

```
Create database Retail_and_Warehouse_sales;
DROP database Retail_and_Warehouse_sales;
USE Retail_and_Warehouse_sales;
```

```
CREATE TABLE retail (
    YEAR INT,
    MONTH INT,
    SUPPLIER VARCHAR(255),
    ITEM_CODE VARCHAR(50),
    ITEM_DESCRIPTION VARCHAR(255),
    ITEM_TYPE VARCHAR(50),
    RETAIL_SALES DECIMAL(10, 2),
    RETAIL_TRANSFERS DECIMAL(10, 2)
);
```

```
CREATE TABLE warehouse (
    YEAR INT,
    MONTH INT,
    SUPPLIER VARCHAR(255),
    ITEM_CODE VARCHAR(50),
    ITEM_DESCRIPTION VARCHAR(255),
    ITEM_TYPE VARCHAR(50),
    WAREHOUSE_SALES DECIMAL(10, 2)
);
```

```
SELECT * FROM retail;
SELECT * FROM warehouse;
```

Q1:) Retrieve all retail sales for a specific item in a given year and month:

```
SELECT YEAR, MONTH, ITEM_CODE, ITEM_DESCRIPTION, RETAIL_SALES
FROM retail
WHERE YEAR = 2020
    AND MONTH = 5
    AND ITEM_CODE = '103608';
```

Q2:) Calculate the total retail sales for each item type in a specific year:

```
SELECT YEAR, ITEM_TYPE, SUM(RETAIL_SALES) AS TOTAL_RETAIL_SALES
FROM retail
WHERE YEAR = 2019
GROUP BY YEAR, ITEM_TYPE;
```

Q3:) Find the top 5 suppliers by total warehouse sales in a particular month:

```
SELECT MONTH, SUPPLIER, SUM(WAREHOUSE_SALES) AS TOTAL_WAREHOUSE_SALES
FROM warehouse
WHERE MONTH = 8
GROUP BY MONTH, SUPPLIER
ORDER BY TOTAL_WAREHOUSE_SALES DESC
LIMIT 5;
```

Q4:) Calculate the total retail transfers for a specific supplier across all years:

```
SELECT SUPPLIER, SUM(RETAIL_TRANSFERS) AS TOTAL_RETAIL_TRANSFERS
FROM retail
WHERE SUPPLIER = 'REPUBLIC NATIONAL DISTRIBUTING CO'
GROUP BY SUPPLIER;
```

Q5:) Identify the items with no retail sales in a given month:

```
SELECT DISTINCT r.ITEM_CODE, r.ITEM_DESCRIPTION
FROM retail r
LEFT JOIN warehouse w ON r.ITEM_CODE = w.ITEM_CODE AND r.YEAR = w.YEAR AND
r.MONTH = w.MONTH
WHERE r.YEAR = 20219
      AND r.MONTH = 10
      AND w.ITEM_CODE IS NULL;
```

Q6:) Calculate the total warehouse sales for each item type in a specific year and month:

```
SELECT YEAR, MONTH, ITEM_TYPE, SUM(WAREHOUSE_SALES) AS
TOTAL_WAREHOUSE_SALES
FROM warehouse
WHERE YEAR = 2020
      AND MONTH = 6
GROUP BY YEAR, MONTH, ITEM_TYPE;
```

Q7:) Find the average retail sales for each item type across all years:

```
SELECT ITEM_TYPE, AVG(RETAIL_SALES) AS AVERAGE_RETAIL_SALES
FROM retail
GROUP BY ITEM_TYPE;
```

Q8:) Retrieve the top 3 items with the highest retail sales in a specific month:

```
SELECT YEAR, MONTH, ITEM_CODE, ITEM_DESCRIPTION, RETAIL_SALES
FROM retail
WHERE YEAR = 2018
      AND MONTH = 7
ORDER BY RETAIL_SALES DESC
LIMIT 3;
```

Q9:) Find the total retail sales and warehouse sales for a specific item across all years:

```
SELECT r.ITEM_CODE, r.ITEM_DESCRIPTION, SUM(r.RETAIL_SALES) AS
TOTAL_RETAIL_SALES, SUM(w.WAREHOUSE_SALES) AS TOTAL_WAREHOUSE_SALES
FROM retail r
LEFT JOIN warehouse w ON r.ITEM_CODE = w.ITEM_CODE
WHERE r.ITEM_CODE = '100145'
```

```
GROUP BY r.ITEM_CODE, r.ITEM_DESCRIPTION;
```

Q10:) Calculate the total retail transfers and warehouse sales for each supplier in a specific year:

```
SELECT r.SUPPLIER, SUM(r.RETAIL_TRANSFERS) AS TOTAL_RETAIL_TRANSFERS,
SUM(w.WAREHOUSE_SALES) AS TOTAL_WAREHOUSE_SALES
FROM retail r
LEFT JOIN warehouse w ON r.SUPPLIER = w.SUPPLIER AND r.YEAR = w.YEAR
WHERE r.YEAR = 2019
GROUP BY r.SUPPLIER;
```

Q11:) Retrieve all items with retail sales greater than warehouse sales in a specific month:

```
SELECT r.YEAR, r.MONTH, r.ITEM_CODE, r.ITEM_DESCRIPTION, r.RETAIL_SALES,
w.WAREHOUSE_SALES
FROM retail r
JOIN warehouse w ON r.ITEM_CODE = w.ITEM_CODE AND r.YEAR = w.YEAR AND
r.MONTH = w.MONTH
WHERE r.YEAR = 2020
AND r.MONTH = 8
AND r.RETAIL_SALES > w.WAREHOUSE_SALES;
```

Q12:) Find the total retail sales and warehouse sales for each item across all months and years

```
SELECT ITEM_CODE, ITEM_DESCRIPTION, SUM(RETAIL_SALES) AS
TOTAL_RETAIL_SALES, SUM(WAREHOUSE_SALES) AS TOTAL_WAREHOUSE_SALES
FROM (
    SELECT ITEM_CODE, ITEM_DESCRIPTION, RETAIL_SALES, 0 AS WAREHOUSE_SALES
    FROM retail
    UNION ALL
    SELECT ITEM_CODE, ITEM_DESCRIPTION, 0 AS RETAIL_SALES, WAREHOUSE_SALES
    FROM warehouse
) AS combined_sales
GROUP BY ITEM_CODE, ITEM_DESCRIPTION;
```

Q13:) Retrieve the top 3 months with the highest total retail sales:

```
SELECT YEAR, MONTH, SUM(RETAIL_SALES) AS TOTAL_RETAIL_SALES
FROM retail
GROUP BY YEAR, MONTH
ORDER BY TOTAL_RETAIL_SALES DESC
LIMIT 3;
```