

Relational algebra

1.	$\pi \rho_{in_stock} (\pi \text{count}(\text{book.book_id}) (\sigma(\text{book.book_name} = "?" \vee \text{author.first_name} = "?" \vee \text{author.last_name} = "?") (\text{book} \bowtie \text{book_author} \bowtie \text{author} \bowtie \text{book_storage})))$	select (case when <i>count</i> (b.book_id) > 0 then true else false end) as in_stock from book b join book_author ba using (book_id) join author a using (author_id) join book_storage where b.book_name like '%? %' or a.first_name like '%?%' or a.last_name like '%?%'
2	$\pi \text{first_name, last_name} (\sigma (\text{client})) \tau \text{created_date}$	select first_name, last_name from client order by created_date limit 1;
3	$\pi \text{book.book_name, editorial_book.publication_year, book_copy.weight, book_copy.amount_pages, book_translation.language_name, book_storage.start_date} (\sigma \neg \text{book_storage.end_date} = \text{null}) (\text{book_copy} \bowtie \text{book_storage} \bowtie \text{editorial_book} \bowtie \text{book} \bowtie \text{book_translation}) \tau \text{book_storage.start_date}$	select b.book_name, eb.publication_year, weight, amount_pages, bt.language_name, bs.start_date from book_copy join book_storage bs using (book_copy_id) join editorial_book eb using (editorial_book_id) join book b using (book_id) join book_translation bt using (book_id) where end_date is null order by bs.start_date limit 1;
4	$\pi \text{book.book_name, client.first_name, client.last_name, oorder.created_date} ((\sigma \text{communication.communication_id} = \text{null}) (\text{oorder} \bowtie \text{communication} \bowtie \text{book} \bowtie \text{client}) \gamma \text{created_date}))$	select b.book_name, c2.first_name, last_name, oo.created_date from oorder oo left join communication c using (order_id) join book b using (book_id) join client c2 using (client_id) where communication_id is null order by oo.created_date;
5	$\sigma \text{count}(\text{book_sale.sale_id}) (\sigma(\text{book.book_name} = "?" \vee \text{author.first_name} = "?" \vee \text{author.last_name} = "?") (\text{sale} \bowtie \text{book_sale} \bowtie \text{book_copy} \bowtie \text{editorial_book} \bowtie \text{book} \bowtie \text{book_author} \bowtie \text{author}))$	select <i>count</i> (bs.sale_id) from sale join book_sale bs using (sale_id) join book_copy copy2 using (book_copy_id) join editorial_book eb using (editorial_book_id) join book b using (book_id) join book_author ba using (book_id) join author a using (author_id) where b.book_name like '%?%' or a.first_name like '%?%' or a.last_name like '%?%';

6	π author.first_name, author.last_name ((σ sale.created_date > '?' \wedge sale.created_date < '?') (sale \bowtie book_sale \bowtie book_copy \bowtie editorial_book \bowtie book_author \bowtie author)) γ author.author_id τ count(book_sale.book_sale_id)	select a.first_name, a.last_name from sale join book_sale bs using (sale_id) join book_copy copy2 using (book_copy_id) join editorial_book book using (editorial_book_id) join book_author ba using (book_id) join author a using (author_id) where created_date > ? and created_date < ? group by author_id order by count(book_sale_id) desc limit 1;
7	π client.first_name, client.last_name (sale \bowtie client) γ client.client_id τ count (sale.sale_id)	select c.first_name, c.last_name from sale join client c using (client_id) group by c.client_id order by count(sale_id) desc limit 3;
8	π book.book_name (book \bowtie book_translation) γ book.book_id τ count (book.book_id)	select b.book_name from book b join book_translation translation using (book_id) group by b.book_id order by count(b.book_id) desc limit 1;
9	π sale.created_date, ρ_{final_price} (book_copy.original_price + book_copy.profit), book.book_name ((σ client.first_name = '?' \wedge client.last_name = '?') (sale \bowtie client \bowtie book_sale \bowtie book_copy \bowtie editorial_book \bowtie book)) τ client.created_date	select s.created_date, (copy2.original_price + copy2.profit) as final_price, b.book_name from sale s join client c using (client_id) join book_sale using (sale_id) join book_copy copy2 using (book_copy_id) join editorial_book book2 using (editorial_book_id) join book b using (book_id) where first_name = '?' and last_name = ?' order by c.created_date;
10	1. $\rho_{orders_with_com}$ (π oorder.order_id (σ client_id = '?' (oorder \bowtie communication))) 2. $\rho_{orders_with_sale}$ (π oorder.order_id (σ client_id = '?' \wedge sale.created_date > oorder.created_date (oorder \bowtie sale \bowtie book_sale \bowtie book_copy \bowtie editorial_book))) 3. π book.book_name, oorder.created_date, ρ_{was_found} (σ order_id != null (orders_with_com)), $\rho_{was_aquired}$ (σ order_id != null (orders_with_sale)) ((σ oorder.client_id = '?') (oorder \bowtie orders_with_com \bowtie orders_with_sale \bowtie book))	select book_name, created_date, (CASE WHEN orders_with_com.order_id is null THEN false ELSE true END) as was_found, (CASE WHEN orders_with_sale.order_id is null THEN false ELSE true END) as was_aquired from oorder left join (select oorder.order_id

		<pre> from oorder join communication c on oorder.order_id = c.order_id where client_id = 1) as orders_with_com using (order_id) left join (select oo.order_id from oorder oo join sale s using (client_id) join book_sale bs using (sale_id) join book_copy copy2 using (book_copy_id) join editorial_book book2 using (editorial_book_id) where s.created_date >= oo.created_date and oo.client_id = ?) as orders_with_sale using (order_id) join book b on oorder.book_id = b.book_id where client_id = ?; </pre>
11	<ol style="list-style-type: none"> 1. $\rho_{\text{delivery_by_book}}(\sigma(\text{book} \times \text{delivery_type}))$ 2. $\pi \text{ delivery_by_book.book_name, delivery_by_book.delivery_name, delivery_by_book.company_name, } \rho_{\text{delivery_price}}(\text{delivery_by_book.price_per_kilo} * \text{editorial_book.weight}) (\text{delivery_by_book} \bowtie \text{editorial_book})$ 	<pre> select delivery_by_book.book_name, delivery_by_book.delivery_name, delivery_by_book.company_name, TRUNCATE((delivery_by_book.price_per_kilo * eb.weight), 2) as delivery_price from (select * from book b, delivery_type d) delivery_by_book join editorial_book eb using (book_id); </pre>
12	<ol style="list-style-type: none"> 1. $\rho_{\text{splitted_deliveries}}(\pi \text{ count(book_sale.book_sale_id), sale.sale_id } (\sigma \text{ sale.client_id} = '?' (\text{book_sale} \bowtie \text{book_sale_delivery} \bowtie \text{delivery} \bowtie \text{sale})))$ 2. $\pi \text{ book_name, tracking_number, delivery_name, company_name, delivery_status_name } (\text{delivery} \bowtie \text{book_sale_delivery} \bowtie \text{delivery_type} \bowtie \text{delivery_status} \bowtie \text{book_sale} \bowtie \text{book_copy} \bowtie \text{editorial_book} \bowtie \text{book} \bowtie \text{splitted_deliveries})$ 	<pre> select book_name, tracking_number, delivery_name, company_name, delivery_status_name from delivery join book_sale_delivery delivery2 on delivery.delivery_id = delivery2.delivery_id join delivery_type t on delivery.delivery_type_id = t.delivery_type_id join delivery_status s on delivery.delivery_status_id = s.delivery_status_id join book_sale sale on delivery2.book_sale_id = sale.book_sale_id join book_copy copy2 on sale.book_copy_id = copy2.book_copy_id join editorial_book book2 using (editorial_book_id) join book b using (book_id) join (select count(bs.book_sale_id), sale_id from book_sale bs join book_sale_delivery delivery on bs.book_sale_id = delivery.book_sale_id join delivery d on delivery.delivery_id = </pre>

		d.delivery_id join sale using (sale_id) where client_id = ? group by sale_id having count(bs.book_sale_id) > 1) splitted_deliveries using (sale_id);
13	π delivery_status.delivery_status_name (σ delivery.tracking_number = '?' (delivery_status \bowtie delivery))	select delivery_status_name from delivery_status join delivery d using (delivery_status_id) where tracking_number = '?';
14	π count(*) (σ delivery.company_name = 'XPress' \wedge month(sale.created_date) = '?' (book_sale_delivery \bowtie book_sale \bowtie delivery \bowtie delivery_type \bowtie sale))	select count(*) from book_sale_delivery join book_sale sale on book_sale_delivery.book_sale_id = sale.book_sale_id join delivery d on book_sale_delivery.delivery_id = d.delivery_id join delivery_type t on d.delivery_type_id = t.delivery_type_id join sale s on sale.sale_id = s.sale_id where company_name = 'XPress' and month(s.created_date) = ?
15	π ρ_{total} (sum(profit + original_price)) (σ payment_type = 'Bit' \wedge month(sale.created_date) = ? (book_sale \bowtie book_copy \bowtie sale \bowtie payment_type))	select TRUNCATE(IFNULL(sum(profit + original_price), 0), 2) as total from book_sale bs join book_copy copy2 on bs.book_copy_id = copy2.book_copy_id join sale s using (sale_id) join payment_type t2 using (payment_type_id) where payment_name = 'Bit' and month(s.created_date) = ?;
16	1. $\rho_{average}$ (π avg(profit) (book_sale \bowtie book_copy \bowtie sale \bowtie payment_type)) 2. π book.book_name, client.first_name, client.last_name, (ρ_{final_price} (book_copy.profit + book_copy.original_price)) ((σ sale.created_date < NOW \wedge sale.created_date > NOW - 12 MONTHS \wedge book_copy.profit > average))	select book_name, CONCAT(CONCAT(first_name, ' '), last_name), (profit + original_price) as final_price from book_sale bs join book_copy copy2 using (book_copy_id) join sale s using (sale_id) join payment_type t2 using (payment_type_id) join editorial_book eb using (editorial_book_id) join book b using (book_id) join client c using (client_id) where s.created_date < NOW() and s.created_date > DATE_ADD(NOW(), INTERVAL -12 MONTH) and profit > (select avg(profit) from book_sale bs join book_copy copy2 on bs.book_copy_id = copy2.book_copy_id join sale s using (sale_id) join payment_type t2 using (payment_type_id));
17	π delivery.company_name, count(book_sale_delivery.book_delivery_id)	select company_name,

	$((\sigma \text{ sale.created_date} < \text{NOW} \text{ and } \text{sale.created_date} > \text{NOW} - 12 \text{ MONTH } \vee \text{ delivery.company_name})$ $(\text{book_sale_delivery} \bowtie \text{book_sale} \bowtie \text{delivery} \bowtie \text{book_copy} \bowtie \text{editorial_book} \bowtie \text{sale}))$	$\text{count}(\text{book_delivery_id})$ from book_sale_delivery bs join book_sale sale on bs.book_sale_id = sale.book_sale_id join delivery d on bs.delivery_id = d.delivery_id join delivery_type t on d.delivery_type_id = t.delivery_type_id join book_copy copy2 on sale.book_copy_id = copy2.book_copy_id join editorial_book book2 using (editorial_book_id) join sale s on sale.sale_id = s.sale_id where s.created_date < NOW() and s.created_date > DATE_ADD(NOW(), INTERVAL -12 MONTH) group by company_name;
18	1. $\rho_{\text{multiple_edition_deliveries}}(\pi \text{ count}(\text{distinct book.editorial_book_id}), \text{book_id}, \text{book_sale_delivery.delivery_id} (\sigma (\text{delivery} \bowtie \text{book_sale_delivery} \bowtie \text{book_sale} \bowtie \text{book_copy} \bowtie \text{editorial_book}) \vee \text{book.book_id}, \text{book_sale_delivery.delivery_id}))$ 2. $\pi \text{ book.book_name}, \text{delivery.tracking_number}, \text{delivery_status.delivery_status_name}, \text{delivery_type.delivey_name}, \text{delivery_type.company_name} (\text{delivery} \bowtie \text{multiple_edition_deliveries} \bowtie \text{delivery_status} \bowtie \text{delivery_type} \bowtie \text{book})$	select book_name, tracking_number, delivery_status_name, delivery_name, company_name from delivery join (select count(distinct book.editorial_book_id), book_id, delivery2.delivery_id from delivery join book_sale_delivery delivery2 on delivery.delivery_id = delivery2.delivery_id join book_sale sale on delivery2.book_sale_id = sale.book_sale_id join book_copy copy2 on sale.book_copy_id = copy2.book_copy_id join editorial_book book on copy2.editorial_book_id = book.editorial_book_id group by book_id, delivery2.delivery_id having count(distinct book.editorial_book_id) > 1) multiple_edition_deliveries using (delivery_id) join delivery_status s using (delivery_status_id) join delivery_type t using (delivery_type_id) join book using (book_id);
19	$\pi \text{ client.first_name}, \text{client.last_name}, \text{client.mobile_number}, \text{client.created_date} (\sigma \text{ client_id} - (\pi \text{ client_id} (\sigma \text{ created_date} < \text{NOW} \wedge \text{created_date} > \text{NOW} - 24 \text{ MONTHS} (\text{sale}))))$	select CONCAT(CONCAT(first_name, ' '), last_name), phone_number, mobile_number, created_date from client where client_id not in (select client_id from sale s where s.created_date < NOW() and s.created_date > DATE_ADD(NOW(), INTERVAL -24 MONTH));

20	π client.first_name, client.last_name, client.mobile_number, created_date (σ client.client_id \cap (π oorder.client_id (σ communication.created_date < NOW - 14 DAYS \wedge bought_book.book_id = editorial_book.book_id \wedge sale.client_id = oorder.client_id \wedge sale.created_date \geq communication.created_date (communication \bowtie oorder \bowtie editorial_book \bowtie book_copy \bowtie book_sale \bowtie pbought_book(editorial_book) \bowtie sale))))	select CONCAT(CONCAT(first_name, ' '), last_name), phone_number, mobile_number, created_date from client where client_id in (select o.client_id from communication c join oorder o using (order_id) join editorial_book using (book_id) join book_copy copy2 using (editorial_book_id) join book_sale sale using (book_copy_id) join editorial_book bought_book on copy2.editorial_book_id = bought_book.editorial_book_id join sale s using (sale_id) where c.created_date < DATE_ADD(NOW(), INTERVAL -14 DAY) and bought_book.book_id = editorial_book.book_id and s.client_id = o.client_id and s.created_date \geq c.created_date);
21	1. $\rho_{\text{book_per_month}}(\pi$ calendar.dd, book_storage.book_copy_id, book_storage.start_date, book_storage.end_date, $\rho_{\text{is_between}}(\text{calendar.dd between}$ book_storage.start_date and book_storage.end_date) (σ book_storage.storage_name = 'Warehouse' (calendar \times book_storage \bowtie storage))) 2. π MONTH(dd), YEAR(dd), sum(is_between) (σ book_per_month γ MONTH(dd), YEAR(dd) τ YEAR(DD), MONTH(DD))	select MONTH(dd), YEAR(dd), sum(is_between) from (SELECT c.dd, bs.book_copy_id, bs.start_date, IFNULL(bs.end_date, DATE(NOW())) as end_date, (CASE WHEN c.dd between start_date and IFNULL(bs.end_date, DATE(NOW())) THEN 1 ELSE 0 END) as is_between from calendar c, book_storage bs join storage s using (storage_id) where storage_name = 'Warehouse') books_per_month group by 1, 2 order by 2, 1;
22	π count(book_copy_id), sum(original_price) (σ created_date between '?' and '?' (purchase \bowtie book_copy))	select count(book_copy_id), sum(original_price) from purchase join book_copy copy2 using (book_copy_id) where created_date between '?' and '?';
23	$\rho_{\text{month_profit}}(\pi$ (π sum(original_price + profit) (σ MONTH(created_date) = ? (sale \bowtie book_sale \bowtie book_copy))) - (π sum(original_price) (σ MONTH(created_date) = ? (purchase \bowtie book_copy)))	select (select sum(original_price + profit) from sale join book_sale bs on sale.sale_id = bs.sale_id join book_copy copy2 on bs.book_copy_id = copy2.book_copy_id where MONTH(created_date) = ?) - (select sum(original_price)

		from purchase join book_copy copy2 using (book_copy_id) where MONTH(created_date) = ?) as month_profit;
24	π YEAR(dd), count(sale.sale_id) / 12 (σ calendar \bowtie sale) γ YEAR(dd) τ YEAR(dd)	select YEAR(dd), count(s.sale_id)/12 from calendar left join sale s on YEAR(s.created_date) = YEAR(dd) group by YEAR(dd) order by YEAR(dd) asc;
25	π $\rho_{\text{net_salary}}$ (sum(hours_count * hourly_salary)) (σ month_id = ? \wedge year_id = ? \wedge employee_id = ? (employee_history \bowtie employee_schedule \bowtie employee))	select IFNULL(sum(hours_count * hourly_salary), 0) as net_salary from employee_history join employee_schedule es using (employee_id) join employee e using(employee_id) where month_id = ? and year_id = ? and employee_id = ?;
26	π employee.first_name, employee.last_name (σ month(created_date) = ? \wedge year(created_date) = ? (sale \bowtie employee) γ employee_id τ count(sale_id))	select CONCAT(CONCAT(e.first_name, ' '), e.last_name) from sale join employee e using(employee_id) where month(created_date) = ? and year(created_date) = ? group by employee_id order by count(sale_id) desc limit 1