

Amazon Orders SQL Practice - With Notes

Creating the `amazon_orders` table

```
create table amazon_orders (  
    order_id INTEGER,  
    order_date DATE,  
    product_name VARCHAR(20),  
    total_price DECIMAL(6,2),  
    payment_method VARCHAR(10)  
);
```

Note: Defines a table to store Amazon order details.

Inserting sample data

```
insert into amazon_orders values(1,'2025-11-01','bag',190,'upi');  
insert into amazon_orders values(2,'2025-10-02','milk',20,'upi');  
insert into amazon_orders values(3,'2025-10-03','mobile',2000,'bank');  
insert into amazon_orders values(4,'2025-11-01','pen',10,'upi');  
insert into amazon_orders values(5,'2025-11-01','pen',100,'upi');
```

Note: Adds rows with example products.

Selecting all and specific columns

```
select * from amazon_orders;  
select order_id, product_name from amazon_orders;
```

Note: Fetches all rows or only selected columns.

Limiting rows

```
select top 2 * from amazon_orders;
```

Note: Shows only the first 2 rows.

Deleting and dropping

```
delete from amazon_orders;  
drop table amazon_orders;
```

Note: DELETE removes data, DROP removes the entire table.

Ordering results

```
select * from amazon_orders order by product_name, total_price desc;
```

Note: Sorts by product name, then by total_price in descending order.

Creating a backup table

```
create table amazon_orders_backup (  
    order_id INTEGER,  
    order_date DATE,  
    product_name VARCHAR(20),  
    total_price DECIMAL(6,2),  
    payment_method VARCHAR(10)  
);
```

Note: Creates a backup copy of the orders table.

Altering table columns

```
alter table amazon_orders alter column order_date DATETIME;  
alter table amazon_orders alter column total_price INTEGER;
```

Note: Changes data types of columns.

Primary keys and constraints

```
create table amazon_orders (  
    order_id INTEGER NOT NULL UNIQUE,  
    order_date DATE,  
    product_name VARCHAR(20) default 'shirt',  
    total_price DECIMAL(6,2),  
    payment_method VARCHAR(10),  
    discount INTEGER check (discount <=20),  
    PRIMARY KEY (payment_method)  
);
```

Note: Adds constraints: NOT NULL, UNIQUE, DEFAULT, CHECK, and PRIMARY KEY.

Filtering with WHERE

```
select * from amazon_orders where product_name = 'pen' and total_price < 15;
```

Note: Filters rows where product is 'pen' and price is less than 15.

Deleting specific rows

```
delete from amazon_orders where product_name = 'pen';
```

Note: Removes only rows matching the condition.

Restore database from backup

```
RESTORE FILELISTONLY
FROM DISK = '/var/opt/mssql/backups/AdventureWorksLT2017.bak';

RESTORE DATABASE AdventureWorksLT2017
FROM DISK = '/var/opt/mssql/backups/AdventureWorksLT2017.bak'
WITH MOVE 'AdventureWorksLT2012_Data' TO '/var/opt/mssql/data/AdventureWorksLT2017.mdf',
        MOVE 'AdventureWorksLT2012_Log' TO '/var/opt/mssql/data/AdventureWorksLT2017.ldf',
        REPLACE;

SELECT name FROM sys.databases;
```

Note: Shows how to restore a database backup into SQL Server.