

Q1. The Most Recent Three Orders

Context:

NextGen Retail is a leading e-commerce platform known for its exceptional customer service and wide range of products and is committed to delivering a superior online shopping experience through data-driven insights and personalized customer service. The company aims to enhance customer satisfaction by analyzing order patterns and understanding the purchasing behaviour of its customers. This helps NextGen Retail to tailor their marketing strategies and provide personalized recommendations.

You are a data analyst at NextGen Retail. Your manager has requested a report on the most recent three orders for each customer. This will help the company identify active customers and understand their recent purchasing trends.

Problem Statement:

Write a query to find the most recent three orders of each user. If a user ordered three or less than three orders, return all of their orders.

Result:

- Return the result table ordered by customer_name in **ascending order** and in case of a tie by the customer_id in **ascending order**. If there is still a tie, order them by order_date in **descending order**.

Sample Input:

Table: customers

customer_id	name
1	Winston
2	Jonathan
3	Annabelle
4	Marwan
5	Khaled

Table: orders

order_id	order_date	customer_id	cost
1	2020-07-31	1	30
2	2020-07-30	2	40
3	2020-07-31	3	70
4	2020-07-29	4	100
5	2020-06-10	1	1010
6	2020-08-01	2	102
7	2020-08-01	3	111
8	2020-08-03	1	99
9	2020-08-07	2	32
10	2020-07-15	1	2

Sample output:

customer_name	customer_id	order_id	order_date
Annabelle	3	7	2020-08-01
Annabelle	3	3	2020-07-31
Jonathan	2	9	2020-08-07
Jonathan	2	6	2020-08-01
Jonathan	2	2	2020-07-30
Marwan	4	4	2020-07-29
Winston	1	8	2020-08-03
Winston	1	1	2020-07-31
Winston	1	10	2020-07-15

Explanation:

- We will return the data by ascending order of name
- We will start with Annabelle, she has only 2 orders, we return them.
- Jonathan has exactly 3 orders.
- Marwan ordered only one time.
- Winston has 4 orders, we discard the order of "2020-06-10" because it is the oldest order.

Q2. Most Recent – Use hr dataset

Context:

At Innovate Solutions, a forward-thinking technology firm, tracking the most recent job roles of employees is crucial for understanding their career development and current roles. Accurate records of job start dates help in strategic HR planning and managing employee transitions effectively.

As a data analyst at Innovate Solutions, you are tasked with generating a report that identifies the start date of the most recent job for each employee. This information will assist the HR department in maintaining up-to-date records and making informed decisions regarding employee management.

Problem Statement:

Write a Query to find the **first day of the most recent job** of every employee and return it as the 'recent_job'. Refer to the job_history table to get the job details of the employees.

Result:

- Return the columns **first_name** and **recent_job**.
- Return the output ordered by **first_name** in ascending order.

Sample Input:

Table: employees

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	commission_pct	manager_id	department_id
101	Neena	Kochhar	NKOCHHAR	515.123.4568	1989-09-21	AD_VP	17000	NULL	100	90
102	Lex	De Haan	LDEHAAN	515.123.4569	1993-01-13	AD_VP	17000	NULL	100	90
103	Alexander	Hunold	AHUNOLD	590.423.4567	1990-01-03	IT_PROG	9000	NULL	102	60
105	David	Austin	DAUSTIN	590.423.4569	1997-06-25	IT_PROG	4800	NULL	103	60
115	Alexander	Khoo	AKHOO	515.127.4562	1995-05-18	PU_CLERK	3100	NULL	114	30
151	David	Bernstein	DBERNSTE	011.44.1344.345268	1997-03-24	SA_REP	9500	0.25	145	80
165	David	Lee	DLEE	011.44.1346.529268	2000-02-23	SA_REP	6800	0.1	147	80

Table: job_history

employee_id	start_date	end_date	job_id	department_id
101	1989-09-21	1993-10-27	AC_ACCOUNT	110
101	1993-10-28	1997-03-15	AC_MGR	110
102	1993-01-13	1998-07-24	IT_PROG	60
200	1987-09-17	1993-06-17	AD_ASST	90
200	1994-07-01	1998-12-31	AC_ACCOUNT	90

Sample Output:

first_name	recent_job
Lex	1993-01-13
Neena	1993-10-28

Sample Explanation:

- The output lists each employee's first name along with the start date of their most recent job.
- The results are ordered by first_name in ascending order.

Q3. First job salary – Use hr dataset

Context:

At DataXperts Inc., understanding the initial compensation trends of employees is vital for analyzing career progression and salary growth patterns. By identifying the starting maximum salary of the first job held by each employee, the HR department can gain insights into hiring practices and market competitiveness.

As a data analyst at DataXperts Inc., your task is to generate a report that lists the starting maximum salary of the first job that each employee held. This information will help in evaluating the company's initial salary offerings and comparing them with industry standards.

Problem Statement:

Write a query to find the starting **maximum salary of the first job** that every employee held and return it as 'first_job_sal'.

Result:

- Return the columns 'first_name', 'last_name', 'first_job_sal'.
- Return the result sorted by **first_name** in ascending order.

Note:

- Refer to the job_history table to get the job details of the employees.
- Refer to the employees table for first_name and last_name.
- Refer to the jobs table for the maximum salary.

Sample Input:

Table: employees

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	commission_pct	manager_id	department_id
114	Den	Raphaely	DRAPHEAL	515.127.4561	1994-12-07	PU_MAN	11000	NULL	100	30
116	Shelli	Baida	SBAIDA	515.127.4563	1997-12-24	PU_CLERK	2900	NULL	114	30
118	Guy	Himuro	GHIMURO	515.127.4565	1998-11-15	PU_CLERK	2600	NULL	114	30
134	Michael	Rogers	MROGERS	650.127.1834	1998-08-26	ST_CLERK	2900	NULL	122	50
178	Kimberely	Grant	KGRANT	011.44.1644.429263	1999-05-24	SA_REP	7000	0.15	149	NULL
189	Jennifer	Dilly	JDILLY	650.505.2876	1997-08-13	SH_CLERK	3600	NULL	122	50
200	Jennifer	Whalen	JWHALEN	515.123.4444	1987-09-17	AD_ASST	4400	NULL	101	10
201	Michael	Hartstein	MHARTSTE	515.123.5555	1996-02-17	MK_MAN	13000	NULL	100	20
202	Pat	Fay	PFAY	603.123.6666	1997-08-17	MK_REP	6000	NULL	201	20
203	Susan	Mavris	SMAVRIS	515.123.7777	1994-06-07	HR_REP	6500	NULL	101	40

Table: jobs

job_id	job_title	min_salary	max_salary
AD_ASST	Administration Assistant	3000	6000
HR_REP	Human Resources Representative	4000	9000
MK_MAN	Marketing Manager	9000	15000
MK_REP	Marketing Representative	4000	9000
PU_CLERK	Purchasing Clerk	2500	5500
PU_MAN	Purchasing Manager	8000	15000
SA_REP	Sales Representative	6000	12000
SH_CLERK	Shipping Clerk	2500	5500
ST_CLERK	Stock Clerk	2000	5000

Table: job_history

employee_id	start_date	end_date	job_id	department_id
101	1989-09-21	1993-10-27	AC_ACCOUNT	110
101	1993-10-28	1997-03-15	AC_MGR	110
102	1993-01-13	1998-07-24	IT_PROG	60
114	1998-03-24	1999-12-31	ST_CLERK	50
122	1999-01-01	1999-12-31	ST_CLERK	50
176	1998-03-24	1998-12-31	SA_REP	80
176	1999-01-01	1999-12-31	SA_MAN	80
200	1987-09-17	1993-06-17	AD_ASST	90
200	1994-07-01	1998-12-31	AC_ACCOUNT	90
201	1996-02-27	1999-12-19	MK_REP	20

Sample Output:

first_name	last_name	first_job_sal
Den	Raphaely	5000
Jennifer	Whalen	6000
Michael	Hartstein	9000

Sample Explanation:

- The output lists each employee's first name, last name, and the starting maximum salary of their first job.
- The results are sorted by first_name in ascending order.

Q4. First day job – Use hr dataset

Context:

At Enterprise HR Solutions, understanding employee career progression is crucial for talent management and planning. Tracking the start dates of employees' first jobs within the company helps in analyzing tenure, growth, and retention strategies.

As a data analyst at Enterprise HR Solutions, you are tasked with generating a report that lists the start date of the first job for every employee. This report will assist the HR team in workforce planning and understanding employee tenure.

Problem Statement:

Write a query to find the **first day of the first job** of every employee and return it as 'first_day_job'.

Result:

- Return the columns '**first_name**', and '**first_day_job**'.
- Return the result ordered by **first_name** in ascending order.

Sample Input:

Table: employees

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	commission_pct	manager_id	department_id
114	Den	Raphaely	DRAPHEAL	515.127.4561	1994-12-07	PU_MAN	11000	NULL	100	30
125	Julia	Nayer	JNAYER	650.124.1214	1997-07-16	ST_CLERK	3200	NULL	120	50
133	Jason	Mallin	JMALLIN	650.127.1934	1996-06-14	ST_CLERK	3300	NULL	122	50
134	Michael	Rogers	MROGERS	650.127.1834	1998-08-26	ST_CLERK	2900	NULL	122	50
141	Trenna	Rajs	TRAJS	650.121.8009	1995-10-17	ST_CLERK	3500	NULL	124	50
186	Julia	Dellinger	JDELLING	650.509.3876	1998-06-24	SH_CLERK	3400	NULL	121	50
189	Jennifer	Dilly	JDILLY	650.505.2876	1997-08-13	SH_CLERK	3600	NULL	122	50
200	Jennifer	Whalen	JWHALEN	515.123.4444	1987-09-17	AD_ASST	4400	NULL	101	10
201	Michael	Hartstein	MHARTSTE	515.123.5555	1996-02-17	MK_MAN	13000	NULL	100	20
202	Pat	Fay	PFAY	603.123.6666	1997-08-17	MK_REP	6000	NULL	201	20

Table: job_history

employee_id	start_date	end_date	job_id	department_id
101	1989-09-21	1993-10-27	AC_ACCOUNT	110
101	1993-10-28	1997-03-15	AC_MGR	110
102	1993-01-13	1998-07-24	IT_PROG	60
114	1998-03-24	1999-12-31	ST_CLERK	50
122	1999-01-01	1999-12-31	ST_CLERK	50
176	1998-03-24	1998-12-31	SA_REP	80
176	1999-01-01	1999-12-31	SA_MAN	80
200	1987-09-17	1993-06-17	AD_ASST	90
200	1994-07-01	1998-12-31	AC_ACCOUNT	90
201	1996-02-27	1999-12-19	MK_REP	20

Sample Output:

first_name	first_day_job
Den	1998-03-24
Jennifer	1987-09-17
Michael	1996-02-27

Sample Explanation:

The output lists each employee's first name along with the start date of their first job.

Q5. Market Analysis II

Problem Statement:

Write a query to find for each user whether the brand of the **second item** (by date) they sold is their **favorite brand**. If a user sold **less than** two items, report the answer for that user as **no**. It is guaranteed that no seller sold more than one item on a day.

- Return the result table ordered by seller_id in ascending order.

Sample Input:

Table: users

user_id	join_date	favorite_brand
1	2019-01-01	Lenovo
2	2019-02-09	Samsung
3	2019-01-19	LG
4	2019-05-21	HP

Table: orders2

order_id	order_date	item_id	buyer_id	seller_id
1	2019-08-01	4	1	2
2	2019-08-02	2	1	3
3	2019-08-03	3	2	3
4	2019-08-04	1	4	2
5	2019-08-04	1	3	4
6	2019-08-05	2	2	4

Table: items

item_id	item_brand
1	Samsung
2	Lenovo
3	LG
4	HP

Sample output:

seller_id	2nd_item_fav_brand
1	no
2	yes
3	yes
4	no

Explanation:

- The answer for the user with id 1 is no because they sold nothing.
- The answer for the users with id 2 and 3 is yes because the brands of their second sold items are their favorite brands.
- The answer for the user with id 4 is no because the brand of their second sold item is not their favorite brand.