PROBLEM STATEMENT

Using the given Salary, Income and Deduction tables:

1. **Populate the Emp_Transaction table as shown in next page**
2. **Generate a salary report as shown.**

Input Tables:

SALARY		
EMP_ID	EMP_NAME	BASE_SALARY
1	Rohan	5000
2	Alex	6000
3	Maryam	7000

INCOME		
ID	INCOME	PERCENTAGE
1	Basic	100
2	Allowance	4
3	Others	6

DEDUCTION		
ID	DEDUCTION	PERCENTAGE
1	Insurance	5
2	Health	6
3	House	4

EXPECTED OUTPUT 1

EXPECTED OUTPUT - EMP_TRANSACTION			
EMP_ID	EMP_NAME	TRNS_TYPE	AMOUNT
1	Rohan	Insurance	250
2	Alex	Insurance	300
3	Maryam	Insurance	350
1	Rohan	House	200
2	Alex	House	240
3	Maryam	House	280
1	Rohan	Basic	5000
2	Alex	Basic	6000
3	Maryam	Basic	7000
1	Rohan	Health	300
2	Alex	Health	360
3	Maryam	Health	420
1	Rohan	Allowance	200
2	Alex	Allowance	240
3	Maryam	Allowance	280
1	Rohan	Others	300
2	Alex	Others	360
3	Maryam	Others	420

EXPECTED OUTPUT 2

EXPECTED OUTPUT - SALARY REPORT									
EMPLOYEE	BASIC	ALLOWANCE	OTHERS	GROSS	INSURANCE	HEALTH	HOUSE	TOTAL_DEDUCTIONS	NET_PAY
Alex	6000	240	360	6600	300	360	240	900	5700
Maryam	7000	280	420	7700	350	420	280	1050	6650
Rohan	5000	200	300	5500	250	300	200	750	4750

SOLUTION 1

```sql
create table emp_transaction
(
    emp_id      int,
    emp_name    varchar(50),
    trns_type   varchar(20),
    amount      numeric
);
```

```sql
-- Populating Emp_transaction table

INSERT INTO emp_transaction (emp_id,emp_name,trns_type,amount)
(select s.emp_id,s.emp_name,d.deduction as trns_type,d.percentage*s.base_salary/100 as amount from salary as s
cross join deduction as d
union
select s.emp_id,s.emp_name,i.income as trns_type,i.percentage*s.base_salary/100 as amount from salary as s
cross join income as i
)
```

SOLUTION 2

```sql
-- Salary Report

With base_cte as(
select emp_name,
sum(case when trns_type = 'Allowance' then amount end) as Allowance,
sum(case when trns_type = 'Basic' then amount end) as Basic,
sum(case when trns_type = 'Others' then amount end) as Others,
sum(case when trns_type = 'Insurance' then amount end) as Insurance,
sum(case when trns_type = 'Health' then amount end) as Health,
sum(case when trns_type = 'House' then amount end) as House
from emp_transaction
group by emp_name
)
select *, (Allowance + Basic + Others) as gross,
(Insurance + Health + House) as Total_deductions,
(Allowance + Basic + Others)-(Insurance + Health + House) as net_pay
from base_cte
```

18 %

Results | Messages

	emp_name	Allowance	Basic	Others	Insurance	Health	House	gross	Total_deductions	net_pay
1	Alex	240	6000	360	300	360	240	6600	900	5700
2	Maryam	280	7000	420	350	420	280	7700	1050	6650
3	Rohan	200	5000	300	250	300	200	5500	750	4750

SOLUTION - USING PIVOT

```
select emp_name,
Basic,Allowance,Others,
(Allowance + Basic + Others) as gross,
Insurance,Health, House,
(Insurance + Health + House) as Total_deductions,
(Allowance + Basic + Others)-(Insurance + Health + House) as net_pay
from
    (
        select emp_name,trns_type, amount
        from emp_transaction
    )bq
pivot
    (
        sum(amount)
        for trns_type in([Allowance],[Basic],[Others],[Insurance],[Health],[House])
    )pq;
```

18 %

Results | Messages

	emp_name	Basic	Allowance	Others	gross	Insurance	Health	House	Total_deductions	net_pay
1	Alex	6000	240	360	6600	300	360	240	900	5700
2	Maryam	7000	280	420	7700	350	420	280	1050	6650
3	Rohan	5000	200	300	5500	250	300	200	750	4750

LOGIC BEHIND - PIVOT

- Pivot function transposes Rows to Columns. It requires 3 or more columns in it's Base Query.

- First Column "emp_name" is the unique identifier for each row.
- Second column "trns_type" should provide list of categories/ columns required.
- Third column "amount" is the value that should be returned for each column in "trns_type"

- In second query, We need to provide list of columns returned from "trns_type" and perform aggregation[SUM() here] on "amount" values.

Pivot Query Structure →

```
select *
from
    (
        base query
    )bq
pivot
    (
        agg func
        list of columns
    )pq;
```