---------- SALARY GENERATOR OF AN EMPLOYEE ----------

Relation Database Management System (RDMS): PostgreSQL.

Prepared by: Michael Henry Bujiku.

PROBLEM STATEMENT:

Using the given Salary, Income and Deduction tables.

- First write a SQL query to populate the Emp_Transaction table and then generate a salary report as shown.

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

We have Two table.

- Salary Table.
- Income Table.
- Deduction Table.

We need to create a new Table from these three tables will called: Emp_Transaction table.

SQL QUERIES PARTS

```
select * from salary;
select * from income;
select * from deduction;
```

To create a new table "Emp_transaction" from the existed table which is income, deduction and salary.

```
insert into emp_transaction
select emp_id, emp_name, trns_type
,      case when trns_type = 'Insurance' then round(base_salary* (percentage/100),2)
            when trns_type = 'House'     then round(base_salary* (percentage/100),2)
            when trns_type = 'Health'    then round(base_salary* (percentage/100),2)
            when trns_type = 'Basic'     then round(base_salary* (percentage/100),2)
            when trns_type = 'Allowance' then round(base_salary* (percentage/100),2)
            when trns_type = 'Others'    then round(base_salary* (percentage/100),2)
       end as Amount
from salary
cross join
          (select income as trns_type, cast(percentage as decimal) as percentage from income
           union
           select deduction as trns_type, cast(percentage as decimal) as percentage from deduction) x
```

EMP_TRANSACTION TABLE			
emp_id	**emp_name**	**trns_type**	**amount**
1	Rohan	Others	300
2	Alex	Others	360
3	Maryam	Others	420
1	Rohan	Allowance	200
2	Alex	Allowance	240
3	Maryam	Allowance	280
1	Rohan	Insurance	250
2	Alex	Insurance	300
3	Maryam	Insurance	350
1	Rohan	Health	300
2	Alex	Health	360
3	Maryam	Health	420
1	Rohan	House	200
2	Alex	House	240
3	Maryam	House	280
1	Rohan	Basic	5000
2	Alex	Basic	6000
3	Maryam	Basic	7000

The task is to convert those information's from Row's Level Data into Column's Level Data.

Because we are using PostgreSQL we going to use "Cross Tab" instead of "Pivot table".

Syntax of "Cross Tab"

```
select *
from crosstab('base query'
              , 'list of columns')
        as result(final columns with data type)
```

Note: Base query must have at least 3 columns.
First column: Needs to be the unique identifier which it will show at last as (*First column*).
emp_name -- Column
Second column: Needs to be the column that is going to return new columns that your cross tab needs to create.
trns_type -- Column
Last column: Once you created all of these new columns the values that need to get inserted into these columns should come from this column.
amount – Column
Note: You need to use "Order by" in order to do aggregation of the columns to remove nulls values and be aggregated in open values column/rows

Base query:

```
select emp_name, trns_type, amount
from emp_transaction;
order by emp_name, trns_type
```

List of Columns:

```
select distinct trns_type from emp_transaction order by trns_type;
```

Final columns with data type;

In this part you need to arrange the column in the same order as it is in this "List of columns". This will make each column to have their real data value. Failure to do that the value will be different from the actual table which is "Emp_transactions" Table. Like the values of allowance will appear in House column and vice versa.

```
as result (employee varchar, allowance numeric, basic numeric, health numeric
        , house numeric, insurance numeric, others numeric)
```

FINAL QUERY:

```
select employee
, basic, allowance, others
, (basic + allowance + others) as gross
, insurance, health, house
, (insurance + health + house) as total_deduction
, ((basic + allowance + others) - (insurance + health + house)) as net_pay
from crosstab('select emp_name, trns_type, amount
                from emp_transaction
                order by emp_name, trns_type'
              , 'select distinct trns_type from emp_transaction order by trns_type')
    as result(employee varchar, allowance numeric, basic numeric, health numeric
                , house numeric, insurance numeric, others numeric);
```

THE OUTPUT:

Employee	Basic	Allowance	Others	Gross	Insurance	Health	House	Total_deduction	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