SQL Queries for Summer Sales Analysis

-- 1. Database Setup and Preparation

```
-- Select the Summer SALES database and query the summer table
USE Summer_SALES;
SELECT * FROM summer;

-- Create a copy of the summer table to preserve original data
SELECT * INTO summer_data FROM summer;
SELECT * FROM summer_data;
```

-- 2. Data Type Verification

```
-- Examine the structure of the summer_data table
EXEC sp_help 'summer_data';
```

-- 3. Distinct Analysis

```
-- Identify potential duplicates by Product_ID and Product_Name
SELECT Product_ID, Product_Name, COUNT(*) AS Count
FROM summer_data
GROUP BY Product_ID, Product_Name
HAVING COUNT(*) > 1
ORDER BY Count DESC;
```

-- 4. Brand Count Analysis

```
-- Count distinct brands and total brand occurrences
SELECT Brand_Name, COUNT(*) AS BrandCount
FROM summer_data
GROUP BY Brand_Name
```

```
ORDER BY BrandCount DESC;

-- Count distinct Brand_Name values
SELECT COUNT(DISTINCT Brand_Name) AS TotalDistinctBrandCount
FROM summer_data;

-- 5. Sales Performance and Trends
-- Calculate total units sold and total revenue
SELECT SUM(Quantity) AS Total_Units_Sold, SUM(Revenue) AS Total_Revenue
FROM summer_data;

-- 6. Top-Performing Products
-- Rank products by total revenue and quantity sold
SELECT Product_Name, SUM(Quantity) AS Total_Quantity, SUM(Revenue) AS Total_Revenue
FROM summer_data
GROUP BY Product_Name
ORDER BY Total_Revenue DESC;

-- 7. Top Repeated Purchase Items
-- Identify frequently purchased items
SELECT Product_Name, COUNT(*) AS Purchase_Frequency
FROM summer_data
GROUP BY Product_Name
ORDER BY Purchase_Frequency DESC;

-- 8. Top-Performing Brands
-- Rank brands by revenue and count of unique products
SELECT Brand_Name, SUM(Revenue) AS Total_Revenue, COUNT(DISTINCT Product_ID) AS
Unique_Products
FROM summer_data
```

```sql
GROUP BY Brand_Name
ORDER BY Total_Revenue DESC;
```

-- 9. Average Revenue per Transaction by Category

```sql
-- Calculate average revenue per transaction by category
SELECT Category, AVG(Revenue) AS Avg_Revenue
FROM summer_data
GROUP BY Category
ORDER BY Avg_Revenue DESC;
```

-- 10. Time-Based Insights

```sql
-- Sales by hour and shift
SELECT Hour_Shift, SUM(Revenue) AS Total_Revenue
FROM summer_data
GROUP BY Hour_Shift
ORDER BY Hour_Shift;

-- Daily sales trends
SELECT Day, SUM(Revenue) AS Total_Revenue, SUM(Quantity) AS Total_Units
FROM summer_data
GROUP BY Day
ORDER BY Total_Revenue DESC;

-- Time-based peak sales (weekends vs. weekdays)
SELECT Shift_Weekend_or_Weekdays, SUM(Revenue) AS Total_Revenue
FROM summer_data
GROUP BY Shift_Weekend_or_Weekdays
ORDER BY Total_Revenue DESC;
```

```
-- Revenue by month

SELECT FORMAT(Date, 'yyyy-MM') AS Month, SUM(Revenue) AS Total_Revenue, SUM(Quantity) AS Total_Units

FROM summer_data

GROUP BY FORMAT(Date, 'yyyy-MM')

ORDER BY Month;
```

-- 11. Sales by Price Category

```
-- Analyze sales by price category, including total revenue and units sold

SELECT Price_Category, SUM(Revenue) AS Total_Revenue, SUM(Quantity) AS Total_Units_Sold

FROM summer_data

GROUP BY Price_Category

ORDER BY Total_Revenue DESC;
```