

# Base de Datos Northwind (Adaptada para PostgreSQL)

---

Este proyecto incluye una versión adaptada del conocido conjunto de datos Northwind, especialmente preparada para funcionar con PostgreSQL. Se trata de una base de datos que simula el funcionamiento de una empresa distribuidora de alimentos, con información detallada sobre clientes, productos, pedidos, empleados, envíos y otros aspectos clave del negocio.

## Funcionalidades Añadidas

---

Organización Jerárquica de Categorías: Se incorporaron subcategorías para lograr una estructura más ordenada y fácil de navegar.

## Tecnologías Utilizadas

---

PostgreSQL 12 o superior como sistema de gestión de bases de datos  
pgAdmin para administración y visualización  
Archivo SQL Dump incluido para facilitar una instalación rápida

## Estructura del Repositorio

---

```
northwind-postgres-modificado/  
├── README.md – Información general  
├── northwind_modificado.sql – ★ Dump completo de la BD  
├── docs/ – Guías de instalación, mejoras y ejemplos  
└── img/ – Diagrama ER y capturas de consultas
```

## Instalación Rápida

---

Requisitos Previos: PostgreSQL 12+ Cliente psql o pgAdmin

## Instalación en 3 pasos

---

1. clonar repositorio

```
git clone https://github.com/Tarik404/UF1472_Tarek.git  
cd UF1472_Tarek
```

## 2. crear base de datos

```
createdb northwind_01 -U tu_usuario_postgre
```

## 3. Restaurar dump completo

```
psql -U tu_usuario -d northwind_01 -f dump_northwind.sql
```

¡Y listo! La base de datos estará completamente configurada con datos de ejemplo.



# Vistas SQL del Proyecto Northwind (PostgreSQL)

Object Explorer

- > 1..3 Sequences
- > Tables
- > Trigger Functions
- > Types
- ▼ Views (18)
  - > vw\_analisis\_clientes
  - > vw\_clientes\_frecuentes
  - > vw\_estado\_inventario
  - > vw\_ordenes\_ciudad
  - > vw\_ordenes\_detalladas
  - > vw\_performance\_categorias
  - > vw\_performance\_proveedores
  - > vw\_precios\_categoria
  - > vw\_productos\_mas\_vendidos
  - > vw\_stock\_bajo
  - > vw\_top\_clientes
  - > vw\_top\_productos
  - > vw\_ventas\_categoria
  - > vw\_ventas\_diarias
  - > vw\_ventas\_empleado
  - > vw\_ventas\_mensuales
  - > vw\_ventas\_mes
  - > vw\_ventas\_pais

## ☑ Verificar vistas en PostgreSQL

---

```
SELECT * FROM vw_productos_stock LIMIT 5;
SELECT * FROM vw_clientes_frecuentes LIMIT 5;
SELECT * FROM vw_empleados_ventas LIMIT 5;
SELECT * FROM vw_total_ventas_por_pais LIMIT 5;
SELECT * FROM vw_top_productos LIMIT 5;
SELECT * FROM vw_envios_pendientes LIMIT 5;
```

## 🚀 Instalación de la Base de Datos Northwind\_curso en PostgreSQL desde la Terminal

---

```
C:\Users\ademo>psql -U postgres -d postgres
Contraseña para usuario postgres:

psql (17.4)
ADVERTENCIA: El código de página de la consola (850) difiere del código
de página de Windows (1252).
Los caracteres de 8 bits pueden funcionar incorrectamente.
Vea la página de referencia de psql «Notes for Windows users»
para obtener más detalles.
Digite «help» para obtener ayuda.

postgres=# createdb northwind_curso
postgres=# psql -d northwind_curso -f northwind_modificado.sql
postgres=# \q

C:\Users\ademo>createdb -U postgres northwind_curso
Contraseña:
```

## 📌 Realizar una copia de seguridad completa (dump) de la base de datos Northwind y almacenarla en un archivo.

---

```
pg_dump -U postgres -d northwind -f "C:\Users\ademo\Desktop\git\SQL-POSTGRES-
Northwind.sql"
```

## 💰 Ventas totales por empleado

---

```
SELECT
    e.first_name || ' ' || e.last_name AS empleado,
    COUNT(o.order_id) AS total_pedidos
FROM employees e
JOIN orders o ON e.employee_id = o.employee_id
GROUP BY empleado
ORDER BY total_pedidos DESC;
```



## Pedidos realizados en un año específico

---

```
SELECT order_id, order_date, customer_id, ship_country
FROM orders
WHERE EXTRACT(YEAR FROM order_date) = 1997
ORDER BY order_date;
```



## Días promedio de entrega por país

---

```
SELECT
    ship_country,
    ROUND(AVG(ship_date - order_date)) AS promedio_dias_envio
FROM orders
WHERE ship_date IS NOT NULL
GROUP BY ship_country
ORDER BY promedio_dias_envio;
```

## 4 Análisis creativo: ¿Quiénes son nuestros "clientes VIP"?

---

```
SELECT c.company_name, COUNT(o.order_id) AS total_pedidos
FROM customers c
JOIN orders o ON c.customer_id = o.customer_id
GROUP BY c.company_name
ORDER BY total_pedidos DESC
LIMIT 5;
```



## Validación del Entorno y Extensiones

---


Después de restaurar la base de datos Northwind y aplicar las extensiones personalizadas, se realizaron una serie de pruebas de verificación para confirmar el correcto funcionamiento de las tablas, vistas y funciones añadidas.







```
-- Validar tabla de regiones comerciales
SELECT COUNT(*) FROM regiones_comerciales;      -- Esperado: 5 regiones

-- Validar tabla de descuentos promocionales
SELECT COUNT(*) FROM descuentos_promocionales; -- Esperado: 12 registros activos





-- Validar tabla de niveles de stock
SELECT COUNT(*) FROM niveles_stock_critico;      -- Debe mostrar productos con stock bajo
```

## Detalles Técnicos del Proyecto Northwind+

 Resumen del entorno:

Característica	Valor
 PostgreSQL versión	12 o superior
 Tamaño del dump	~500 KB
 Tablas totales	17 (13 estándar + 4 ampliadas)
 Vistas integradas	4 (para consultas rápidas)
 Funciones definidas	3 (lógica de negocio y cálculo)
 Triggers activos	1 (gestión de stock automatizada)

## Extras implementados:

-  Subcategorías de productos
-  Vista de productos con bajo stock
-  Vista mensual de ventas por empleado
-  Función para calcular descuentos por volumen

## Habilidades y Logros Técnicos Desarrollados

- ✓ Ajuste y ampliación de esquemas de base de datos existentes
- ✓ Diseño y creación de tablas interrelacionadas para modelado avanzado
- ✓ Implementación de triggers automáticos para mantenimiento y control
- ✓ Construcción de vistas sofisticadas para análisis eficiente de datos
- ✓ Programación de funciones en PL/pgSQL para lógica personalizada
- ✓ Mejora del rendimiento mediante índices estratégicos

✓ Configuración de sistemas de auditoría para seguimiento de cambios ✓ Creación y manejo de copias de seguridad mediante dumps automatizados



## Conclusión y Próximos Pasos

---

Gracias por acompañarme en este recorrido por la base de datos Northwind y sus extensiones personalizadas. Hemos explorado desde la estructura básica hasta análisis avanzados, optimizaciones y automatizaciones que hacen de esta base un recurso poderoso para la toma de decisiones y el aprendizaje práctico.

Este proyecto no solo demuestra el manejo de PostgreSQL a nivel avanzado, sino también la capacidad de transformar datos en información valiosa para el negocio.