

INSTITUCIÓN UNIVERSITARIA DIGITAL DE ANTIOQUIA
Bases de Datos II

Profesor: Víctor Hugo Mercado Alumno: Johan
Sebastián Zamudio Cortes

Ingeniería de desarrollo de software y datos
2025

Contenido

Objetivo General.....	3
Objetivos específicos	4
PLAN DE PRUEBAS	5
ANÁLISIS DE DISTRIBUCIÓN Y TENDENCIAS	6
VERIFICACIÓN DE FORMATOS DE DATOS.....	7

Objetivo General

Recopilar todas aquellas actividades previas en un solo documento, así como generar un plan de pruebas que nos ayude a verificar la integridad de los datos, documentando el proceso realizado para poder llevar a cabo esta actividad

Objetivos específicos

- Asegurarse de que todos los campos obligatorios en las tablas contengan información válida es fundamental.
- Es importante identificar los registros que presenten valores nulos o faltantes y documentar adecuadamente los hallazgos.
- Además, se debe llevar a cabo la identificación de registros duplicados en las tablas utilizando consultas SQL y aplicando las reglas de negocio correspondientes.
- Se implementarán estrategias que faciliten la eliminación o fusión de registros redundantes.
- Recopilarán todas las actividades realizadas en un único documento, en el que se detallará cada prueba efectuada.
- Detectar posibles discrepancias en nombres, códigos o identificadores.

PLAN DE PRUEBAS

El primer paso del plan de pruebas es realizar una verificación de todos los datos de la tabla de jardinería para verificar su estado, así como la integridad de los mismos, para eso realizaremos una serie de consultas que nos permitan verificar la integridad de los mismos

Los datos incompletos pueden afectar los análisis. Si el resultado muestra valores mayores a 0 hay registros incompletos que requieren limpieza.

```
1 SELECT 'Clientes' AS Tabla, COUNT(*) AS Registros_Incompletos
2 FROM staging.stg_cliente
3 WHERE nombre_cliente IS NULL OR telefono IS NULL;
4
```

Detectar valores inconsistentes o fuera de rango, esto verifica la integración de los datos y que los valores sean coherentes

```
1 SELECT ID_producto, nombre, precio_venta
2 FROM staging.stg_producto
3 WHERE precio_venta < 0;
4
```

Verificación de claves primarias por posible duplicidad en los datos de la misma Si hay resultados, significa que hay clientes duplicados y deben consolidarse.

```
1 SELECT ID_cliente, COUNT(*) AS Repeticiones
2 FROM staging.stg_cliente
3 GROUP BY ID_cliente
4 HAVING COUNT(*) > 1;
```

Validar relaciones entre tablas Si hay pedidos sin cliente asociado, hay problemas de integridad referencial.

```
1 SELECT p.ID_pedido, p.ID_cliente
2 FROM staging.stg_pedido p
3 LEFT JOIN staging.stg_cliente c ON p.ID_cliente = c.ID_cliente
4 WHERE c.ID_cliente IS NULL;
5
```

Comprobar formatos incorrectos Si aparecen registros, hay valores inválidos que deben corregirse.

```
1 SELECT ID_cliente, telefono
2 FROM staging.stg_cliente
3 WHERE telefono NOT REGEXP '^([0-9])+$';
4
```

Validar Fechas Incorrectas encontrar fechas incoherentes o en el futuro/pasado no lógico

```
1 SELECT * FROM staging.stg_pedido
2 WHERE fecha_pedido > CURDATE();
3
```

Medir la Densidad de Datos evaluar qué porcentaje de datos es nulo o incompleto.

```
SELECT
    (COUNT(*) - COUNT(nombre_cliente)) / COUNT(*) * 100 AS Porcentaje_Nulos
FROM staging.stg_cliente;
```

ANÁLISIS DE DISTRIBUCIÓN Y TENDENCIAS

El segundo paso es verificar que los datos sigan una distribución lógica y detectar valores atípicos o inconsistencias, en las tablas de la base de datos

```
1 SELECT
2     MIN(limite_credito) AS Minimo,
3     MAX(limite_credito) AS Maximo,
4     AVG(limite_credito) AS Promedio,
5     STDDEV(limite_credito) AS Desviacion_Estandar
6 FROM staging.stg_cliente;
```

Análisis de Frecuencia de Valores Categóricos revisar cuántos clientes pertenecen a cada país.

```
1 SELECT pais, COUNT(*) AS cantidad
2 FROM staging.stg_cliente
3 GROUP BY pais
4 ORDER BY cantidad DESC;
```

Verificar Consistencia de Datos en el Tiempo ver cuántos pedidos se hacen cada mes y detectar anomalías.

```
1 SELECT YEAR(fecha_pedido) AS Año, MONTH(fecha_pedido) AS Mes, COUNT(*) AS Total_Pedidos
2 FROM staging.stg_pedido
3 GROUP BY Año, Mes
4 ORDER BY Año DESC, Mes DESC;
5
```

VERIFICACIÓN DE FORMATOS DE DATOS

Comprobar que los datos cumplen con los formatos esperados validar que los códigos postales solo contengan números y tengan la longitud correcta.

```
1 SELECT codigo_postal
2 FROM staging.stg_cliente
3 WHERE codigo_postal NOT REGEXP '^[0-9]{5,10}$';
4
```

Comprobar que los correos electrónicos sean válidos, y contengan caracteres que contengan los caracteres correctos

```
1 SELECT email
2 FROM staging.stg_cliente
3 WHERE email NOT REGEXP '^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$';
4
```

Asegurar que las fechas sean lógicas y no contengan valores erróneos buscar pedidos con fechas imposibles.

```
1 SELECT * FROM staging.stg_pedido
2 WHERE fecha_pedido < '2000-01-01' OR fecha_pedido > CURDATE();
3
```