

furniture category issued and not issued. The query should display the issue status and the number of furniture as **No\_of\_Furnitures**. Take the string for furniture as "Furniture".

- Display the records sorted in ascending order based on issue\_status.
- **Column Name:** issue\_status and No\_of\_Furniture

#### Sample Output

issue_status	No_of_Furniture
N	10
Y	19

2 rows in set (0.00 sec)

#### Note:

- Use the column names as given in the query as they are **case-sensitive**

#### Execution time limit

99 seconds

Having an issue with this question? Report

Oracle

```
1  /*
2  * Enter your query below.
3  * Please append a semicolon ";" at the end of the query
4  */
5
6
7  SELECT issue_status , count(*) as No_of_Furniture
8  FROM item_master
9  WHERE item_category='Furniture'
10 GROUP BY issue_status
11 ORDER BY issue_status ASC;
```

Ln 1, Col 1 Oracle

Test Results

TEST QUERY

► RUN & SUBMIT

» NEXT QUESTION

- Write a query to display the employee id and employee name who joined the company after 2005.
- Display the records sorted in ascending order based on employee id.
- **Column Name:** employee\_id and employee\_name

### Sample Output

```
+-----+-----+
| employee_id | employee_name |
+-----+-----+
| E00082      | Abhay        |
| E00083      | Anita        |
+-----+-----+
2 rows in set (0.00 sec)
```

### Note:

- Use the column names as given in the query as they are **case-sensitive**

### Execution time limit

99 seconds

Having an issue with this question?  Report

Oracle

```
1  /*
2   * Enter your query below.
3   * Please append a semicolon ";" at the end of the query
4   */
5
6  SELECT
7    employee_id,
8    employee_name
9  FROM
10   employee_master
11 WHERE
12   EXTRACT(YEAR FROM date_of_joining)>2005
13 ORDER BY
14   employee_id ASC;
15
```

Test Results

TEST QUERY

RUN & SUBMIT

NEXT QUESTION

Ln 1, Col 1

## Problem Description

- Write a query to display the number of flights flying from each location. The Query should display the from location and the number of flights to other locations as "**No\_of\_Flights**".
- Hint: Get the distinct from location and to location. Display the records sorted in ascending order based on **from location**.
- Columns: **from\_location**, **No\_of\_Flights**.

## Sample Output

from_location	No_of_Flights
chennai	4
delhi	2
kolkata	3

## Note:

- Use the column names as given in the query as they are **case-sensitive**.

Execution time limit

Oracle

```
1  /*
2   * Enter your query below.
3   * Please append a semicolon ";" at the end of the query
4   */
5
6
7  SELECT
8    ... DISTINCT from_location,
9    ... COUNT(*) AS No_of_Flights
10 FROM
11  ... air_flight
12 GROUP BY
13  ... from_location
14 ORDER BY
15  ... from_location ASC;
```

Test Results

TEST QUERY

RUN & SUBMIT

NEXT QUESTION

Ln 1, Col 1

3 rows in set (0.00 sec)

## Problem Description

- Write a query to display the employee id and employee name who doesn't have loan cards.
- Display the records sorted in ascending order based on employee id.
- **Column Name:** employee\_id, employee\_name

## Sample Output

```
+-----+-----+
| employee_id | employee_name |
+-----+-----+
| E00014      | Zubeen       |
| E00015      | Rashika      |
+-----+-----+
2 rows in set (0.01 sec)
```

### Note:

- Use the column names as given in the query as they are **case-sensitive**

## Execution time limit

99 seconds

Having an issue with this question?  Report

Oracle

```
1  /*
2   * Enter your query below.
3   * Please append a semicolon ";" at the end of the query
4   */
5
6  SELECT
7    em.employee_id,
8    em.employee_name
9  FROM
10   employee_master em
11  LEFT JOIN
12    employee_card_details ecd ON em.employee_id = ecd.employee_id
13  WHERE
14    ecd.loan_id IS NULL
15  ORDER BY
16    em.employee_id ASC;
17
```

Ln 1, Col 1 Oracle

Test Results

TEST QUERY

▶ RUN & SUBMIT

» NEXT QUESTION

STOMER\_DATE\_OF\_BIRTH | date

s | | NULL | |

+-----+

+-----+-----+

ows in set (0.00 sec)

## Item Description

Write a query to display the number of customers from Delhi(i.e., the city name is 'DELHI'). Give the count an alias name of Cust\_Count.

- **Column Name:** Cust\_Count

## Sample Output

Cust\_Count

+-----

7

## Note:

- Use the column names as given in the query as they are **case-sensitive**.

## Execution time limit

99 seconds

Oracle

```
1 /*  
2  * Enter your query below.  
3  * Please append a semicolon ";" at the end of the query  
4  */  
5  
6 SELECT COUNT(*) as Cust_Count FROM customer_master  
7 WHERE CUSTOMER_CITY='DELHI';  
8
```

Test Results

TEST QUERY

RUN & SUBMIT

NEXT

34m 09s  
left

Help

All

	CUSTOMER_CITY	varchar(15)
1	YES	NULL
2	CUSTOMER_CONTACT_NO	varchar(10)
3	YES	NULL
4	occupation	varchar(10)
5	YES	NULL
6	CUSTOMER_DATE_OF_BIRTH	date
7	YES	NULL

+-----+-----+  
+-----+-----+  
+-----+-----+  
8 rows in set (0.00 sec)

## Problem Description

- Write a query to display the account number, customer's number, customer's firstname, lastname, and account opening date.
- Display the records sorted in ascending order based on account number.  

- **Column Name:** account\_number, customer\_number, firstname, lastname, account\_opening\_date.

## Sample Output

	CUSTOMER_CITY	varchar(15)
1	YES	NULL
2	CUSTOMER_CONTACT_NO	varchar(10)
3	YES	NULL
4	occupation	varchar(10)
5	YES	NULL
6	CUSTOMER_DATE_OF_BIRTH	date
7	YES	NULL

+-----+-----+  
+-----+-----+  
+-----+-----+  
+-----+-----+

Oracle

```
1  /*
2  * Enter your query below.
3  * Please append a semicolon ";" at the end of the query
4  */
5  SELECT
6    am.account_number,
7    cm.CUSTOMER_NUMBER,
8    cm.FIRSTNAME,
9    cm.lastname,
10   am.account_opening_date
11  FROM
12    account_master am
13  JOIN
14    customer_master cm
15  ON am.customer_number = cm.CUSTOMER_NUMBER
16  ORDER BY
17    am.account_number ASC;
```

Test Results

TEST QUERY

▶ RUN &amp; SUBMIT

» NEXT QUES

Ln 1, Co

## Problem Description

- Write a query to display the distinct customer id, customer name, card id, card description, and card amount in dollars of customers who have not taken a movie on the same day the library card is registered.
- For Example, Assume John registered a library card on 12th Jan 2013 and took a movie on 12th Jan 2013 then display his details.

AMOUNT\_DOLLAR = amount/52 and round it to zero decimal places and display it as \$Amount.

For Example, Assuming 500 is the amount then the dollar value will be \$10.

- Hint: Use AMOUNT\_DOLLAR as an alias name for the amount in dollars
- Display the records in ascending order based on the customer name.
- **Column Name:** customer\_id, customer\_name, card\_id, DESCRIPTION, AMOUNT\_DOLLAR

## Sample Output

Oracle

```
1  /*
2   * Enter your query below.
3   * Please append a semicolon ";" at the end of the query
4   */
5
6  SELECT
7    DISTINCT cm.CUSTOMER_ID , cm.CUSTOMER_NAME ,
8    ccd.CARD_ID AS card_id,lcm.DESCRIPTION,lcm.AMOUNT AS AMOUNT_DOLLAR
9  FROM
10   CUSTOMER_MASTER cm JOIN CUSTOMER_ISSUE_DETAILS cid ON cm.CUSTOMER_ID = cid.
11   CUSTOMER_ID JOIN CUSTOMER_CARD_DETAILS ccd ON cid.CUSTOMER_ID=ccd.CUSTOMER_ID.
12   JOIN LIBRARY_CARD_MASTER lcm ON ccd.CARD_ID = lcm.CARD_ID
13
14 WHERE
15   NOT EXISTS(
16     SELECT 1
17     FROM
18   )
19 ORDER BY
20   cm.CUSTOMER_NAME ASC;
21
```

Ln 1, Col 1 Oracle

Test Results

TEST QUERY

► RUN & SUBMIT

» NEXT QUESTION

6 rows in set (0.00 sec)

## Problem Description

- Write a query to find the number of items in each category, make, and description. The Query should display the Item Category, Make, description, and the number of items as **No\_of\_Items**.

- Display the records in ascending order based on Item Category, then by item make, and then by item description.
- **Column Name:** item\_category, item\_make, item\_description, and No\_of\_items

## Sample Output

item_category	item_make	item_description	No_of_items
Crockery Set	Bonechina	Dinning	1
Crockery	Bonechina	Tea Set	1
Furniture	Steel	Cupboard	

Oracle

```
1  /*
2   * Enter your query below.
3   * Please append a semicolon ";" at the end of the query
4   */
5
6   SELECT
7     item_category, item_make, item_description,
8     count(*) as No_of_Items
9   FROM
10    item_master
11  GROUP BY
12    item_category,
13    item_make,
14    item_description
15
16    ORDER BY
17    item_category, item_make, item_description;
```

Test Results

TEST QUERY

RUN & SUBMIT

NEXT QUESTION

Ln 1, Col 1

## Problem Description

- Write a query to display the flight\_id, from\_location,to\_location, number of services as "No\_of\_Service", and average ticket price as "Average\_Price" whose average ticket price is greater than the total average ticket cost of all flights.
- Order the result at the lowest average price.
- **Column Name:** flight\_id, from\_location, to\_location, No\_of\_Service, Average\_Price

## Sample Output

flight_id	from_location	to_location	No_of_Service	Average_Price
3175	chennai	hyderabad	1	3500

Oracle

```
1  /*
2   * Enter your query below.
3   * Please append a semicolon ";" at the end of the query
4   */
5
6  SELECT
7    af.flight_id,af.from_location,af.to_location,COUNT(af.flight_id) as
8    No_of_Service,
9    AVG(price) AS Average_Price
10   FROM
11     air_flight_af JOIN air_flight_details_afd
12   ON
13     af.flight_id = afd.flight_id
14   GROUP BY
15     af.flight_id,af.from_location,af.to_location
16   HAVING
17     AVG(afd.price)>(
18       SELECT AVG(price) from air_flight_details
19     )
20   ORDER BY
21     Average_Price ASC;
```

Test Results

TEST QUERY

RUN & SUBMIT

> NEXT QUES

## Problem Description

- Write a query to display the director's name, movie name, and lead\_actor\_name1 as of all the movies directed by the director who directed at least one movie.
- Display the director's name in capital letters. Use DIRECTOR\_NAME as an alias name for the director name column.
- Display the records sorted in ascending order based on director\_name and then by movie\_name in descending order.
- **Column Name:** DIRECTOR\_NAME, movie\_name and lead\_role\_1

## Sample Output

```
+-----+-----+  
|-----+  
| DIRECTOR_NAME | movie_name |  
| lead_role_1 |
```

```
5  
6 · SELECT · UPPER(DIRECTOR) · as · DIRECTOR_NAME, · MOVIE_NAME,LEAD_ROLE_1  
7 · FROM · MOVIES_MASTER  
8 · ORDER · BY · DIRECTOR · ASC · , · MOVIE_NAME · DESC;
```

Test Results

TEST QUERY

▶ RUN & SUBMIT

» NEXT QUES

Ln 1, C



33m 12s

left



Help

All

5

6

7

8

9

10

11

12

```
+-----+  
| 6 rows in set (0.01 sec)
```

## Problem Description

- Write a query to display the employee id and employee name who was issued an item of the highest valuation. In case of multiple records, display the records sorted in ascending order based on employee id.
- [Hint Suppose the cost of an item called a dining table is Rs. 22000 and that is the highest price of the item that has been issued. So, display the employee id and employee name who issued the dining table whose price is 22000.]

- Column Name:** employee\_id, employee\_name



## Sample Output

```
+-----+  
| employee_id | employee_name |  
+-----+-----+  
| E00014      | Zaara        |  
+-----+-----+  
1 row in set (0.00 sec)
```

**Note:**

Oracle

```
1  /*  
2   * Enter your query below.  
3   * Please append a semicolon ";" at the end of the query  
4   */  
5  
6  SELECT  
7    em.employee_id,em.employee_name  
8  FROM  
9    employee_master em  
10   JOIN employee_issue_details eid ON em.employee_id = eid.employee_id  
11   JOIN item_master im ON eid.item_id = im.item_id  
12 WHERE  
13   im.item_valuation = (   
14     SELECT MAX(item_valuation)  
15     FROM item_master  
16   )  
17 ORDER BY  
18   em.employee_id ASC;
```

Test Results

TEST QUERY

▶ RUN &amp; SUBMIT

» NEXT Q

5 rows in set (0.01 sec)

## Problem Description

- Write a query to display the employee id, employee name, and total valuation of the product issued to each employee.
- Give the alias name TOTAL\_VALUATION.
- Display the records sorted in ascending order based on employee id.
- Consider only employees who have been issued at least 1 item.
- **Column Name:** employee\_id, employee\_name, and TOTAL\_VALUATION

## Sample Output

employee_id	employee_name	TOTAL_VALUATION
E00021	Ram	7000
E00022	Abhay	

Oracle

```
1  /*
2  * Enter your query below.
3  * Please append a semicolon ";" at the end of the query
4  */
5
6  SELECT
7      em.employee_id,
8      em.employee_name,
9      SUM(im.item_valuation) AS TOTAL_VALUATION
10
11 FROM
12     employee_master em
13     JOIN employee_issue_details eid ON em.employee_id = eid.employee_id
14     JOIN item_master im ON eid.item_id = im.item_id;
15 WHERE
16     eid.item_id IS NOT NULL
17 GROUP BY
18     em.employee_id,
19     em.employee_name
20 ORDER BY
21     em.employee_id ASC;
```

Test Results

TEST QUERY

RUN & SUBMIT

NEXT QU