

---

-- Data Inspection and Validation Checks

-- Checking for Nulls in Important Fields

-- 1.1 Checking for NULL values in essential fields of the sales table

SELECT

```
    COUNT(*) AS Null_Sale_Dates
FROM sales
WHERE sale_date IS NULL;
```

SELECT

```
    COUNT(*) AS Null_Product_IDs
FROM sales
WHERE product_id IS NULL;
```

SELECT

```
    COUNT(*) AS Null_Rep_IDs
FROM sales
WHERE rep_id IS NULL;
```

SELECT

```
    COUNT(*) AS Null_Quantities
FROM sales
WHERE quantity_sold IS NULL;
```

GO

-- Checking for Duplicates

-- 2.1 Checking for duplicate sales records based on important fields

SELECT

```
    sale_date,
    product_id,
    rep_id,
    customer_id,
    COUNT(*) AS record_count
FROM sales
GROUP BY
    sale_date,
    product_id,
    rep_id,
    customer_id
HAVING COUNT(*) > 1;
GO
```

-- Checking for Invalid Foreign Key References

-- 3.1 Checking for sales records with invalid product IDs

SELECT

```
    sale_id, product_id
FROM sales
WHERE product_id NOT IN (SELECT product_id FROM products);
```

-- 3.2 Checking for sales records with invalid sales representative IDs

SELECT

sale\_id, rep\_id

FROM sales

WHERE rep\_id NOT IN (SELECT rep\_id FROM sales\_representatives);

-- 3.3 Checking for sales records with invalid customer IDs

SELECT

sale\_id, customer\_id

FROM sales

WHERE customer\_id IS NOT NULL

AND customer\_id NOT IN (SELECT customer\_id FROM customers);

-- 3.4 Checking for sales records with invalid payment method IDs

SELECT

sale\_id, payment\_method\_id

FROM sales

WHERE payment\_method\_id IS NOT NULL

AND payment\_method\_id NOT IN (SELECT payment\_method\_id FROM payment\_methods);

GO

-- Checking Logical Consistency of Values

-- 4.1 Checking for negative quantity sold

SELECT \*

FROM sales

WHERE quantity\_sold < 0;

-- 4.2 Checking for zero or negative unit prices

SELECT \*

FROM sales

WHERE unit\_price <= 0;

-- 4.3 Checking for zero or negative total sales

SELECT \*

FROM sales

WHERE total\_sales <= 0;

-- 4.4 Checking if total\_sales = quantity\_sold \* unit\_price (Allowing small margin for floating-point errors)

SELECT \*

FROM sales

WHERE ABS((quantity\_sold \* unit\_price) - total\_sales) > 0.01;

GO

-- Checking Date Fields for Anomalies

-- 5.1 Check if there are any future sale dates

SELECT \*

FROM sales

---

```
WHERE sale_date > GETDATE();
```

```
-- 5.2 Check for extremely old sale dates (before 2000)
```

```
SELECT *  
FROM sales  
WHERE sale_date < '2023-01-01';
```

```
GO
```

```
-- Checking for Inconsistent Category Assignment
```

```
-- 6.1 List products with missing or wrong category_id
```

```
SELECT  
    product_id,  
    product_name,  
    category_id  
FROM products  
WHERE category_id NOT IN (SELECT category_id FROM categories);
```

```
GO
```

```
-- Checking if Rep-Region Relationship is Valid
```

```
-- 7.1 Validate that every sales representative is assigned to a valid region
```

```
SELECT  
    rep_id,  
    rep_name,  
    region_id  
FROM sales_representatives  
WHERE region_id NOT IN (SELECT region_id FROM regions);
```