

**UNIVERSIDAD SAN BUENAVENTURA**

**PROCEDIMIENTOS DE ALMACENAMIENTOS  
EN BASES DE DATOS**

**TALLER 3**

**FUNCIÓN DE OPERACIÓN ÚTIL PARA EL PROYECTO  
INTEGRADOR FOCUS MIND**

**Estudiante**

**Nicolas Quiroz Gómez.**

**Profesor**

**MARY LUZ RUBIANO ACOSTA**

**Bogotá, 09 de marzo de 2025**

## Introducción

La función Obtener Resumen Usuario tiene como objetivo proporcionar un resumen claro y detallado de las actividades asignadas a un usuario en la base de datos del sistema Focus-Mind. Al ejecutar esta función, se obtiene el nombre del usuario, el número total de actividades planificadas y un listado de los nombres de dichas actividades.

## Objetivo de la Función

- Facilitar la obtención de un informe rápido sobre la planificación de cada usuario.
- Evitar consultas repetitivas a la base de datos al consolidar la información en una sola llamada.
- Mejorar la experiencia del usuario al visualizar de manera clara su agenda.

## Definición de la Función

sql

CopiarEditar

DELIMITER //

```
CREATE FUNCTION ObtenerResumenUsuario(  
    p_IdUsuario INT  
) RETURNS VARCHAR(500)  
DETERMINISTIC  
BEGIN  
    DECLARE resumen VARCHAR(500);  
  
    SELECT CONCAT('Usuario: ', P.Nombre, ' - Total Actividades: ',  
COUNT(A.IdActividades),  
        ' - Actividades: ', GROUP_CONCAT(A.Nombre SEPARATOR ', '))  
    INTO resumen  
    FROM Personas P  
    LEFT JOIN Planificador PL ON P.IdPersonas = PL.IdUsuario
```

```
LEFT JOIN Actividades A ON PL.IdActividad = A.IdActividades

WHERE P.IdPersonas = p_IdUsuario

GROUP BY P.IdPersonas;

RETURN IFNULL(resumen, 'Usuario no encontrado o sin actividades');

END //
```

DELIMITER ;

### **Explicación del Código**

#### **1. Parámetro de Entrada:**

- p\_IdUsuario INT: Se recibe el ID del usuario como argumento para filtrar los datos y obtener su resumen.

#### **2. Declaración de Variable:**

- resumen VARCHAR(500): Se usa para almacenar el texto que contendrá el nombre del usuario, el número de actividades y la lista de actividades.

#### **3. Consulta SQL Principal:**

- Se usa SELECT CONCAT(...) para construir una cadena de texto con Usuario: [Nombre] - Total Actividades: [Cantidad] - Actividades: [Lista de Actividades]
- COUNT(A.IdActividades): Cuenta la cantidad de actividades que tiene el usuario en la tabla Planificador.
- GROUP\_CONCAT(A.Nombre SEPARATOR ', '): Genera una lista con los nombres de las actividades separadas por comas.
- Se usa LEFT JOIN para garantizar que incluso los usuarios sin actividades aparezcan en la consulta.

#### **4. Filtro de Usuario:**

- WHERE P.IdPersonas = p\_IdUsuario: Se filtra para obtener información solo del usuario especificado.

#### **5. Agrupación de Datos:**

- GROUP BY P.IdPersonas: Se agrupan los datos por usuario para evitar resultados duplicados.

## 6. Manejo de Valores Nulos:

- IFNULL (resumen, 'Usuario no encontrado o sin actividades'): Si el usuario no tiene actividades, se devuelve este mensaje.

### Ejemplo de Uso

Para consultar el resumen de actividades del usuario con ID 1, ejecuta:

SQL

```
SELECT ObtenerResumenUsuario(1) AS Resumen;
```

### Ejemplo de Salida

Si el usuario 1 (Juan Pérez) tiene actividades planificadas, la función devolverá:

yaml

CopiarEditar

Usuario: Juan Pérez - Total Actividades: 2 - Actividades: Correr en el parque,  
Estudiar para el examen

Si el usuario no tiene actividades registradas, mostrará:

Usuario no encontrado o sin actividades

### Conclusión

La función ObtenerResumenUsuario optimiza la consulta de información relevante para cada usuario, consolidando en una sola consulta datos esenciales sobre su planificación. Esto mejora la eficiencia del sistema Focus\_Mind y facilita la experiencia del usuario al visualizar su agenda de manera estructurada y comprensible.

Imágenes de uso en base de datos.

```
Query 1  SQL File 1*  SQL File 2*  SQL File 3*  SQL File 6*  SQL File 7*  SQL File 8*  SQL File 10*  SQL File 11*
Limit to 1000 rows
1 DELIMITER //
2
3 CREATE FUNCTION ObtenerResumenUsuario(
4     p_IdUsuario INT
5 ) RETURNS VARCHAR(500)
6 DETERMINISTIC
7 BEGIN
8     DECLARE resumen VARCHAR(500);
9
10    SELECT CONCAT('Usuario: ', P.Nombre, ' - Total Actividades: ', COUNT(A.IdActividades),
11        ' - Actividades: ', GROUP_CONCAT(A.Nombre SEPARATOR ', '))
12    INTO resumen
13    FROM Personas P
14    LEFT JOIN Planificador PL ON P.IdPersonas = PL.IdUsuario
15    LEFT JOIN Actividades A ON PL.IdActividad = A.IdActividades
16    WHERE P.IdPersonas = p_IdUsuario
17    GROUP BY P.IdPersonas;
18
19    RETURN IFNULL(resumen, 'Usuario no encontrado o sin actividades');
20 END //
21
22 DELIMITER ;
23
```

Query 1 SQL File 1\* SQL File 2\* SQL File 3\* SQL File 6\* SQL File 7\* SQL File 8\* SQL File 10\* SQL File 11\* x

Limit to 1000 rows

1 • SELECT ObtenerResumenUsuario(1) AS Resumen;

2

Result Grid Filter Rows: Export: Wrap Cell Content: I

Resumen
Usuario: Juan Perez - Total Actividades: 1 - Actividades: Correr en el parque