**Task 3: SQL Data Analysis**

**Creating a database(shop) and tables (Items, sales) and inserting the dataset to these tables using Import data wizard option in the MySQL Workbench.**

**Query:**

create database shop;

use shop;

create table Items (

Item\_Identifier VARCHAR(20),

Item\_Weight DECIMAL(6,2),

Item\_Fat\_Content VARCHAR(20),

Item\_Type VARCHAR(50),

Item\_MRP DECIMAL(8,2)

);

select \* from Items;

create table Sales (

Item\_Identifier VARCHAR(20),

Outlet\_Identifier VARCHAR(20),

Item\_Visibility DECIMAL(6,4),

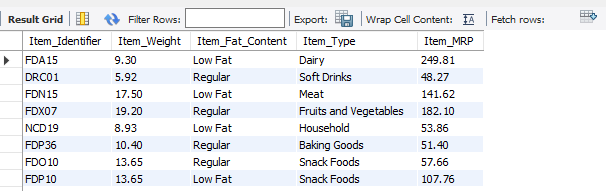
Item\_Outlet\_Sales DECIMAL(8,4)

);

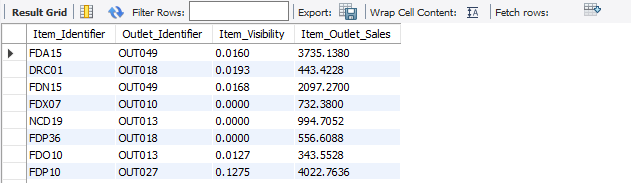
select \* from sales;

**Output Screenshot:**

**Table 1: Items**

****

**Table 2: Sales**

****

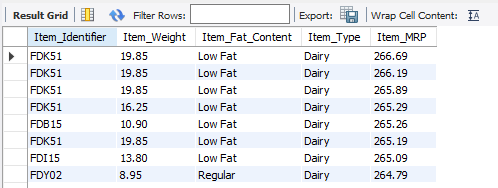
**SQL Analysis:**

1. **Write a query to retrieve all items with Item\_Type = 'Dairy' and Item\_MRP greater than 100. Order the result by Item\_MRP in descending order.**

**Query:**

select \* from items where Item\_Type='Dairy' and Item\_MRP > 100 order by Item\_MRP desc;

**Output Screenshot:**

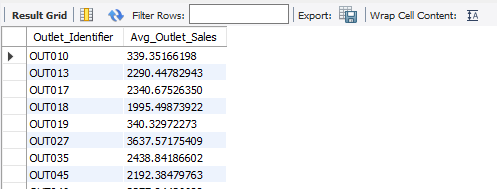
****

1. **Find the average Item\_Outlet\_Sales for each Outlet\_Identifier and group the result accordingly.**

**Query:**

select Outlet\_Identifier, avg(Item\_Outlet\_Sales) as Avg\_Outlet\_Sales from Sales group by Outlet\_Identifier;

**Output Screenshot:**

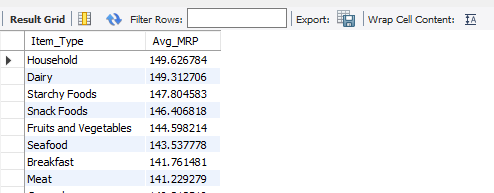
****

1. **List each Item\_Type and its average MRP, sorted from highest to lowest.**

**Query:**

select Item\_Type, avg(Item\_MRP) as Avg\_MRP from items group by Item\_Type order by Avg\_MRP desc;

**Output Screenshot:**

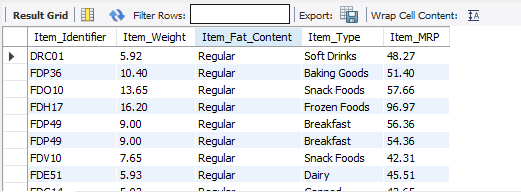
****

1. **Show all items that have “Regular” fat content and MRP below 100.**

**Query:**

select \* from items where Item\_Fat\_Content='Regular' and Item\_MRP < 100;

**Output Screenshot:**

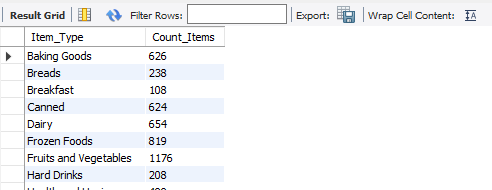
****

1. **Count the number of items per Item\_Type.**

**Query:**

select Item\_Type, count(\*) as Count\_Items from items group by Item\_Type;

**Output Screenshot:**

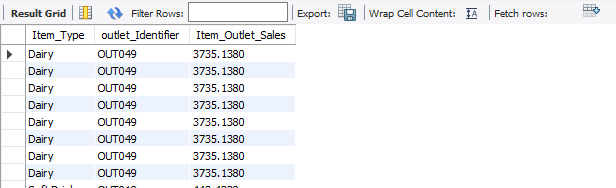
****

1. **Show item type, outlet identifier, and outlet sales using an INNER JOIN between Items and Sales.**

**Query:**

select i.Item\_Type, s.outlet\_Identifier, s.Item\_Outlet\_Sales from Items as i inner join Sales as s on i.Item\_Identifier=s.Item\_Identifier;

**Output Screenshot:**

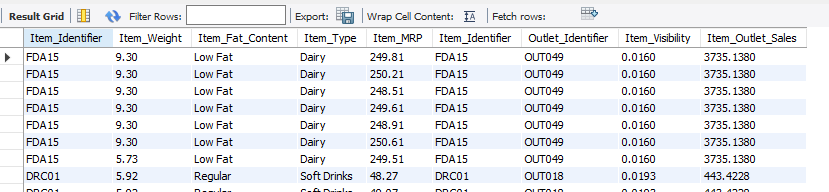
****

1. **Display all items with their sales info using a LEFT JOIN — include items even if not sold.**

**Query:**

select \* from Items left join Sales on Items.Item\_Identifier=Sales.Item\_Identifier;

**Output Screenshot:**

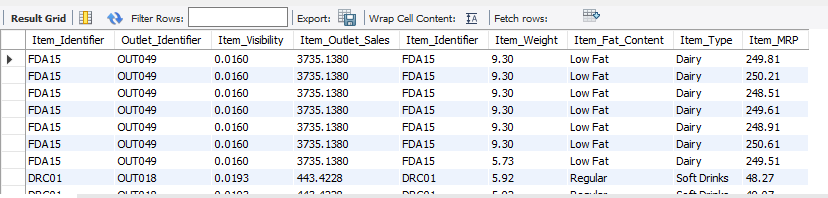
****

1. **Display all sales, including those whose item details are missing (use a RIGHT JOIN).**

**Query:**

select \* from sales right join items on sales.Item\_Identifier=items.Item\_Identifier;

**Output Screenshot:**

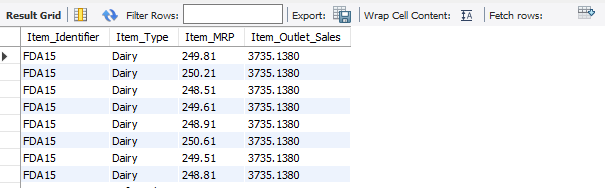
****

1. **Write an INNER JOIN to display Item\_Identifier, Item\_Type, Item\_MRP, and Item\_Outlet\_Sales.**

**Query:**

select i.Item\_Identifier, i.Item\_Type, i.Item\_MRP, s.Item\_Outlet\_Sales from Items i inner join sales s on i.Item\_Identifier=s.Item\_Identifier;

**Output Screenshot:**

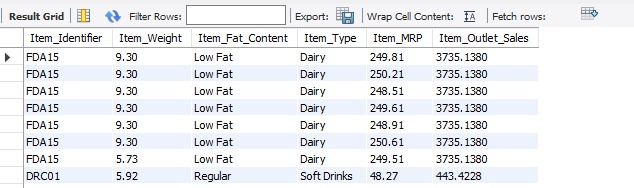
****

1. **Write a LEFT JOIN to display all items from the items table along with their Item\_Outlet\_Sales (if any). Include items even if they have no sales data.**

**Query:**

select i.Item\_Identifier,i.Item\_Weight,i.Item\_Fat\_Content,i.Item\_Type, i.Item\_MRP, s.Item\_Outlet\_Sales from items i left join sales s on i.Item\_Identifier=s.Item\_Identifier;

**Output Screenshot:**

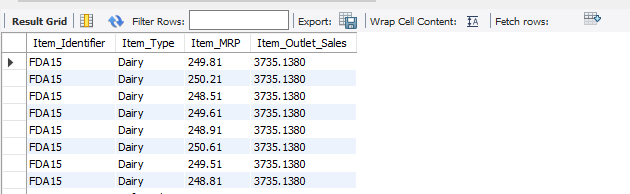
****

1. **Write a RIGHT JOIN to display all sales along with corresponding item details from items (if any). Include sales even if the item details are missing.**

**Query:**

select s.Item\_Identifier, i.Item\_Type, i.Item\_MRP, s.Item\_Outlet\_Sales from items i right join sales s on i.Item\_Identifier=s.Item\_Identifier;

**Output Screenshot:**

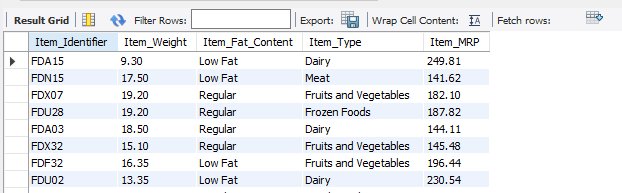
****

1. **Find all items whose MRP is greater than the average MRP of all items.**

**Query:**

select \* from Items where Item\_MRP > (select avg(Item\_MRP) as Avg\_MRP from Items);

**Output Screenshot:**

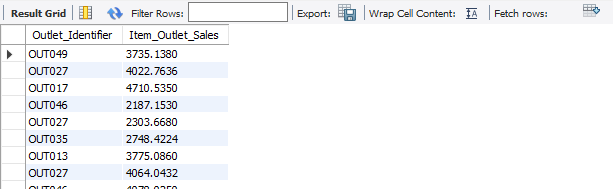
****

1. **Find the outlet(s) with sales higher than the average sales across all outlets.**

**Query:**

select Outlet\_Identifier, Item\_Outlet\_Sales from sales where Item\_Outlet\_Sales > (select avg(Item\_Outlet\_Sales) from sales);

**Output Screenshot:**

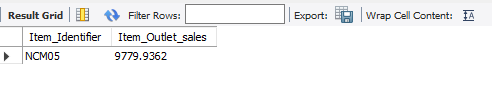
****

1. **Write a query to find the Item\_Identifier with the maximum Item\_Outlet\_Sales using a subquery**.

**Query:**

select Item\_Identifier, Item\_Outlet\_sales from sales where Item\_Outlet\_sales in (select max(Item\_Outlet\_sales) from sales);

**Output Screenshot:**

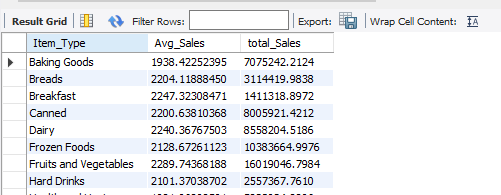
****

1. **Find the total sales (SUM) and average sales (AVG) for each Item\_Type. Join items and sales tables appropriately.**

**Query:**

select i.Item\_Type, avg(s.Item\_Outlet\_Sales) as Avg\_Sales, sum(s.Item\_Outlet\_Sales) as total\_Sales from items i join sales s on i.Item\_Identifier=s.Item\_Identifier group by i.Item\_Type;

**Output Screenshot:**

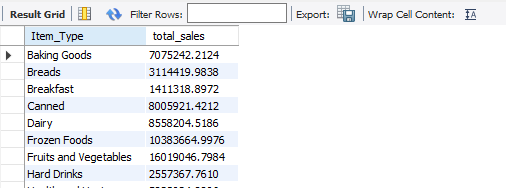
****

1. **Find the total sales per item type**.

**Query**:

select i.Item\_Type, sum(s.Item\_Outlet\_sales) as total\_sales from sales s join items i on i.Item\_Identifier=s.Item\_Identifier group by Item\_Type;

**Output Screenshot:**

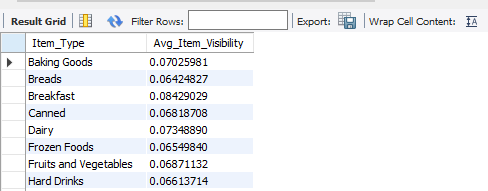
****

1. **Show the average visibility per Item\_Type.**

**Query:**

select i.Item\_Type, avg(s.Item\_Visibility) as Avg\_Item\_Visibility from items i join sales s on i.Item\_Identifier=s.Item\_Identifier group by i.Item\_Type;

**Output Screenshot:**

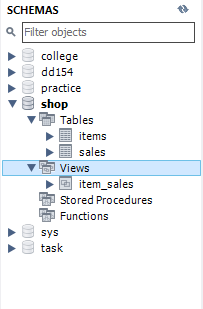
****

1. **Create a view that shows** **Item\_Identifier, Item\_Type, and total outlet sales**.

**Query**:

create view Item\_sales as select i.Item\_Identifier, i.Item\_Type, sum(s.Item\_Outlet\_Sales) as Total\_Sales from items i join sales s on i.Item\_Identifier=s.Item\_Identifier group by i.Item\_Identifier, i.Item\_Type;

**Output Screenshot:**

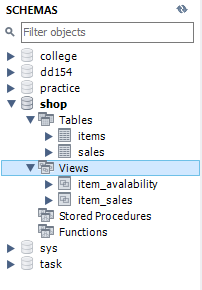
****

1. **Create a view for** **items with MRP above 150 and visibility below 0.02**.

**Query**:

create view Item\_avalability as select i.Item\_MRP, s.Item\_Visibility from items i join sales s on i.Item\_Identifier=s.Item\_Identifier where i.Item\_MRP > 150 and s.Item\_Visibility < 0.02;

**Output Screenshot:**

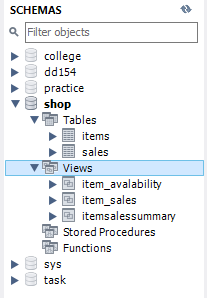
****

1. **Create a** **view** **called ItemSalesSummary that includes Item\_Identifier, Item\_Type, Item\_MRP, Item\_Outlet\_Sales.**

**Query:**

create view ItemSalesSummary as select i.Item\_Identifier, i.Item\_Type, i.Item\_MRP, s.Item\_Outlet\_Sales from items i join sales s on i.Item\_Identifier=s.Item\_Identifier;

**Output Screenshot:**

****

1. **Suggest an index that can optimize the query used in Q1 (WHERE Item\_Type = 'Dairy' AND Item\_MRP > 100 ORDER BY Item\_MRP DESC).**

**Query:**

create index idx\_Item\_type\_MRP on items(Item\_Type,Item\_MRP);

1. **Create indexes on Item\_Identifier, Item\_Type, and Outlet\_Identifier to improve JOIN and filtering performance.**

**Query:**

create index idx\_Item on items(Item\_Identifier,Item\_Type);

create index idx\_Sale on sales(Outlet\_Identifier);