

“Año de la recuperación y consolidación de la economía peruana”

UNIVERSIDAD PERUANA LOS ANDES

FACULTAD DE INGENIERÍA”

ESCUELA PROFESIONAL “INGENIERÍA DE SISTEMAS Y COMPUTACIÓN”

Cátedra: Base de datos II

Catedrático: Fernandez Bejarano Raul Enrique

Estudiante: Egoavil Machado Jesus Miguel

HUANCAYO PERÚ

Módulo 5: Consultas Avanzadas en SQL Server

Contenido estructurado por sección

Consultas de agregación

- Funciones de agregación (SUM, COUNT, AVG, MIN, MAX)

```
-- SUM: Total cantidad de productos vendidos
SELECT SUM(Cantidad) AS TotalVendidos FROM Ventas;

-- COUNT: Cuántas ventas se hicieron
SELECT COUNT(*) AS NumeroVentas FROM Ventas;

-- AVG: Precio promedio de los productos
SELECT AVG(Precio) AS PrecioPromedio FROM Productos;

-- MIN y MAX: Precio mínimo y máximo
SELECT MIN(Precio) AS PrecioMinimo, MAX(Precio) AS PrecioMaximo FROM Productos;
```

- Agrupación de datos

```
-- Agrupación de datos
SELECT Categoria, SUM(Precio) AS SumaPrecios
FROM Productos
GROUP BY Categoria;
```

- Subconsultas

```
-- Subconsulta: Productos más caros que el promedio
SELECT * FROM Productos
WHERE Precio > (SELECT AVG(Precio) FROM Productos);
```

Consultas de unión

- Tipos de uniones (INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN)

```
-- INNER JOIN: Productos vendidos y su cantidad
SELECT P.Nombre, V.Cantidad
FROM Productos P
INNER JOIN Ventas V ON P.IdProducto = V.IdProducto;

-- LEFT JOIN: Todos los productos y sus ventas (si las tienen)
SELECT P.Nombre, V.Cantidad
FROM Productos P
LEFT JOIN Ventas V ON P.IdProducto = V.IdProducto;

-- RIGHT JOIN: Todas las ventas y sus productos
SELECT P.Nombre, V.Cantidad
FROM Productos P
RIGHT JOIN Ventas V ON P.IdProducto = V.IdProducto;

-- FULL JOIN: Todos los productos y todas las ventas (incluso si no coinciden)
SELECT P.Nombre, V.Cantidad
FROM Productos P
FULL JOIN Ventas V ON P.IdProducto = V.IdProducto;
```

- Múltiples uniones
- Subconsultas en uniones

```
-- Subconsulta en uniones: Productos vendidos más de 1 vez
SELECT P.Nombre, V.Cantidad
FROM Productos P
INNER JOIN (
    SELECT IdProducto, SUM(Cantidad) AS Cantidad
    FROM Ventas
    GROUP BY IdProducto
    HAVING SUM(Cantidad) > 1
) V ON P.IdProducto = V.IdProducto;
```

Consultas con funciones de ventana

- Funciones OVER

```
-- SUM sobre ventana
SELECT IdProducto, Cantidad,
       SUM(Cantidad) OVER() AS TotalGlobal
FROM Ventas;
```

- Particiones de ventana

```
-- Partición de ventana: suma por producto
SELECT IdProducto, Cantidad,
       SUM(Cantidad) OVER(PARTITION BY IdProducto) AS TotalPorProducto
FROM Ventas;
```

- Marcos de ventana

```
-- Marcos de ventana: acumulado por fecha
SELECT FechaVenta, Cantidad,
       SUM(Cantidad) OVER(ORDER BY FechaVenta ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS Acumulado
FROM Ventas;
```