Rational algebra

1. π books (σ invertory =existing)

2. π purchases(σ Status NOT LIKE 'buy')

3. π customers

4. π suppliers

5. π purchases(σ (purchases.purchase_date≥y1 ^ purchses.purchase_date≤y2))

6. π(books $\bowtie$ discount) (σ discount.global_discount!=0)

7. π storage(σ (storage.is_exists_in_storage=yes ^ storage.s_number=x)

8. π suppliers(σ suppliers.first_name,suppliers.last_name ^ suppliers.s_number=x)

9. π purchases(σ purchases.s_number=x ^ purchases.purchases_date≥y)

10. π(customer $\bowtie$ order to customer)

(σ order_To_customer.how_much_items_per_customer,costomer.first_name,costumer.last_name ^customer.date_order≥y)

11. π purchases(σmax (purchases.customer_id ^ purchases.status=buy^ purchases.purchases_date≥y)

12. $\pi($ suppliers $\bowtie$ order from suppliers $)$ (σ max(order_from_supplier.how_much_items_has_ordered), suppliers.first_name,suppliers.last_name^ order from suppliers.date_suppliers_order≥y).

13. πorder_from_suppliers(σsum(orders_from_supplier.order_ supplier_number^(order_date=y1 ^order_date=y2))

14. πpurchases(σsum(purchases.s_number ^ purchases.status=buy)^ (order_date=y1 ^order_date=y2)).

15. π business revenue (σsum ($(amount)/3$)^ (business_revenue=q1,q2,q3)) &&($(amount)/1$)^ (business_revenue=q4))).

16. $\pi($ discount $\bowtie$ order to customer $)$ (σ sum(discount.local_discount)^(order_to_customer.date_order=y)).

17.customers((σcount(customer.customer_id^ (customer.join_date≥y)).

18. $\pi($purchases $\bowtie$ order from suppliers $)$ (σ(sum) order_from_supplier.price)^(purchases_date=y1 ^purchases_date=y2)).

19. $\pi_{(worker)}$ (σ(sum)worker.sales, worker.first_name, worker.last_name)^(selling_date=y1 ^selling_date=y2)).

20. . $\pi_{(purchases \bowtie books)}$ (σ(distinct) book.title)^ (purchases_date=y1 ^purchases_date=y2)^ (purchases.status=buy)).