```
Create database EDA;
use EDA;

select * from customer_shopping_data;



-- Data Analytics / EDA (Exploratory Data Analysis)  with SQL
            -- Data Exploration
                        -- connecting to database
        -- explore schemas
    -- Data Cleaning - Duplicate, NULL, Inconsistent data
                        -- count, case, null, is null
            -- Data Aggregation - statistics (avg, sum, min, max, count)
                        -- data distribution and outliers, group by
    -- Data Transformation
                        -- join clause, order by, where
    -- Data Visualization
    -- Hypothesis Testing - correlations
    -- Reporting & Communication



-- EDA Process - Data Exploration

select count(*) from customer_shopping_data;

desc customer_shopping_data;

alter table customer_shopping_data add id int auto_increment primary key;

select * from customer_shopping_data limit 10;
```

```
select count(*) as column_count
        from information_schema.columns
   where table_schema = 'EDA'
   and table_name = 'customer_shopping_data';

select count(*) as row_count, (select count(*)
        from information_schema.columns
   where table_schema = 'EDA'
   and table_name = 'customer_shopping_data') as column_count from customer_shopping_data;

-- Data Cleaning

select count(distinct invoice_no) from customer_shopping_data;

select count(distinct customer_id) from customer_shopping_data;

select gender
from customer_shopping_data
group by gender;

select * from customer_shopping_data where gender='m';

update customer_shopping_data set gender='Male' where gender='m';

select count(distinct gender) from customer_shopping_data;

select count(age) from customer_shopping_data;

select category
from customer_shopping_data
```

```
group by category;

select count(distinct category) from customer_shopping_data;

select count(quantity) from customer_shopping_data;

select count(price) from customer_shopping_data;

select * from customer_shopping_data where category='Clothing' and quantity=5;

update customer_shopping_data set price=1500.4 where price is NULL and category='Clothing' and
quantity=5;

select * from customer_shopping_data where category='Souvenir' and quantity=2;

update customer_shopping_data set price=23.46 where price is NULL and category='Souvenir' and
quantity=2;

select * from customer_shopping_data where category='Shoes' and quantity=4;

update customer_shopping_data set price=2400.68 where price is NULL and category='Shoes' and
quantity=4;

select * from customer_shopping_data where price is null;

select payment_method
from customer_shopping_data
group by payment_method;
```

```
select * from customer_shopping_data where payment_method is null;

select count(distinct payment_method) from customer_shopping_data;

select count(invoice_date) from customer_shopping_data;

select shopping_mall
from customer_shopping_data
group by shopping_mall;

select count(distinct shopping_mall) from customer_shopping_data;

-- update the table

update customer_shopping_data set price=NULL where id=1;

update customer_shopping_data set price=NULL where id in (2,3,4);

update customer_shopping_data set age=NULL where id in (5,8);

update customer_shopping_data set payment_method=NULL where id=4;

update customer_shopping_data set age=400 where id=1;

update customer_shopping_data set gender='m' where id=4;


select * from customer_shopping_data where shopping_mall = 'Metrocity';

select min(payment_method), max(payment_method) from customer_shopping_data;
```

```
update customer_shopping_data set payment_method='Cash' where payment_method is NULL;

-- delete from customer_shopping_data where id = 4;

-- Data Aggregation

select min(age), max(age) from customer_shopping_data;

select * from customer_shopping_data where age=400;

select avg(age) from customer_shopping_data;

update customer_shopping_data set age=43 where age=400;

select * from customer_shopping_data;

update customer_shopping_data set age=43 where age is null;

select min(quantity), max(quantity) from customer_shopping_data;

select min(price), max(price) from customer_shopping_data;
```