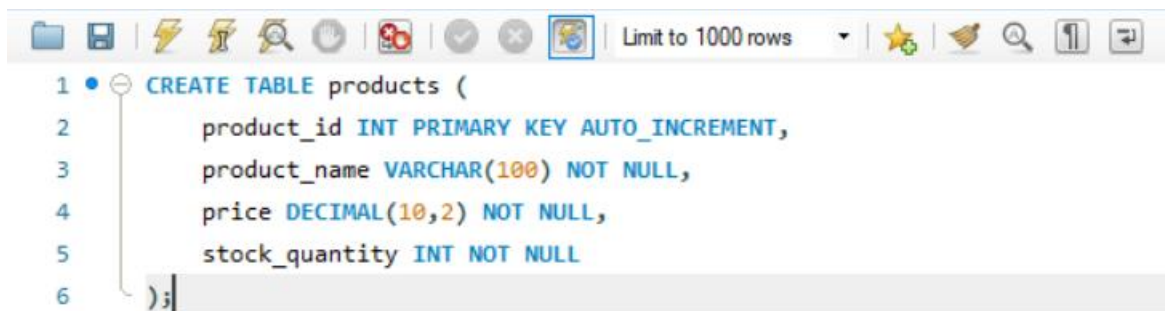
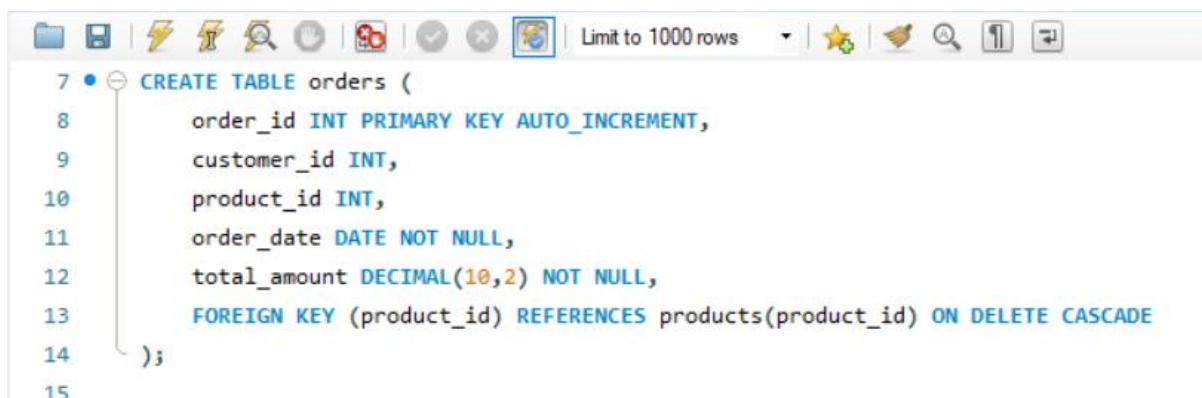


Assignment 4: Compose SQL statements to BEGIN a transaction, INSERT a new record into the 'orders' table, COMMIT the transaction, then UPDATE the 'products' table, and ROLLBACK the transaction.

Step 1 : Create the products and orders tables.



```
1 • CREATE TABLE products (  
2     product_id INT PRIMARY KEY AUTO_INCREMENT,  
3     product_name VARCHAR(100) NOT NULL,  
4     price DECIMAL(10,2) NOT NULL,  
5     stock_quantity INT NOT NULL  
6 );
```



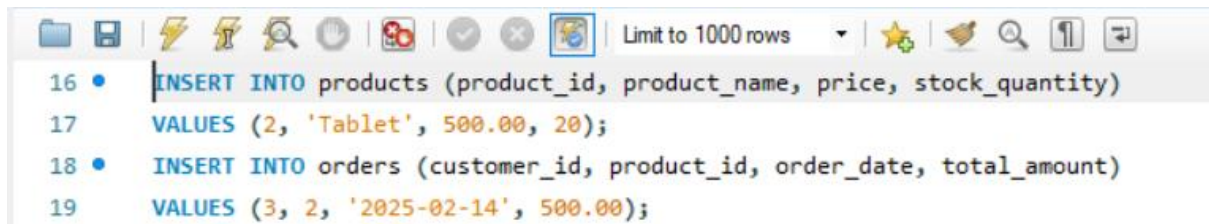
```
7 • CREATE TABLE orders (  
8     order_id INT PRIMARY KEY AUTO_INCREMENT,  
9     customer_id INT,  
10    product_id INT,  
11    order_date DATE NOT NULL,  
12    total_amount DECIMAL(10,2) NOT NULL,  
13    FOREIGN KEY (product_id) REFERENCES products(product_id) ON DELETE CASCADE  
14 );  
15
```

Step 2 : Start a Transaction.



```
15 • START TRANSACTION;
```

Step 3 : Insert a New Product & Insert a New Order.



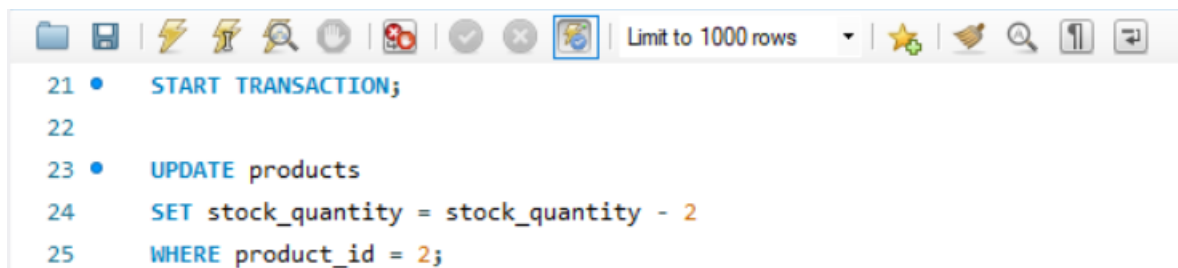
```
16 • INSERT INTO products (product_id, product_name, price, stock_quantity)
17   VALUES (2, 'Tablet', 500.00, 20);
18 • INSERT INTO orders (customer_id, product_id, order_date, total_amount)
19   VALUES (3, 2, '2025-02-14', 500.00);
```

Step 4 : Commit the Transaction.



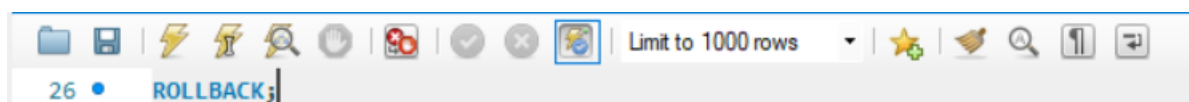
```
20 • COMMIT;
```

Step 5 : Start Another Transaction and Update the products Table.



```
21 • START TRANSACTION;
22
23 • UPDATE products
24   SET stock_quantity = stock_quantity - 2
25   WHERE product_id = 2;
```

Step 6 : Rollback the Stock Update.



```
26 • ROLLBACK;
```

Step 7 : Verify the Results.

