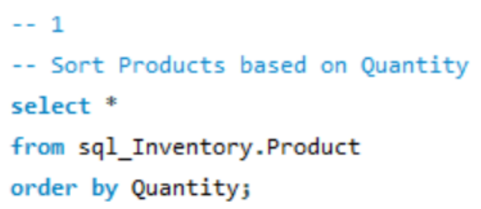
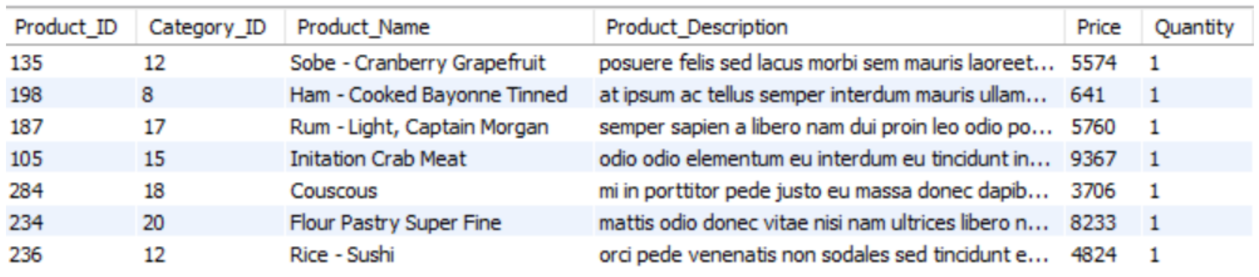


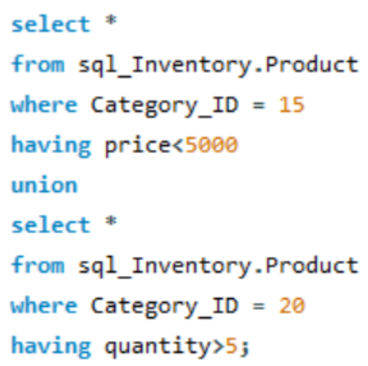
```
-- 1
-- Sort Products based on Quantity
select *
from sql_Inventory.Product
order by Quantity;
```

Here the products are being sorted by quantity in ascending order. The table we get on running the query is given below.



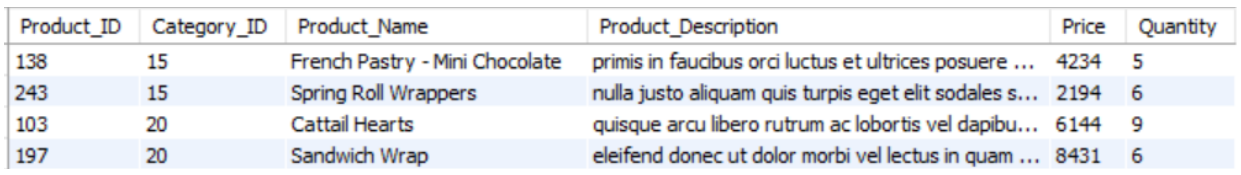
| Product_ID | Category_ID | Product_Name | Product_Description | Price | Quantity |
|---|---|---|---|---|---|
| 135 | 12 | Sobe - Cranberry Grapefruit | posuere felis sed lacus morbi sem mauris laoreet... | 5574 | 1 |
| 198 | 8 | Ham - Cooked Bayonne Tinned | at ipsum ac tellus semper interdum mauris ullam... | 641 | 1 |
| 187 | 17 | Rum - Light, Captain Morgan | semper sapien a libero nam dui proin leo odio po... | 5760 | 1 |
| 105 | 15 | Initation Crab Meat | odio odio elementum eu interdum eu tincidunt in... | 9367 | 1 |
| 284 | 18 | Couscous | mi in porttitor pede justo eu massa donec dapib... | 3706 | 1 |
| 234 | 20 | Flour Pastry Super Fine | mattis odio donec vitae nisi nam ultrices libero n... | 8233 | 1 |
| 236 | 12 | Rice - Sushi | orci pede venenatis non sodales sed tincidunt e... | 4824 | 1 |

---



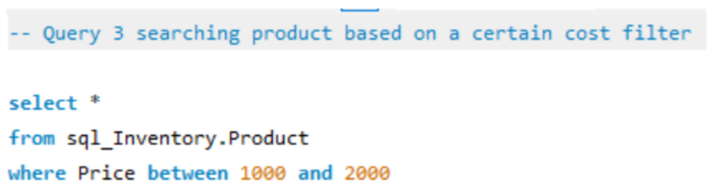
```
select *
from sql_Inventory.Product
where Category_ID = 15
having price<5000
union
select *
from sql_Inventory.Product
where Category_ID = 20
having quantity>5;
```

The query here uses union to group the result of two queries. The products here are being sorted by their category IDs.



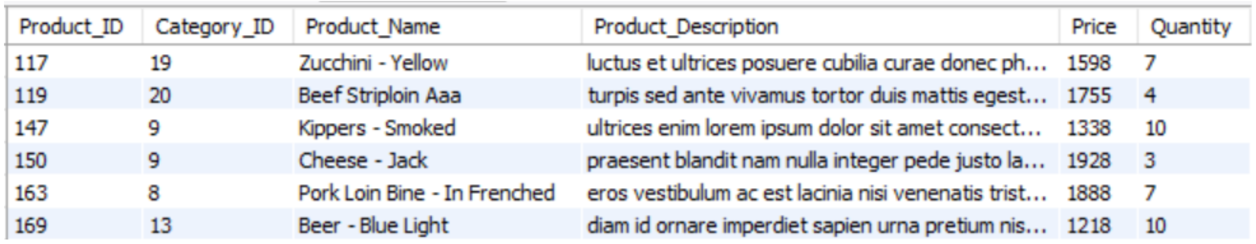
| Product_ID | Category_ID | Product_Name | Product_Description | Price | Quantity |
|---|---|---|---|---|---|
| 138 | 15 | French Pastry - Mini Chocolate | primis in faucibus orci luctus et ultrices posuere ... | 4234 | 5 |
| 243 | 15 | Spring Roll Wrappers | nulla justo aliquam quis turpis eget elit sodales s... | 2194 | 6 |
| 103 | 20 | Cattail Hearts | quisque arcu libero rutrum ac lobortis vel dapibu... | 6144 | 9 |
| 197 | 20 | Sandwich Wrap | eleifend donec ut dolor morbi vel lectus in quam ... | 8431 | 6 |

---



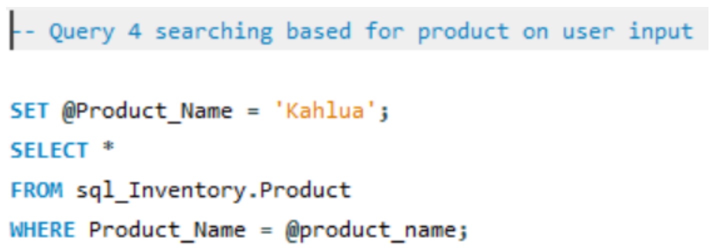
```sql
-- Query 3 searching product based on a certain cost filter

select *
from sql_Inventory.Product
where Price between 1000 and 2000
```

Here the customer will be able to apply specific constraints to search the inventory.



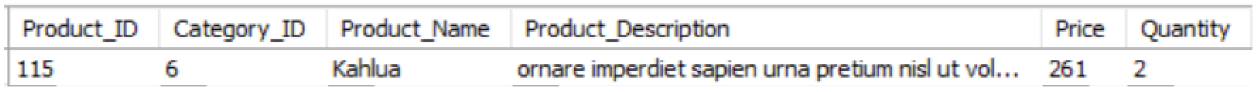
| Product_ID | Category_ID | Product_Name | Product_Description | Price | Quantity |
|---|---|---|---|---|---|
| 117 | 19 | Zucchini - Yellow | luctus et ultrices posuere cubilia curae donec ph... | 1598 | 7 |
| 119 | 20 | Beef Striploin Aaa | turpis sed ante vivamus tortor duis mattis egest... | 1755 | 4 |
| 147 | 9 | Kippers - Smoked | ultrices enim lorem ipsum dolor sit amet consect... | 1338 | 10 |
| 150 | 9 | Cheese - Jack | praesent blandit nam nulla integer pede justo la... | 1928 | 3 |
| 163 | 8 | Pork Loin Bine - In Frenched | eros vestibulum ac est lacinia nisi venenatis trist... | 1888 | 7 |
| 169 | 13 | Beer - Blue Light | diam id ornare imperdiet sapien urna pretium nis... | 1218 | 10 |

---



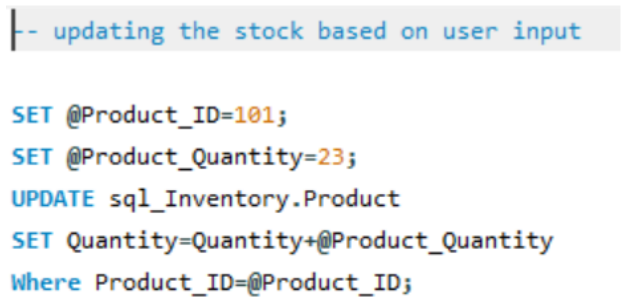
```sql
-- Query 4 searching based for product on user input

SET @Product_Name = 'Kahlua';
SELECT *
FROM sql_Inventory.Product
WHERE Product_Name = @product_name;
```

Based on user input, the product can be searched for. This can be extended to other attributes of the product as well. The user input can vary.



| Product_ID | Category_ID | Product_Name | Product_Description | Price | Quantity |
|---|---|---|---|---|---|
| 115 | 6 | Kahlua | ornare imperdiet sapien urna pretium nisl ut vol... | 261 | 2 |

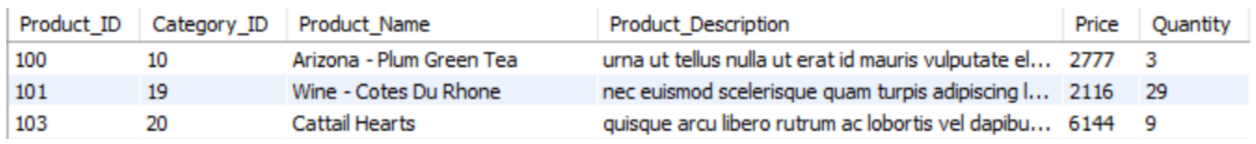
---



```sql
-- updating the stock based on user input

SET @Product_ID=101;
SET @Product_Quantity=23;
UPDATE sql_Inventory.Product
SET Quantity=Quantity+@Product_Quantity
Where Product_ID=@Product_ID;
```

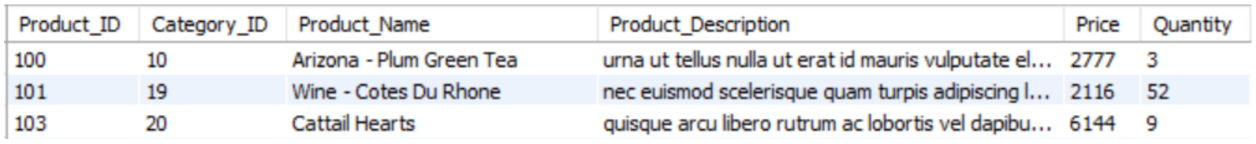
Here the supplier and/or admin would be able to change the stock/quantity of the products present in the inventory based on their input.

*Before*



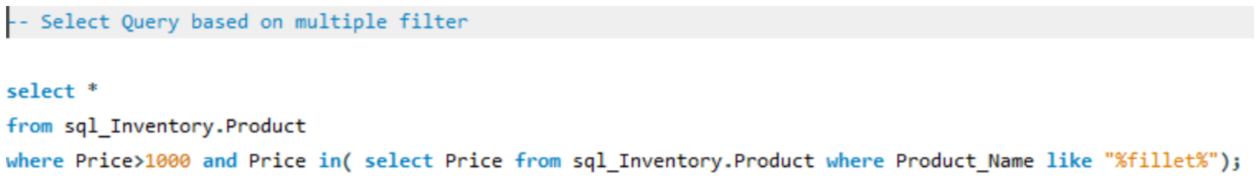
| Product_ID | Category_ID | Product_Name | Product_Description | Price | Quantity |
|---|---|---|---|---|---|
| 100 | 10 | Arizona - Plum Green Tea | urna ut tellus nulla ut erat id mauris vulputate el... | 2777 | 3 |
| 101 | 19 | Wine - Cotes Du Rhone | nec euismod scelerisque quam turpis adipiscing l... | 2116 | 29 |
| 103 | 20 | Cattail Hearts | quisque arcu libero rutrum ac lobortis vel dapibu... | 6144 | 9 |

*After*



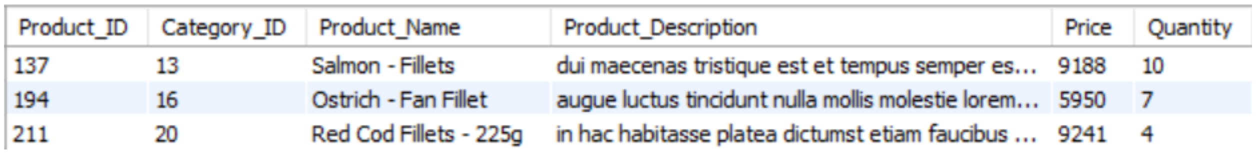
| Product_ID | Category_ID | Product_Name | Product_Description | Price | Quantity |
|---|---|---|---|---|---|
| 100 | 10 | Arizona - Plum Green Tea | urna ut tellus nulla ut erat id mauris vulputate el... | 2777 | 3 |
| 101 | 19 | Wine - Cotes Du Rhone | nec euismod scelerisque quam turpis adipiscing l... | 2116 | 52 |
| 103 | 20 | Cattail Hearts | quisque arcu libero rutrum ac lobortis vel dapibu... | 6144 | 9 |

---



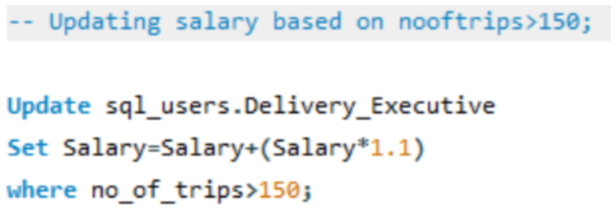
```
-- Select Query based on multiple filter

select *
from sql_Inventory.Product
where Price>1000 and Price in( select Price from sql_Inventory.Product where Product_Name like "%fillet%");
```

The products can be searched for based on various filters. This is an example of a nested query. String functions are used to find products with names similar to the word ‘fillet’.



| Product_ID | Category_ID | Product_Name | Product_Description | Price | Quantity |
|---|---|---|---|---|---|
| 137 | 13 | Salmon - Fillets | dui maecenas tristique est et tempus semper es... | 9188 | 10 |
| 194 | 16 | Ostrich - Fan Fillet | augue luctus tincidunt nulla mollis molestie lorem... | 5950 | 7 |
| 211 | 20 | Red Cod Fillets - 225g | in hac habitasse platea dictumst etiam faucibus ... | 9241 | 4 |

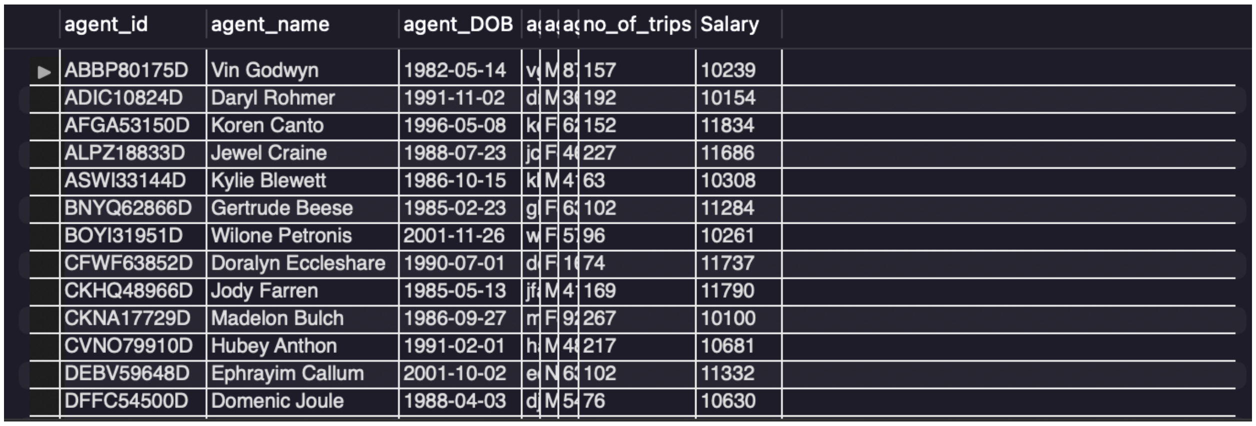
---



```
-- Updating salary based on nooftrips>150;

Update sql_users.Delivery_Executive
Set Salary=Salary+(Salary*1.1)
where no_of_trips>150;
```

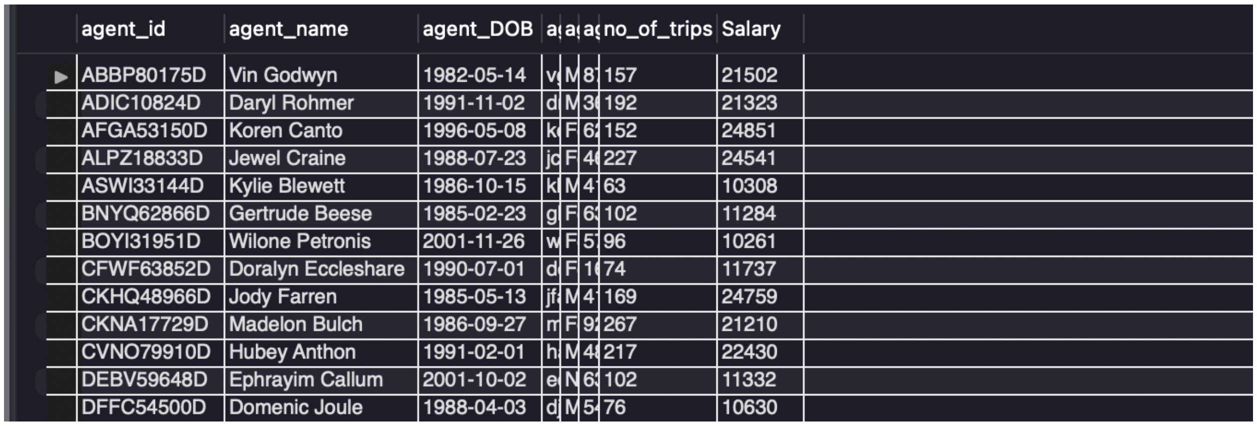
The salary for the delivery executives are being updated according to the number of trips they have gone on.

Before



| agent_id | agent_name | agent_DOB | a | a | a | no_of_trips | Salary |
|---|---|---|---|---|---|---|---|
| ABBP80175D | Vin Godwyn | 1982-05-14 | v | M | 8 | 157 | 10239 |
| ADIC10824D | Daryl Rohmer | 1991-11-02 | d | M | 3 | 192 | 10154 |
| AFGA53150D | Koren Canto | 1996-05-08 | k | F | 6 | 152 | 11834 |
| ALPZ18833D | Jewel Craine | 1988-07-23 | j | F | 4 | 227 | 11686 |
| ASWI33144D | Kylie Blewett | 1986-10-15 | k | M | 4 | 63 | 10308 |
| BNYQ62866D | Gertrude Beese | 1985-02-23 | g | F | 6 | 102 | 11284 |
| BOYI31951D | Wilone Petronis | 2001-11-26 | w | F | 5 | 96 | 10261 |
| CFWF63852D | Doralyn Eccleshare | 1990-07-01 | d | F | 1 | 74 | 11737 |
| CKHQ48966D | Jody Farren | 1985-05-13 | j | M | 4 | 169 | 11790 |
| CKNA17729D | Madelon Bulch | 1986-09-27 | m | F | 9 | 267 | 10100 |
| CVNO79910D | Hubey Anthon | 1991-02-01 | h | M | 4 | 217 | 10681 |
| DEBV59648D | Ephrayim Callum | 2001-10-02 | e | N | 6 | 102 | 11332 |
| DFFC54500D | Domenic Joule | 1988-04-03 | d | M | 5 | 76 | 10630 |

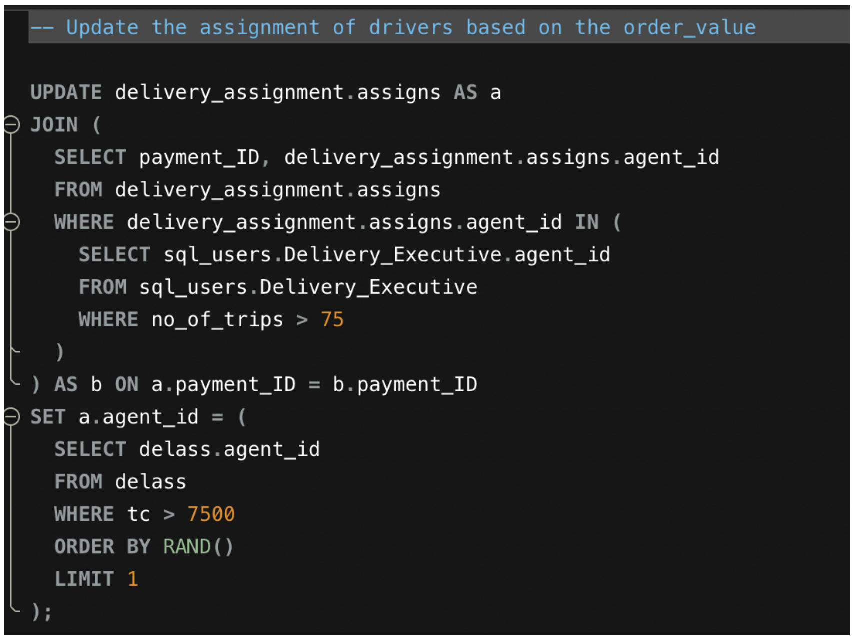
After



| agent_id | agent_name | agent_DOB | a | a | a | no_of_trips | Salary |
|---|---|---|---|---|---|---|---|
| ABBP80175D | Vin Godwyn | 1982-05-14 | v | M | 8 | 157 | 21502 |
| ADIC10824D | Daryl Rohmer | 1991-11-02 | d | M | 3 | 192 | 21323 |
| AFGA53150D | Koren Canto | 1996-05-08 | k | F | 6 | 152 | 24851 |
| ALPZ18833D | Jewel Craine | 1988-07-23 | jc | F | 4 | 227 | 24541 |
| ASWI33144D | Kylie Blewett | 1986-10-15 | k | M | 4 | 63 | 10308 |
| BNYQ62866D | Gertrude Beese | 1985-02-23 | g | F | 6 | 102 | 11284 |
| BOYI31951D | Wilone Petronis | 2001-11-26 | w | F | 5 | 96 | 10261 |
| CFWF63852D | Doralyn Eccleshare | 1990-07-01 | d | F | 1 | 74 | 11737 |
| CKHQ48966D | Jody Farren | 1985-05-13 | jf | M | 4 | 169 | 24759 |
| CKNA17729D | Madelon Bulch | 1986-09-27 | m | F | 9 | 267 | 21210 |
| CVNO79910D | Hubey Anthon | 1991-02-01 | h | M | 4 | 217 | 22430 |
| DEBV59648D | Ephrayim Callum | 2001-10-02 | e | N | 6 | 102 | 11332 |
| DFFC54500D | Domenic Joule | 1988-04-03 | d | M | 5 | 76 | 10630 |

---

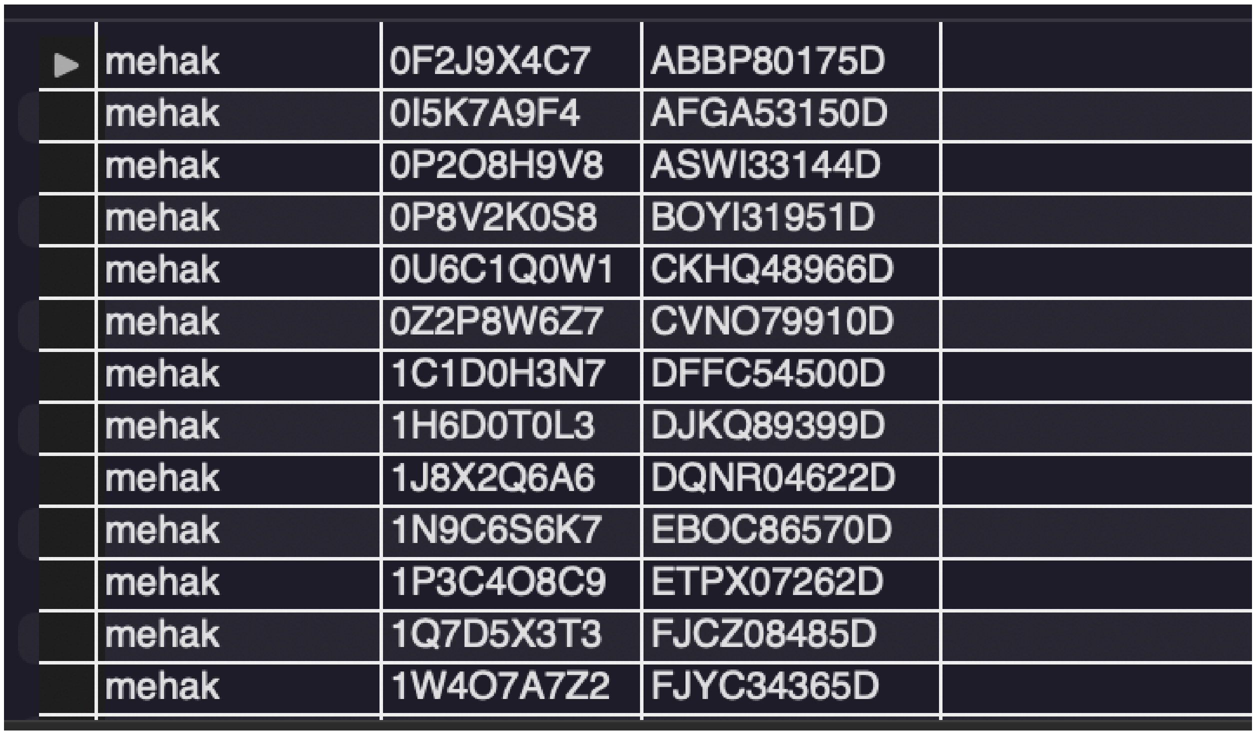
---



```sql
-- Update the assignment of drivers based on the order_value

UPDATE delivery_assignment.assigns AS a
JOIN (
  SELECT payment_ID, delivery_assignment.assigns.agent_id
  FROM delivery_assignment.assigns
  WHERE delivery_assignment.assigns.agent_id IN (
    SELECT sql_users.Delivery_Executive.agent_id
    FROM sql_users.Delivery_Executive
    WHERE no_of_trips > 75
  )
) AS b ON a.payment_ID = b.payment_ID
SET a.agent_id = (
  SELECT delass.agent_id
  FROM delass
  WHERE tc > 7500
  ORDER BY RAND()
  LIMIT 1
);
```

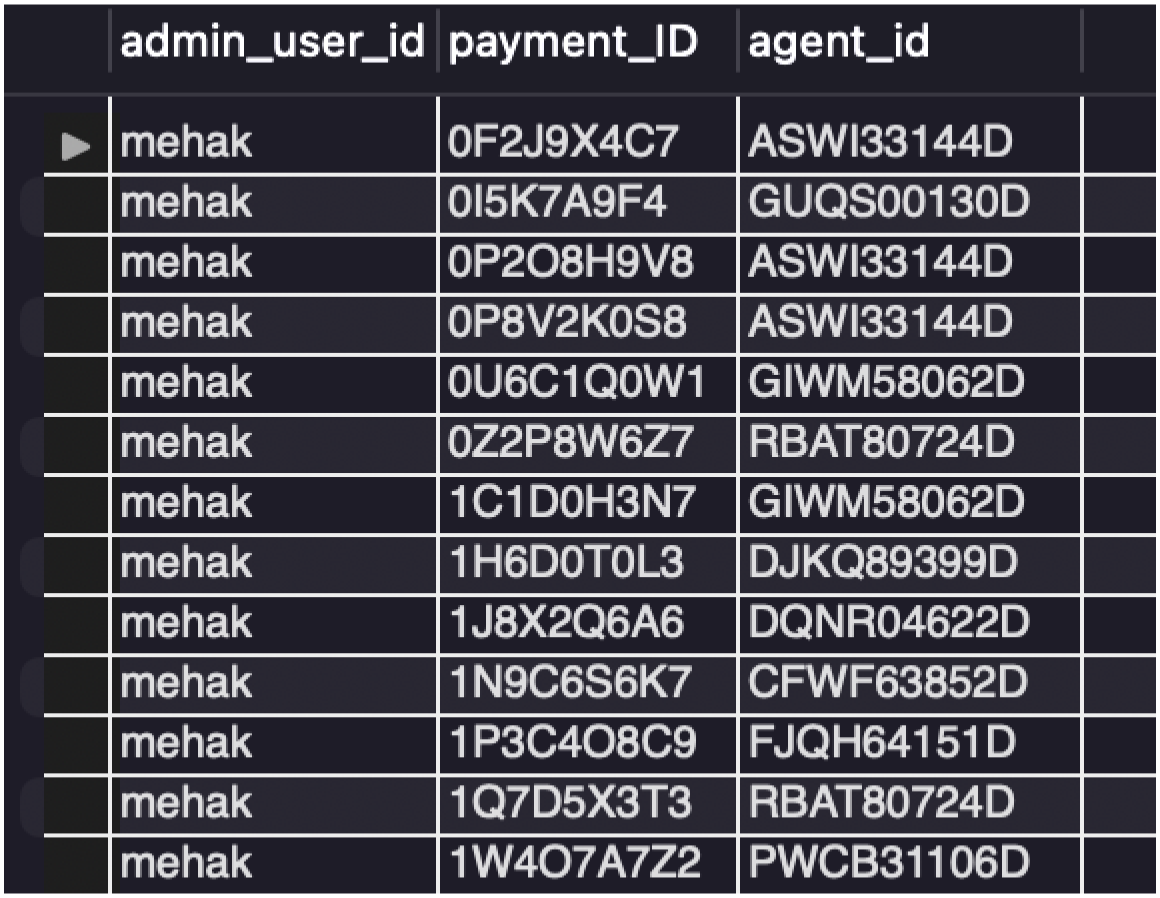
Here we change the delivery assignment for orders with the cost more than 7500, and provide them experienced delivery executives with more than 75 trips.

Before

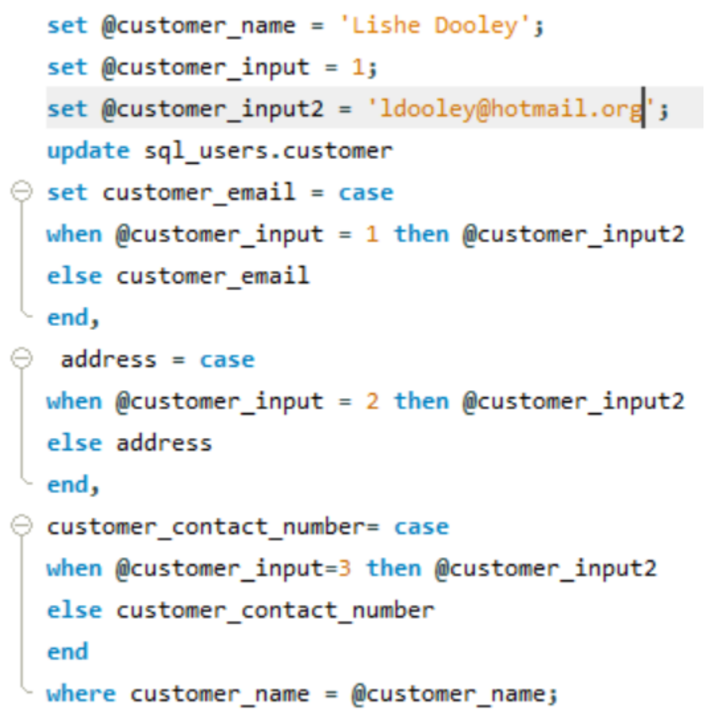


| | | |
|---|---|---|
| mehak | 0F2J9X4C7 | ABBP80175D |
| mehak | 0I5K7A9F4 | AFGA53150D |
| mehak | 0P2O8H9V8 | ASWI33144D |
| mehak | 0P8V2K0S8 | BOYI31951D |
| mehak | 0U6C1Q0W1 | CKHQ48966D |
| mehak | 0Z2P8W6Z7 | CVNO79910D |
| mehak | 1C1D0H3N7 | DFFC54500D |
| mehak | 1H6D0T0L3 | DJKQ89399D |
| mehak | 1J8X2Q6A6 | DQNR04622D |
| mehak | 1N9C6S6K7 | EBOC86570D |
| mehak | 1P3C4O8C9 | ETPX07262D |
| mehak | 1Q7D5X3T3 | FJCZ08485D |
| mehak | 1W4O7A7Z2 | FJYC34365D |

After



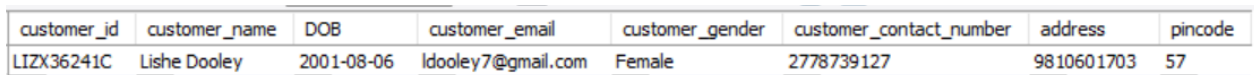
| admin_user_id | payment_ID | agent_id |
|---|---|---|
| mehak | 0F2J9X4C7 | ASWI33144D |
| mehak | 0I5K7A9F4 | GUQS00130D |
| mehak | 0P2O8H9V8 | ASWI33144D |
| mehak | 0P8V2K0S8 | ASWI33144D |
| mehak | 0U6C1Q0W1 | GIWM58062D |
| mehak | 0Z2P8W6Z7 | RBAT80724D |
| mehak | 1C1D0H3N7 | GIWM58062D |
| mehak | 1H6D0T0L3 | DJKQ89399D |
| mehak | 1J8X2Q6A6 | DQNR04622D |
| mehak | 1N9C6S6K7 | CFWF63852D |
| mehak | 1P3C4O8C9 | FJQH64151D |
| mehak | 1Q7D5X3T3 | RBAT80724D |
| mehak | 1W4O7A7Z2 | PWCB31106D |



```
set @customer_name = 'Lishe Dooley';
set @customer_input = 1;
set @customer_input2 = 'ldooley@hotmail.org';
update sql_users.customer
set customer_email = case
when @customer_input = 1 then @customer_input2
else customer_email
end,
 address = case
when @customer_input = 2 then @customer_input2
else address
end,
customer_contact_number= case
when @customer_input=3 then @customer_input2
else customer_contact_number
end
where customer_name = @customer_name;
```

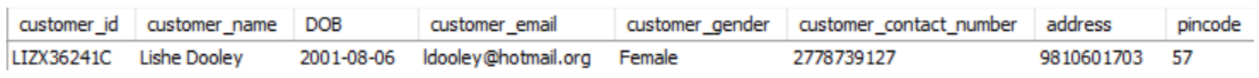
Here the customer details are updated based on user input. The name is given and then the user inputs 1,2 or 3 along with a string. Depending on the number, their email, address and phone number would be updated respectively.

*Before*

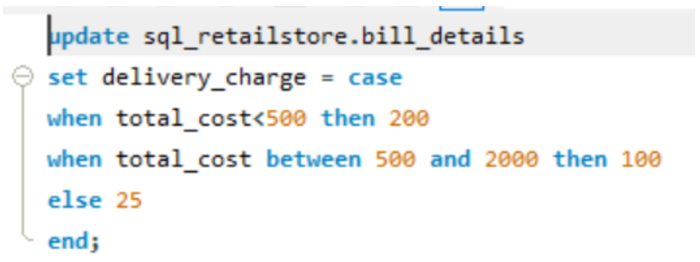


| customer_id | customer_name | DOB | customer_email | customer_gender | customer_contact_number | address | pincode |
|---|---|---|---|---|---|---|---|
| LIZX36241C | Lishe Dooley | 2001-08-06 | ldooley7@gmail.com | Female | 2778739127 | 9810601703 | 57 |

*After*

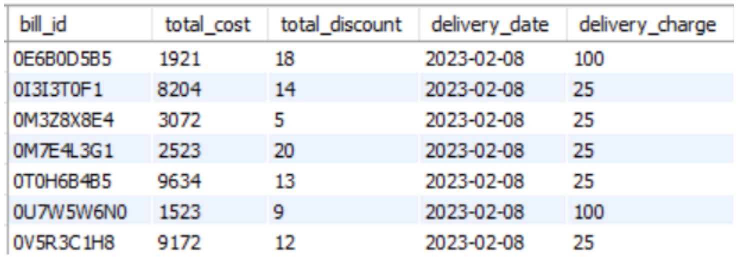


| customer_id | customer_name | DOB | customer_email | customer_gender | customer_contact_number | address | pincode |
|---|---|---|---|---|---|---|---|
| LIZX36241C | Lishe Dooley | 2001-08-06 | ldooley@hotmail.org | Female | 2778739127 | 9810601703 | 57 |



```sql
update sql_retailstore.bill_details
set delivery_charge = case
when total_cost<500 then 200
when total_cost between 500 and 2000 then 100
else 25
end;
```

The delivery charges are updated on depending on the total cost of the order.



| bill_id | total_cost | total_discount | delivery_date | delivery_charge |
|---|---|---|---|---|
| 0E6B0D5B5 | 1921 | 18 | 2023-02-08 | 100 |
| 0I3I3T0F1 | 8204 | 14 | 2023-02-08 | 25 |
| 0M3Z8X8E4 | 3072 | 5 | 2023-02-08 | 25 |
| 0M7E4L3G1 | 2523 | 20 | 2023-02-08 | 25 |
| 0T0H6B4B5 | 9634 | 13 | 2023-02-08 | 25 |
| 0U7W5W6N0 | 1523 | 9 | 2023-02-08 | 100 |
| 0V5R3C1H8 | 9172 | 12 | 2023-02-08 | 25 |