Business Problem

Retail management wants analytical insights from sales data:
- Identify top customers and products
- Track revenue trends over time
- Detect inactive customers
- Improve query performance on large datasets

Data Model (Star Schema)

FACT_SALES
- sale_id
- customer_id
- product_id
- sale_date
- amount

DIM_CUSTOMER
- customer_id
- customer_name
- join_date

DIM_PRODUCT
- product_id
- product_name
- category

```
-- DIMENSION TABLES
CREATE TABLE dim_customer (
    customer_id NUMBER PRIMARY KEY,
    customer_name VARCHAR2(100),
    join_date DATE
);

CREATE TABLE dim_product (
    product_id NUMBER PRIMARY KEY,
    product_name VARCHAR2(100),
    category VARCHAR2(50)
);

-- FACT TABLE
CREATE TABLE fact_sales (
    sale_id NUMBER PRIMARY KEY,
    customer_id NUMBER,
    product_id NUMBER,
    sale_date DATE,
```

```
    amount NUMBER(10,2),
    CONSTRAINT fk_cust FOREIGN KEY (customer_id) REFERENCES
dim_customer(customer_id),
    CONSTRAINT fk_prod FOREIGN KEY (product_id) REFERENCES
dim_product(product_id)
);

-- INDEXES
CREATE INDEX idx_sales_date ON fact_sales(sale_date);
CREATE INDEX idx_sales_customer ON fact_sales(customer_id);


INSERT INTO dim_customer VALUES (1, 'Rahul', DATE '2022-01-10');
INSERT INTO dim_customer VALUES (2, 'Neha', DATE '2022-02-15');

INSERT INTO dim_product VALUES (101, 'Laptop', 'Electronics');
INSERT INTO dim_product VALUES (102, 'Phone', 'Electronics');

INSERT INTO fact_sales VALUES (1,1,101,DATE '2024-01-10',50000);
INSERT INTO fact_sales VALUES (2,1,102,DATE '2024-02-10',30000);
INSERT INTO fact_sales VALUES (3,2,101,DATE '2024-02-12',50000);

COMMIT;
```

sql/analytics_queries.sql

(CORE OF PROJECT)

🔥 1. Top Customer by Revenue

```
SELECT customer_id,
       SUM(amount) total_revenue,
       RANK() OVER (ORDER BY SUM(amount) DESC) rnk
FROM fact_sales
GROUP BY customer_id;
```

🔥 2. Running Total of Sales (Window Function)

```
SELECT sale_date,
       amount,
       SUM(amount) OVER (ORDER BY sale_date) running_total
FROM fact_sales;
```

🔥 3. Month-over-Month Growth

```
SELECT month,
       total_sales,
       total_sales - LAG(total_sales) OVER (ORDER BY month) mom_growth
FROM (
   SELECT TO_CHAR(sale_date,'YYYY-MM') month,
          SUM(amount) total_sales
   FROM fact_sales
   GROUP BY TO_CHAR(sale_date,'YYYY-MM')
);
```

🔥 4. First & Last Purchase (Oracle Analytic)

```
SELECT customer_id,
       MIN(sale_date) KEEP (DENSE_RANK FIRST ORDER BY sale_date) first_purchase,
       MAX(sale_date) KEEP (DENSE_RANK LAST ORDER BY sale_date) last_purchase
FROM fact_sales
GROUP BY customer_id;
```

🔥 5. Inactive Customers (6 Months)

```
SELECT c.customer_id, c.customer_name
FROM dim_customer c
LEFT JOIN fact_sales f ON c.customer_id = f.customer_id
GROUP BY c.customer_id, c.customer_name
HAVING MAX(f.sale_date) < ADD_MONTHS(SYSDATE,-6)
   OR MAX(f.sale_date) IS NULL;
```