



# Tarea HITO2

Docente: William Barra

Estudiante: Brandon Magne Aguilar

Gestión: 2022

Base de Datos II

Unifranz

Sede el Alto

Hito2

**GITHUB:** <https://github.com/branmag/Base-de-Datos-2>



## Manejo de conceptos.

- ¿A que se refiere cuando se habla de bases de datos relacionales?

Es una recopilación de elementos de datos con relaciones predefinidas entre ellos.

- ¿A que se refiere cuando se habla de bases de datos no relacionales?

Se caracterizan por tener una mayor escalabilidad y por soportar una estructura distribuida, son mas flexibles y permiten hacer cambios en los esquemas son para la base de datos.

- ¿Qué es MySQL y MariaDB?. Explique si existen diferencias o son iguales, etc.

MariaDB. - Es un sistema de gestión de base de datos.

MySQL. – Permite almacenar y acceder a los datos a través de multiples motores de almacenamiento.

MariaDB tiene licencia GPL mientras que MySQL tiene un enfoque de doble licencia, cada mango se acumula de una manera diferente. MariaDB soporta muchos motores de almacenamiento diferentes.



- ¿Qué son las funciones de agregación?

Es una función en la que los valores de varias filas se agrupan para formar un unico valor de resumen.

- ¿Qué llegaría a ser XAMPP, WAMP SERVER, o LAMP?

Xampp. - Es una distribución de Apache que incluye varios software libres.

Wamp Server. – El uso de Wamp permite subir paginas html a internet, ademas de poder gestionar datos en ellas.

Lamp. – Sistema operativo Linux, un servidor web Apache, una base de datos MySQL y lenguaje de Programación PHP.

¿Cuál es la diferencia entre las funciones de agregación y funciones creados por el DBA? Es decir funciones creadas por el usuario.

Las funciones de agregacion son las que ya vienen con la base de datos.

Las creadas por el usuario son las que necesitan ser creadas, estas pueden ser funciones de agregacion.

- ¿Para que sirve el comando USE?

Indica al usuario cual base de datos usar.

- ¿Qué es DML y DDL?

DDL. – Son sentencias utilizadas para la creación de base de datos, tablas, triggers, etc.

¿Qué cosas características debe tener una función? Explique sobre el nombre, el return, parámetros, etc.



- ¿Qué son las funciones de agregación?

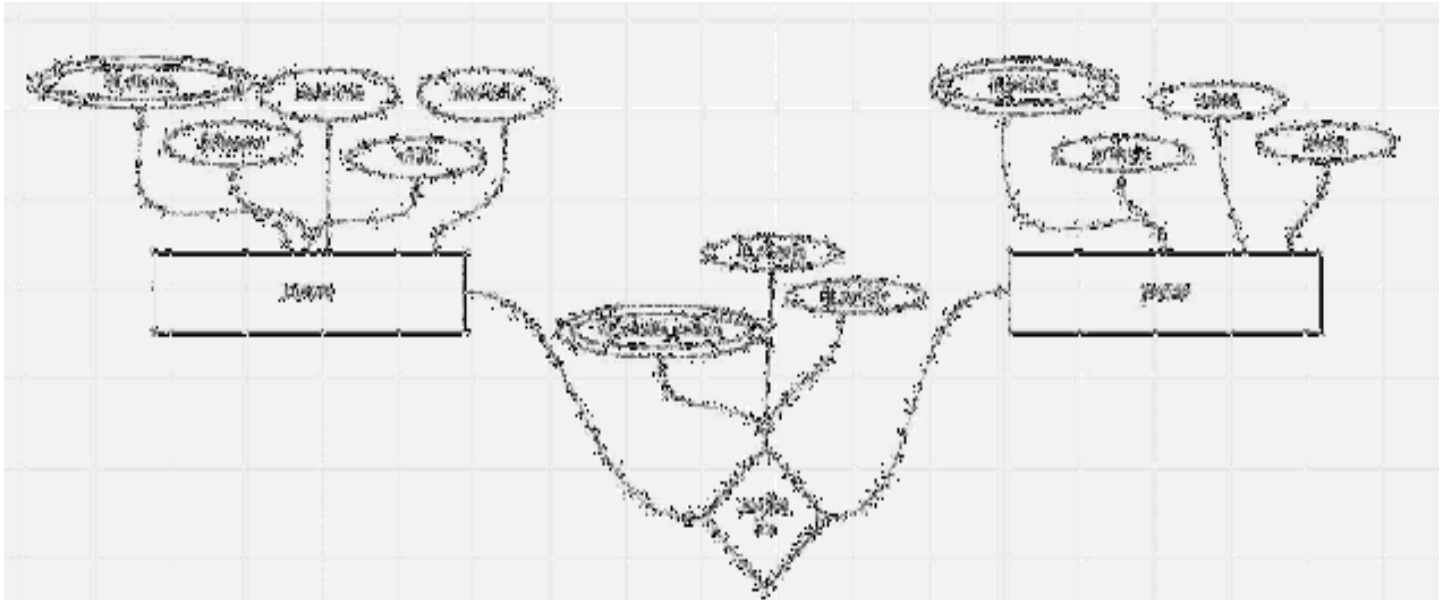
Es una función en la que los valores de varias filas se agrupan para formar un unico valor de resumen.

- ¿Cómo crear, modificar y como eliminar una función?
  - Para crear una función se pone el código: Create Function
  - Para modificar una función se pone el código: Replace
  - Para eliminar una función se pone el código: Drop Function.

## Parte practica

11. Crear las tablas y 2 registros para cada tabla para el siguiente modelo ER.

- Diseño.



- Se **sugiere** crear una base de datos de nombre **POLLOS\_COPA** y en ella crear las tablas:
  - cliente
  - detalle\_pedido
  - pedido

```

1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5  // 计算三角形面积
6  double areaTriangle(
7     double a, double b, double c
8 ) {
9     double s = (a + b + c) / 2;
10    double area = sqrt(s * (s - a) * (s - b) * (s - c));
11    return area;
12 }
13
14 // 计算圆的面积
15 double areaCircle(
16     double r
17 ) {
18     double area = 3.14159 * r * r;
19     return area;
20 }
21
22 // 计算圆的周长
23 double perimeterCircle(
24     double r
25 ) {
26     double perimeter = 2 * 3.14159 * r;
27     return perimeter;
28 }
29
30 int main() {
31     double a, b, c;
32     double r;
33
34     cout << "请输入三角形的三条边长: ";
35     cin >> a >> b >> c;
36
37     double area = areaTriangle(a, b, c);
38     cout << "三角形的面积为: " << area << endl;
39
40     cout << "请输入圆的半径: ";
41     cin >> r;
42
43     double areaCircle = areaCircle(r);
44     double perimeterCircle = perimeterCircle(r);
45     cout << "圆的面积为: " << areaCircle << endl;
46     cout << "圆的周长为: " << perimeterCircle << endl;
47
48     return 0;
49 }

```







The image displays three screenshots of a database client interface, showing SQL queries and their results for a database named 'pollos\_copa'.

**Top Screenshot: Querying the 'cliente' table**

The query is:

```
SELECT id_cliente, fullname, lastname, edad, domicilio FROM cliente
```

The results show two rows:

id_cliente	fullname	lastname	edad	domicilio
1	Miguel	Gonzalez	28	Baraltia
2	Juan	Condori	34	Fialta

**Middle Screenshot: Querying the 'pedido' table**

The query is:

```
SELECT id_pedido, articulo, costo, fecha FROM pedido
```

The results show two rows:

id_pedido	articulo	costo	fecha
1	Telescopio	3500bs	07/09/21
2	Almarna	1000bs	09/09/22

**Bottom Screenshot: Querying the 'detalle\_pedido' table**

The query is:

```
SELECT id_detalle_pedido, id_cliente, id_pedido FROM detalle_pedido
```

The results show two rows:

id_detalle_pedido	id_cliente	id_pedido
1	1	2
2	2	2

The screenshot shows an IDE window with a SQL query in the editor and its results in the Output pane.

**SQL Query:**

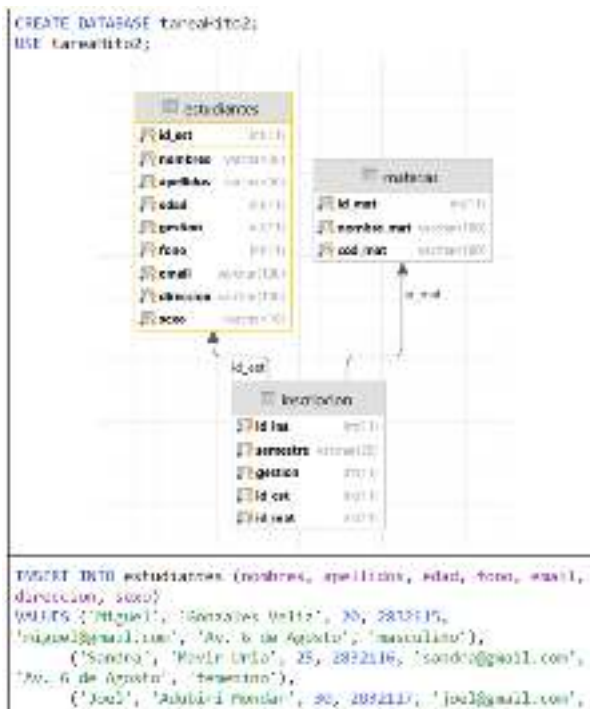
```
SELECT Detalle_Pedido.id_detalle_pedido as compra,
concat(C.fullname, C.lastname) as nombreCompleto, P.articulo, P.costo
from Detalle_Pedido
inner join pedido P on Detalle_Pedido.id_pedido = P.id_pedido
inner join cliente C ON Detalle_Pedido.id_cliente = C.id_cliente;
```

**Results:**

compra	nombreCompleto	articulo	costo
1	1 MiguelGonzales	Alnarra	1000bs
2	2 MiguelGonzales	Alnarra	1000bs

The IDE interface includes a menu bar (File, Edit, View, Navigate, Code, Refactor, Run, Tools, Git, Window, Help), a toolbar with icons for running and debugging, and a sidebar with Database Explorer, Services, and Bookmarks. The Services pane shows a list of database connections and their execution times.

- Crear un función que compare dos códigos de materia.
- o Recrear la siguiente base de datos:



```
'Av. 6 de Agosto', 'masculino'),
('Andrea', 'Arias Ballesteros', 21, 2032118,
'andrea@gmail.com', 'Av. 6 de Agosto', 'femenino'),
('Santos', 'Montes Valenzuela', 24, 2032119,
'santos@gmail.com', 'Av. 6 de Agosto', 'masculino');
```

```
INSERT INTO materias (nombre_mat, cod_mat)
VALUES ('Introduccion a la Arquitectura', 'ARQ-101'),
('Urbanismo y Diseño', 'ARQ-102'),
('Dibujo y Pintura Arquitectonica', 'ARQ-103'),
('Matematica discreta', 'ARO-104'),
('Fisica Basica', 'ARQ-105');
```

```
INSERT INTO inscripcion (id_est, id_mat, semestre, gestion)
VALUES (1, 1, '1er Semestre', 2018),
(1, 2, '2do Semestre', 2018),
(2, 4, '1er Semestre', 2019),
(2, 3, '2do Semestre', 2019),
(3, 3, '2do Semestre', 2020),
(3, 1, '3er Semestre', 2020),
(4, 4, '4to Semestre', 2021),
(5, 5, '5to Semestre', 2021);
```

- Resolver lo siguiente:
- ■ Mostrar los nombres y apellidos de los estudiantes inscritos en la materia ARQ-105, adicionalmente mostrar el nombre de la materia.
- ■ Deberá de crear una función que reciba dos parámetros y esta función deberá ser utilizada en la cláusula WHERE.

The screenshot shows an IDE with a SQL query editor and a results pane. The query is as follows:

```

SELECT est.id_est,
       est.nombre,
       est.apellidos,
       mat.nombre_mat,
       mat.cod_mat
FROM estudiantes AS est
INNER JOIN Inscripcion AS ins ON est.id_est = ins.id_est
INNER JOIN materias AS mat ON ins.id_mat = mat.id_mat
WHERE mat.cod_mat = 'ARQ-105';

```

The results pane shows the following data:

id_est	nombre	apellidos	nombre_mat	cod_mat
5	SANTOS	MONTES VALENZUELA	Fisica Basica	ARQ-105

- 14. Crear una función que permita obtener el promedio de las edades del género masculino o femenino de los estudiantes inscritos en la asignatura ARQ-104.
- o La función recibe como parámetro el género y el código de materia.

The screenshot shows an IDE with a dark theme. The top menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, Git, Window, and Help. The title bar reads 'DEF - console\_3 [localhost:3]'. The 'Database Explorer' on the left shows a tree with 'localhost [3]' and 'console\_3 [localhost:3]'. The main editor displays the following SQL code:

```
Create or replace function promedio(genero varchar(48), materia_cod varchar(48)) returns integer
begin
    declare promedio integer default 0;
    select avg(est.edad) into promedio
    from inscripcion
    inner join estudiantes as est on inscripcion.id_est = est.id_est
    inner join materias as mat on inscripcion.id_mat = mat.id_mat
    where est.sexo = genero and mat.cod_mat = materia_cod;
    return promedio;
end;

select promedio( genero 'Femenino', materia_cod 'ARQ-104');
```

The 'Services' panel on the left lists several services with their execution times. The 'Output' panel on the right shows the result of the function call:

```
'promedio('Femenino', 'ARQ-104')' = 23
```

The status bar at the bottom indicates '1 row retrieved starting from 1 in 36 ms (execution: 3 ms, fetching: 33 ms)' and '115:40 CRLF UTF-8 4 spaces'.

- 15. Crear una función que permita concatenar 3 cadenas.
- o La función recibe 3 parámetros.
- o Si las cadenas fuesen:
  - Pepito
  - Pep
  - 50
- o La salida debería ser: (Pepito), (Pep), (50)
- o La función creada utilizarlo en una consulta SQL.

```

-- 15. Crear una función que permita concatenar 3 cadenas.
-- 1. La función recibe 3 parámetros.
-- 2. Si las cadenas fuesen:
-- 3. ■ Pepito
-- 4. ■ Pep
-- 5. ■ 50
-- 6. La salida debería ser: (Pepito), (Pep), (50)
-- 7. La función creada utilizarlo en una consulta SQL.

-- 1. Crear la función concatenar_3_cadenas que recibe 3 parámetros: (p1), (p2), (p3)
-- 2. La función concatena los parámetros y devuelve el resultado.

CREATE FUNCTION concatenar_3_cadenas(@p1 VARCHAR(50), @p2 VARCHAR(50), @p3 VARCHAR(50)) RETURNS VARCHAR(150)
AS
BEGIN
    DECLARE @resultado VARCHAR(150);
    SET @resultado = CONCAT('(', @p1, '), (', @p2, '), (', @p3, ')');
    RETURN @resultado;
END;

-- 3. Consultar la función concatenar_3_cadenas con los parámetros: 'Pepito', 'Pep', '50'

SELECT concatenar_3_cadenas('Pepito', 'Pep', '50') AS resultado;

```

Query Editor

Query Results

1 (1 row(s) affected)





Database Console: console\_3 [localhost:3]

```
147 Create or replace function fullname(sexo varchar(50), edad integer) returns boolean
148 begin
149     declare suma integer default 0;
150     declare YesorNo boolean;
151     select sum(est.edad) into suma
152     from estudiantes as est
153     where est.sexo=sexo;
154     if suma %2=0 and suma>edad
155     then
156         set YesorNo = 1;
157     end if;
158     return YesorNo;
159 end;
160
161 select est.nombres, est.apellidos, est.edad, inscripcion.semestre
162 from inscripcion
163 inner join estudiantes est on inscripcion.id_est = est.id_est
164 where fullname(sexo: 'Masculino', edad: 23);
165
166 select sum(est.edad)
167 from estudiantes as est
168 group by (est.sexo);
```

Services

- localhost
  - console
- localhost [2]
  - default: 70 ms
  - console: 17 ms
  - console: 17 ms
- localhost [3]
  - default: 34 ms
  - detalle\_pedido: 116 ms
  - detalle\_pedido
  - console\_1: 10 ms
  - console\_1: 10 ms
  - console: 29 ms
  - console: 29 ms
  - 40: 0-01

Output: sum(est.edad): int(11)

	'sum(est.edad)' =
1	46
2	74

168.21 CRLF UTF-8 4 spaces



- 17. Crear una función de acuerdo a lo siguiente:
- o Crear una función sobre la tabla estudiantes que compara un nombre y apellidos. (si existe este nombre y apellido mostrar todos los datos del estudiante).
- La función devuelve un boolean.
- ■ La función debe recibir 4 parámetros, nombres y apellidos.
- ■ Similar al siguiente ejemplo.

