1. CREATE TABLE

```
create table customers (
        customer_id int primary key,
        first_name varchar,
        last_name varchar,
        customer_email varchar,
        customer_phone varchar,
        customer_address varchar,
        customer_city varchar,
        customer_state varchar,
        customer_zip int
);

create table product_category (
        category_id int primary key,
        category_name varchar,
        category_abbreviation varchar
);

create table products (
        prod_number varchar primary key,
        prod_name varchar,
        category_id int,
        price float8,
        foreign key (category_id) references product_category(category_id) on delete set null
);

create table orders (
        order_id int primary key,
        date timestamp,
        customer_id int,
        prod_number varchar,
        quantity int,
        foreign key (customer_id) references customers(customer_id) on delete set null,
        foreign key (prod_number) references products(prod_number) on delete set null
);
```

2. GROWTH ANALYSIS

```
with
monthly_active_users as (
        select
                year,
                round(avg(total),2) as average_mau
        from (
                select
                        extract(year from date) as year,
                        extract(month from date) as month,
                        count(customer_id) as total
```

```
                from
                        orders
                group by 1, 2
                order by 1, 2
                ) as subq
                group by 1
),

new_customer as (
        select
                extract(year from first_purchase) as year,
                count(customer_id) as total_new_customer
        from (
                select
                        c.customer_id,
                        min(o.date) as first_purchase
                from
                        customers as c
                        join orders as o
                        on c.customer_id = o.customer_id
                group by 1
                ) as subq2
                group by 1
),

repeat_order as (
        select
                year,
                sum(total) as total_customer_repeat_order
        from (
                select
                        extract(year from date) as year,
                        customer_id,
                        count(order_id) as total
                from
                        orders
                group by 1, 2
                having count(order_id) > 1
                order by 1, 3 desc
                ) as subq3
                group by 1
),

revenue as (
        select
                extract(year from o.date) as year,
                round(cast(sum(o.quantity * p.price) as numeric),2) as total_revenue
        from
                orders as o
                join products as p
                on o.prod_number = p.prod_number
```

```
        group by 1
),

quantity as (
        select
                extract(year from date) as year,
                sum(quantity) as total_quantity
        from
                orders
        group by 1
)

select
        mau.year,
        mau.average_mau,
        nc.total_new_customer,
        ro.total_customer_repeat_order,
        r.total_revenue,
        q.total_quantity
from
        monthly_active_users as mau
        join new_customer as nc
        on mau.year = nc.year
        join repeat_order as ro
        on mau.year = ro.year
        join revenue as r
        on mau.year = r.year
        join quantity as q
        on mau.year = q.year
```

3. PRODUCT ANALYSIS

```
with
highest_value_product as (
        select
                year,
                rank_product,
                highest_product,
                highest_revenue_product
        from (
                select
                        extract(year from o.date) as year,
                        p.prod_name as highest_product,
                        sum(o.quantity * p.price) as highest_revenue_product,
                        rank() over(partition by extract(year from date) order by sum(o.quantity *
p.price) desc) as rank_product
                from
                        orders as o
                        join products as p
                        on o.prod_number = p.prod_number
                group by 1, 2
```

```
        ) as subq1
        where rank_product in (1,2,3)
),

lowest_value_product as (
        select
                year,
                rank_product,
                lowest_product,
                round(cast((lowest_revenue_product) as numeric),2) as lowest_revenue_product
        from (
                select
                        extract(year from o.date) as year,
                        p.prod_name as lowest_product,
                        sum(o.quantity * p.price) as lowest_revenue_product,
                        rank() over(partition by extract(year from date) order by sum(o.quantity *
p.price) asc) as rank_product
                from
                        orders as o
                        join products as p
                        on o.prod_number = p.prod_number
                group by 1, 2
                ) as subq2
        where rank_product in (1,2,3)
),

highest_value_category as (
        select
                year,
                rank_product,
                highest_category,
                highest_revenue_category
        from (
                select
                        extract(year from o.date) as year,
                        pc.category_name as highest_category,
                        sum(o.quantity * p.price) as highest_revenue_category,
                        rank() over(partition by extract(year from date) order by sum(o.quantity *
p.price) desc) as rank_product
                from
                        orders as o
                        join products as p
                        on o.prod_number = p.prod_number
                        join product_category as pc
                        on p.category_id = pc.category_id
                group by 1, 2
                ) as subq3
                where rank_product in (1,2,3)
),

lowest_value_category as (
```

```
	select
		year,
		rank_product,
		lowest_category,
		round(cast((lowest_revenue_category) as numeric),2) as lowest_revenue_category
	from (
		select
			extract(year from o.date) as year,
			pc.category_name as lowest_category,
			sum(o.quantity * p.price) as lowest_revenue_category,
			rank() over(partition by extract(year from date) order by sum(o.quantity *
p.price) asc) as rank_product
		from
			orders as o
			join products as p
			on o.prod_number = p.prod_number
			join product_category as pc
			on p.category_id = pc.category_id
		group by 1, 2
		) as subq4
		where rank_product in (1,2,3)
),

highest_value_region as (
	select
		year,
		rank_product,
		highest_region,
		round(cast((highest_revenue_region) as numeric),2) as highest_revenue_region
	from (
		select
			extract(year from o.date) as year,
			c.customer_state as highest_region,
			sum(o.quantity * p.price) as highest_revenue_region,
			rank() over(partition by extract(year from date) order by sum(o.quantity *
p.price) desc) as rank_product
		from
			orders as o
			join customers as c
			on o.customer_id = c.customer_id
			join products as p
			on o.prod_number = p.prod_number
		group by 1, 2
		) as subq5
		where rank_product in (1,2,3)
),

lowest_value_region as (
	select
		year,
		rank_product,
```

```
		lowest_region,
		round(cast((lowest_revenue_region) as numeric),2) as lowest_revenue_region
	from (
		select
			extract(year from o.date) as year,
			c.customer_state as lowest_region,
			sum(o.quantity * p.price) as lowest_revenue_region,
			rank() over(partition by extract(year from date) order by sum(o.quantity *
p.price) asc) as rank_product
		from
			orders as o
			join customers as c
			on o.customer_id = c.customer_id
			join products as p
			on o.prod_number = p.prod_number
		group by 1, 2
		) as subq1
		where rank_product in (1,2,3)
)

select
	hvp.year,
	hvp.highest_product,
	hvp.highest_revenue_product,
	lvp.lowest_product,
	lvp.lowest_revenue_product,
	hvc.highest_category,
	hvc.highest_revenue_category,
	lvc.lowest_category,
	lvc.lowest_revenue_category,
	hvr.highest_region,
	hvr.highest_revenue_region,
	lvr.lowest_region,
	lvr.lowest_revenue_region
from
	highest_value_product as hvp
	join lowest_value_product as lvp
	on hvp.year = lvp.year and hvp.rank_product = lvp.rank_product
	join highest_value_category as hvc
	on hvp.year = hvc.year and hvp.rank_product = hvc.rank_product
	join lowest_value_category as lvc
	on hvp.year = lvc.year and hvp.rank_product = lvc.rank_product
	join highest_value_region as hvr
	on hvp.year = hvr.year and hvp.rank_product = hvr.rank_product
	join lowest_value_region as lvr
	on hvp.year = lvr.year and hvp.rank_product = lvr.rank_product
```