**SQL – IN CLASS LAB EXERCISE – 04**

1. **Write a query to display emp\_name, state, age of an employee whose dept\_id = 1(Use:Sub-query).’**

**ANSWER:**

SELECT emp\_name,state,age

FROM hired h

WHERE emp\_id IN

(SELECT emp\_id

FROM department

WHERE dept\_id = 1);

1. **Write a query to display the dept\_id, maximum salary, provided show the departments whose maximum salary is less than the average salary.(USE:SUB-QUERY)**

**ANSWER:**

SELECT dept\_id, MAX(SALARY)

FROM salary

GROUP BY dept\_id

HAVING MAX(SALARY) < (SELECT AVG(SALARY) FROM salary);

1. **Write a query to display emp\_id,dept\_id of the employees whose salary is higher than emp\_id = 02.(USE:SUB-QUERY)**

**ANSWER:**

select emp\_id,dept\_id from salary where salary > (select salary from salary where emp\_id = 02);

1. **Write a query to display emp\_id,salary of the employees whose salary ranges from minimum salary to 35000.**

**(USE:SUB-QUERY)**

**ANSWER:**

SELECT emp\_id,salary

FROM salary

WHERE salary BETWEEN

(SELECT MIN(salary)

FROM salary) AND 35000;

**Table Name: Product**

|  |  |  |
| --- | --- | --- |
| PRODUCT\_ID | PRODUCT\_NAME | PRICE |
| P01 | BISCUITS | 10 |
| P02 | CHOCOLATES | 20 |
| P03 | BREAD | 15 |
| P04 | BUTTER | 30 |

**Table Name: Sales**

|  |  |
| --- | --- |
| SALES\_ID | PRODUCT\_ID |
| 02 | P02 |
| 01 | P01 |
| 01 | P03 |

**Table Name: Orders**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SALES\_ID** | **Cust\_id** | **PRODUCT\_ID** | **ORDER\_QUANTITY** | **Order\_status** |
| 02 | 101 | P02 | 100 | Shipped |
| 01 | 102 | P01 | 130 | shipped |
| 01 | 103 | P03 | 25 | cancelled |
| 02 | 104 | P01 | 50 | cancelled |

**Table Name: Customer**

|  |  |  |  |
| --- | --- | --- | --- |
| **Cust\_id** | **First\_name** | **Last\_name** | **Sales\_id** |
| 101 | Harry | Dany | 02 |
| 102 | Tom | Adein | 01 |
| 103 | Marina | paul | 01 |
| 104 | peter | kevin | 02 |
| 105 | David | warner |  |

1. **Write a query to display the cust\_id,first\_name,**

**last\_name,sales\_id who have not purchased any product or made any sales.(1 ROW)(USE:SUB-QUERY)**

**ANSWER:-**

select cust\_id,first\_name,last\_name,sales\_id from

customer where cust\_id not in (select cust\_id from

orders );

1. **Write a query to display cust\_id,first\_name,last\_name**

**Of the customers whose second letter start with ‘a’.**

**(USE:SUB-QUERY).**

**ANSWER:**

SELECT cust\_id, first\_name, last\_name FROM customer

WHERE cust\_id IN ( SELECT cust\_id FROM customer

WHERE first\_name LIKE '\_a%' );

1. **Write a query to display the product\_name,product\_id,**

**Price, provided show the second highest priced product.(USE:SUB-QUERY).**

**ANSWER:**

SELECT product\_name, product\_id, price

FROM Product p1 where price =

(SELECT MAX(price) FROM Product p2

WHERE p1.product\_id = p2.product\_id)

ORDER BY price DESC

limit 1,1;

1. **Write a Query to display the product\_id,product\_name and list which of the other products were bought apart from ‘P01’. (Use Exists)(2 ROWS)(USE:SUB-QUERY)**

**ANSWER:-**

SELECT p.product\_id, p.product\_name

FROM Product p INNER JOIN Orders o1

ON p.product\_id = o1.product\_id

WHERE p.product\_id != 'P01'

AND EXISTS

(SELECT 1 FROM Orders o2

WHERE product\_id = 'P01'

AND o1.sales\_id = o2.sales\_id);

1. **Write a query to display the product\_id,product\_name,**

**order\_quantity and list which of the other products were bought apart from ‘P01’ provided their total order\_quantity bought is greater than 75.**

**(1 ROWS)(USE:SUB-QUERY)**

**ANSWER:-**

SELECT p.product\_id, p.product\_name,order\_quantity

FROM Product p INNER JOIN Orders o1

ON p.product\_id = o1.product\_id

WHERE p.product\_id != 'P01'

AND EXISTS

(SELECT 1 FROM Orders o2

WHERE product\_id = 'P01'

AND o1.sales\_id = o2.sales\_id)

GROUP BY p.product\_id, product\_name

HAVING sum(order\_quantity) > 75;

1. **Write a query to display product\_id,product\_name,**

**total\_quantity(sum(order\_quantity) , Provided find the most and least sold products (quantity-wise).**

**(2 ROWS)(USE:SUB-QUERY)**

**ANSWER:-**

SELECT p.product\_id, product\_name, SUM(order\_quantity) AS total\_qty

FROM Product p INNER JOIN Orders oi

ON p.product\_id = oi.product\_id

GROUP BY p.product\_id, product\_name

HAVING SUM(order\_quantity) =

(SELECT SUM(order\_quantity) AS tot\_qty FROM Orders

GROUP BY product\_id ORDER BY tot\_qty DESC LIMIT 1)

OR SUM(order\_quantity) =

(SELECT SUM(order\_quantity) AS tot\_qty FROM Orders

GROUP BY product\_id ORDER BY tot\_qty LIMIT 1);