right pane shows the SQL script editor with the following code:

```
select dbo.F1_CantidadJugadores() as jugadores_inscritos

--Crear una funcion que permita saber cuantos jugadores estan inscritos y que sean de la categoria varones o mujeres.
create or alter function F2_CantidadJugadoresParam()
returns integer as
begin
    declare @Cantidad_Jugadores integer;

    select @Cantidad_Jugadores = Count(*)
    from jugador as jug
    inner join equipo as equi on jug.id_equipo = equi.id_equipo
    where equi.categoria = 'Varones'
    return @Cantidad_Jugadores
end;

select dbo.F2_CantidadJugadoresParam() as Jugadores_Varones

--Crear una funcion que contenga el promedio de las edades mayores a
create or alter function F3_PromedioEdades()
```

The 'Jugadores_Varones' table is shown in the 'Output' pane with the following data:

Jugadores_Varones
3

The 'Services' pane at the bottom shows the execution of the script, with the 'console_1' session highlighted, indicating a successful execution.

- 3.4. Crear una función que obtenga el promedio de las edades mayores a una cierta edad.
- ■ La función debe llamarse F3_PromedioEdades()
- ■ La función debe recibir como parámetro 2 valores.
- ■ La categoría. (Varones o Mujeres)
- ■ La edad con la que se comparara (21 años ejemplo)
- ■ Es decir mostrar el promedio de edades que sean de una categoría y que sean mayores a 21 años.

```
117 end;
118
119 select dbo.F2_CantidadJugadoresParam() as Jugadores_Varones
120
121 --Crear una funcion que obtenga el promedio de las edades mayores a una cierta edad.
122 create or alter function F3_PromedioEdades()
123 returns integer as
124 begin
125     declare @Promedio_edades integer;
126
127     select @Promedio_edades = AVG(jug.edad)
128     from jugador as jug
129     inner join equipo as equi 1..n<>>1: on jug.id_equipo = equi.id_equipo
130     where jug.edad <21 and equi.categoria = 'Mujeres'
131     return @Promedio_edades
132
133 end;
134
135 select dbo.F3_PromedioEdades() as edad_and_categoria
136
137
```

Output: edad_and_categoria: int

edad_and_categoria
20

1 row retrieved starting from 1 in 99 ms (execution: 29 ms, fetching: 70 ms)

- 3.5. Crear una función que permita concatenar 3 parámetros.
- ■ La función debe llamarse F4_ConcatItems()
- ■ La función debe de recibir 3 parámetros.
- ■ La función debe de concatenar los 3 valores.
- ■ Para verificar la correcta creación de la función debe mostrar lo siguiente.
- ■ Mostrar los nombres de los jugadores, el nombre del equipo y la sede concatenada, utilizando la función que acaba de crear.

