

SQL PROJECT

Topic: Happy Parcel (Online Gift Store)

Introduction:

This project aims to design and manage a database for an online gift store called **Happy Parcel**. The system helps in maintaining details about **products, categories, customers, and orders**.

Using SQL, this project allows the business to track sales, manage inventory, analyze customer behavior, and generate useful business insights.

Objectives:

- To design a relational database for an online gift store
- To store and manage customer, product, and order data
- To analyze sales performance and revenue trends
- To identify top-selling products and valuable customers
- To understand payment methods and customer behavior

Database Design:

The database **Happy_Parcel** consists of the following tables:

- **Categories:** Stores different gift categories
- **Products:** Stores product details such as name, price, category, and stock
- **Customers:** Stores customer information like name, city, email, and phone number
- **Orders:** Stores order details including order date, quantity, total amount, status, and payment method

Table Creation Queries:

```
1 ● CREATE DATABASE Happy_Parcel;
2 ● USE Happy_Parcel;
3 -----# CREATED CATEGORIES TABLE -----
4 ● CREATE TABLE Categories (
5     Category_id INT PRIMARY KEY AUTO_INCREMENT,
6     Category_name VARCHAR(50) NOT NULL);
7 -----# CREATED PRODUCTS TABLE -----
8 ● CREATE TABLE Products (
9     Product_id INT PRIMARY KEY AUTO_INCREMENT,
10    Product_name VARCHAR(100) NOT NULL,
11    Category_id INT,
12    unit_price DECIMAL(10, 2),
13    Stock_quantity INT,
14    FOREIGN KEY (Category_id) REFERENCES Categories(Category_id));
15 -----# CREATED CUSTOMER TABLE -----
16 ● CREATE TABLE Customers (
17     customer_id INT PRIMARY KEY AUTO_INCREMENT,
18     first_name VARCHAR(50) NOT NULL,
19     last_name VARCHAR(50) NOT NULL,
20     city VARCHAR(50),
21     email VARCHAR(100) UNIQUE,
22     phone_number VARCHAR(15));
23
24 -----# CREATED ORDER TABLE -----
25 ● CREATE TABLE Orders (
26     Order_id INT PRIMARY KEY AUTO_INCREMENT,
27     Customer_id INT,
28     Product_id INT,
```

Data Insertion:

Sample data is inserted into all tables to simulate a real-world business scenario. The data includes multiple customers, products, categories, and orders with different payment methods and order statuses.

```
INSERT INTO Categories (Category_name) VALUES
('Birthday Gifts'),
('Personalized Gifts'),
('Valentines Day Gifts'),
('Home Decor'),
('Wedding Gifts');

INSERT INTO Products (Product_name, Category_id, unit_price, Stock_quantity) VALUES
('Chocolate Explosion Box', 1, 500.00, 10),
('Mini Teddy Bear', 1, 249.00, 10),
('Customized Mug', 2, 299.00, 40),
('Scented Candles Set', 4, 750.00, 60),
('Decorative Wall Clock', 4, 1299.00, 40),
('Romantic Couple Figurine', 5, 899.00, 50),
('Heart-Shaped Chocolate Bouquet', 3, 699.00, 80),
('Wedding Gift Hamper', 5, 1599.00, 40),
('Mini Birthday Cupcake Box', 1, 499.00, 10),
('Customized Calendar with Photos', 2, 699.00, 65),
('Wedding Wish Card Set', 5, 149.00, 40),
('Fridge Magnet Quote', 4, 199.00, 110);

INSERT INTO Customers (first_name, last_name, city, email, phone_number)
VALUES
('Aarav', 'Sharma', 'Mumbai', 'aarav.sharma@gmail.com', '9876543210'),
('Priya', 'Mehta', 'Mumbai', 'priya.mehta@gmail.com', '9876501234'),
('Rohit', 'Verma', 'Bangalore', 'rohit.verma@gmail.com', '9867543210');

INSERT INTO Orders (customer_id, Product_id, Order_date, Quantity, Total_amount, Order_status, payment_method)
VALUES
(2008, 112, '2025-10-20', 1, 199.00, 'Delivered', 'UPI'),
(2005, 106, '2025-10-25', 2, 1798.00, 'Pending', 'UPI'),
(2009, 103, '2025-09-22', 1, 299.00, 'Delivered', 'COD'),
(2002, 109, '2025-10-21', 1, 499.00, 'Shipped', 'Net Banking'),
(2007, 103, '2025-10-09', 1, 299.00, 'Delivered', 'UPI'),
(2001, 108, '2025-08-12', 2, 3198.00, 'Delivered', 'UPI'),
(2006, 102, '2025-10-15', 1, 249.00, 'Delivered', 'UPI'),
(2005, 109, '2025-11-05', 1, 499.00, 'Shipped', 'COD'),
(2004, 111, '2025-10-27', 2, 298.00, 'Shipped', 'Net Banking'),
(2008, 102, '2025-07-11', 2, 498.00, 'Delivered', 'UPI'),
(2003, 107, '2025-07-20', 1, 699.00, 'Delivered', 'UPI'),
(2005, 101, '2025-06-17', 1, 600.00, 'Delivered', 'UPI'),
(2002, 106, '2025-06-14', 1, 899.00, 'Delivered', 'UPI'),
(2001, 101, '2025-10-01', 1, 600.00, 'Delivered', 'COD'),
(2007, 106, '2025-11-10', 1, 899.00, 'Shipped', 'COD'),
(2008, 107, '2025-11-21', 1, 699.00, 'Pending', 'UPI'),
(2004, 103, '2025-10-07', 2, 598.00, 'Cancelled', 'UPI'),
(2003, 104, '2025-09-05', 1, 750.00, 'Delivered', 'UPI'),
(2001, 111, '2025-08-05', 3, 447.00, 'Delivered', 'UPI'),
(2002, 101, '2025-11-02', 1, 600.00, 'Pending', 'COD'),
(2005, 102, '2025-10-29', 2, 498.00, 'Pending', 'UPI');
```

SQL Queries and Output:

Basic Queries:

```
SELECT * FROM products;
```

```
SELECT * FROM customers;
```

```
SELECT * FROM orders;
```

```
SELECT * FROM CATEGORIES;
```

customer_id	first_name	last_name	city	email	phone_number
2001	Aarav	Sharma	Mumbai	aarav.sharma@gmail.com	9876543210
2002	Priya	Mehta	Mumbai	priya.mehta@gmail.com	9876501234
2003	Rohit	Verma	Bangalore	rohit.verma@gmail.com	9867543210
2004	Sneha	Patel	Ahmedabad	sneha.patel@gmail.com	9898745632
2005	Karan	Singh	Chennai	karan.singh@gmail.com	9823456789
2006	Ananya	Nair	Kochi	ananya.nair@gmail.com	9812345678
2007	Rahul	Gupta	Pune	rahul.gupta@gmail.com	9856743120
2008	Simran	Kaur	Mumbai	simran.kaur@gmail.com	9876123450
2009	Arjun	Reddy	Hyderabad	arjun.reddy@gmail.com	9845012345
2010	Neha	Kapoor	Bangalore	neha.kapoor@gmail.com	9876509876

Order_id	Customer_id	Product_id	Order_date	Quantity	Total_amount	Order_status	payment_method
3001	2008	112	2025-10-20	1	199.00	Delivered	UPI
3002	2005	106	2025-10-25	2	1798.00	Pending	UPI
3003	2009	103	2025-09-22	1	299.00	Delivered	COD
3004	2002	109	2025-10-21	1	499.00	Shipped	Net Banking
3005	2007	103	2025-10-09	1	299.00	Delivered	UPI
3006	2001	108	2025-08-12	2	3198.00	Delivered	UPI
3007	2006	102	2025-10-15	1	249.00	Delivered	UPI
3008	2005	109	2025-11-05	1	499.00	Shipped	COD
3009	2004	111	2025-10-27	2	298.00	Shipped	Net Banking
3010	2008	102	2025-07-11	2	498.00	Delivered	UPI

Product_id	Product_name	Category_id	unit_price	Stock_quantity
101	Chocolate Explosion Box	1	500.00	19
102	Mini Teddy Bear	1	249.00	10
103	Customized Mug	2	299.00	40
104	Scented Candles Set	4	750.00	69
105	Decorative Wall Clock	4	1299.00	40
106	Romantic Couple Figurine	5	899.00	50
107	Heart-Shaped Chocolate Bouquet	3	699.00	80
108	Wedding Gift Hamper	5	1599.00	40
109	Mini Birthday Cupcake Box	1	499.00	10
110	Customized Calendar with Photos	2	699.00	65
111	Wedding Wish Card Set	5	149.00	40
112	Fridge Magnet Quote	4	199.00	110

Category_id	Category_name
1	Birthday Gifts
2	Personalized Gifts
3	valentines Day Gifts
4	Home Decor
5	Wedding Gifts

1. Monthly Sales Trend

```
SELECT DATE_FORMAT(Order_date, '%Y-%M') AS Month, SUM(Total_amount) AS Monthly_Revenue
FROM Orders
WHERE Order_status = 'Delivered'
GROUP BY Month
ORDER BY Month;
```

	Month	Monthly_Revenue
▶	2025-August	3645.00
	2025-July	1197.00
	2025-June	1499.00
	2025-October	1347.00
	2025-September	1049.00

2. Top selling products by quantity sold

```
SELECT p.Product_name, COUNT(o.Order_id) AS Orders_Count
FROM Orders o
JOIN Products p ON o.Product_id = p.Product_id
GROUP BY p.Product_name
ORDER BY Orders_Count DESC
LIMIT 5;
```

Product_name	Orders_Count
Chocolate Explosion Box	3
Mini Teddy Bear	3
Customized Mug	3
Romantic Couple Figurine	3
Heart-Shaped Chocolate Bouquet	2

3. Top 5 Customers by Total Spending

```
SELECT c.first_name, c.last_name, SUM(o.Total_amount) AS Total_Spent
FROM Orders o
JOIN Customers c ON o.Customer_id = c.Customer_id
WHERE o.Order_status = 'Delivered'
GROUP BY c.Customer_id
ORDER BY Total_Spent DESC
LIMIT 5;
```

first_name	last_name	Total_Spent
Aarav	Sharma	4245.00
Rohit	Verma	1449.00
Priya	Mehta	899.00
Simran	Kaur	697.00
Karan	Singh	600.00

4. Find Orders Above Average Order Value(AOV)

```
SELECT Order_id,Customer_id,Total_amount
FROM Orders
WHERE Total_amount > (SELECT AVG(Total_amount) FROM Orders);
```

Order_id	Customer_id	Total_amount
3002	2005	1798.00
3006	2001	3198.00
3013	2002	899.00
3015	2007	899.00
3018	2003	750.00
NULL	NULL	NULL

5. Show category-wise total revenue

```
SELECT
    cat.Category_name,
    SUM(o.Total_amount) AS Total_Revenue
FROM Orders o
JOIN Products p ON o.Product_id = p.Product_id
JOIN Categories cat ON p.Category_id = cat.Category_id
WHERE o.Order_status = 'Delivered'
GROUP BY cat.Category_name
ORDER BY Total_Revenue DESC;
```

Category_name	Total_Revenue
Wedding Gifts	4544.00
Birthday Gifts	1947.00
Home Decor	949.00
valentines Day Gifts	699.00
Personalized Gifts	598.00

6. Show products which are out of stock or low in stock (below 20 units)

```
SELECT * FROM products where stock_quantity<20;
```

Product_id	Product_name	Category_id	unit_price	Stock_quantity
101	Chocolate Explosion Box	1	500.00	19
102	Mini Teddy Bear	1	249.00	10
109	Mini Birthday Cupcake Box	1	499.00	10
NULL	NULL	NULL	NULL	NULL

7. Regular customers (more than 2 orders)

```
SELECT c.first_name, c.last_name, COUNT(o.Order_id) AS total_orders
FROM Customers c
JOIN Orders o ON c.Customer_id = o.Customer_id
GROUP BY c.Customer_id
HAVING COUNT(o.Order_id) > 2
ORDER BY total_orders DESC;
```

first_name	last_name	total_orders
Karan	Singh	4
Aarav	Sharma	3
Priya	Mehta	3
Simran	Kaur	3

8. Most used payment method

```
SELECT payment_method, COUNT(*) AS usage_count
FROM orders
GROUP BY payment_method
ORDER BY usage_count DESC
LIMIT 1;
```

	payment_method	usage_count
▶	UPI	14

9. Return Rate (Cancelled Orders %)

```
SELECT ROUND((SUM(CASE WHEN Order_status = 'Cancelled' THEN 1 ELSE 0 END) * 100.0) / COUNT(*),2)
AS Return_Rate_Percentage
FROM Orders;
```

	Return_Rate_Percentage
▶	4.76

10. City-wise Total Revenue

```
SELECT c.city,SUM(o.Total_amount) AS Total_Revenue,COUNT(o.Order_id) AS Total_Orders
FROM Orders o
JOIN Customers c ON o.Customer_id = c.Customer_id
WHERE o.Order_status = 'Delivered'
GROUP BY c.city
ORDER BY Total_Revenue DESC;
```

city	Total_Revenue	Total_Orders
Mumbai	5841.00	6
Bangalore	1449.00	2
Chennai	600.00	1
Hyderabad	299.00	1
Pune	299.00	1
Kochi	249.00	1

Conclusion:

This project helped me understand how **SQL can be used to design databases and analyze real-world business data.**

I learned how to write SQL queries to extract meaningful insights such as sales trends, customer behavior, and inventory status.

The **Happy Parcel SQL Project** improved my confidence in SQL and strengthened my analytical thinking skills.

