Welcome to Updating and Deleting Tables.

What you will learn

At the core of the lesson

You will learn how to:

- Define the characteristics of a transaction
- Use the COMMIT and ROLLBACK commands when you update a table
- Use the DROP TABLE statement to delete a table
- Identify common database design problems and good design practices

Key terms:

- Transaction
- START TRANSACTION
- COMMIT
- ROLLBACK
- DROP TABLE
- Database anomalies

2

aws re/start

You will learn how to:
- Define the characteristics of a transaction
- Use the COMMIT and ROLLBACK commands when you update a table
- Use the DROP TABLE statement to delete a table
- Identify common database design problems and good design practices

A **database transaction** is the reproduction of one or more changes that are performed on a database.

Transactions

Transactions:

- Run a set of operations so that the database never contains the result of partial operations
- Denote any alteration in a database
- Will never be complete unless each operation is successful

Transactions enable you to run a set of operations so that:

- If one operation fails, the database is restored to its original state
- If no errors occur, the full set of statements is bound to the database

4

aws restart

Transactions enable you to:

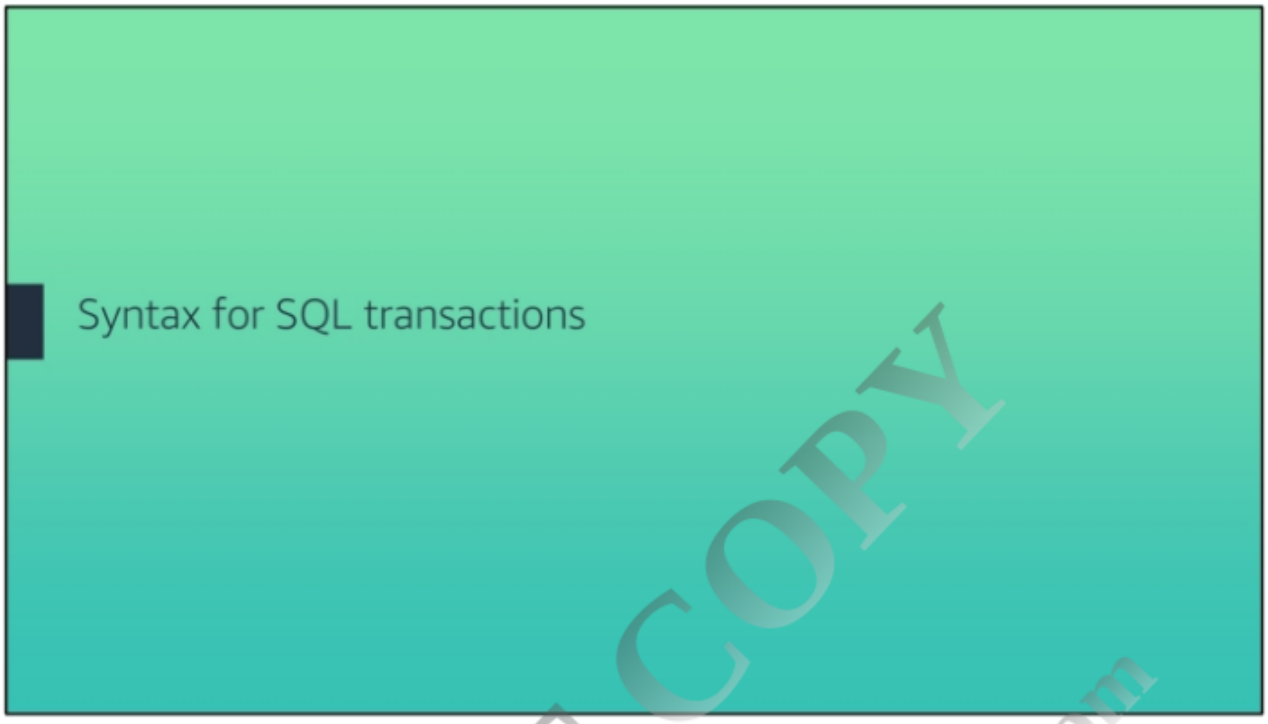
- Run a set of operations so that the database never contains the result of partial operations. If one operation fails, the database is restored to its original state. If no errors occur, the full set of statements is bound to the database.
- Provide isolation between programs that access a database simultaneously. If this isolation does not happen, the outcomes could prove to be incorrect.

Properties of transactions

Transactions follow four standard properties, which are known as **ACID.**

- **A**tomicity: Ensures that all changes are successfully completed.
- **C**onsistency: Ensures that any changes will not violate the integrity of the database, including any constraints.
- **I**solation: All transactions happen in isolation. Transactions are isolated so that they do not interfere with the other transactions.
- **D**urability: As soon as a transaction is committed, any interruption to the database's availability, such as a restart or system failure, does not affect the consistency of the data.

aws re/start



START TRANSACTION: BEGIN

Transaction keywords

Start a new transaction by using the following keywords:

`START TRANSACTION`

```
USE emp_table;
START TRANSACTION;
UPDATE emp_ID
SET lgID = '--';
```


aws re/start

COMMIT and ROLLBACK

ROLLBACK;

To roll back a transaction, use **ROLLBACK**

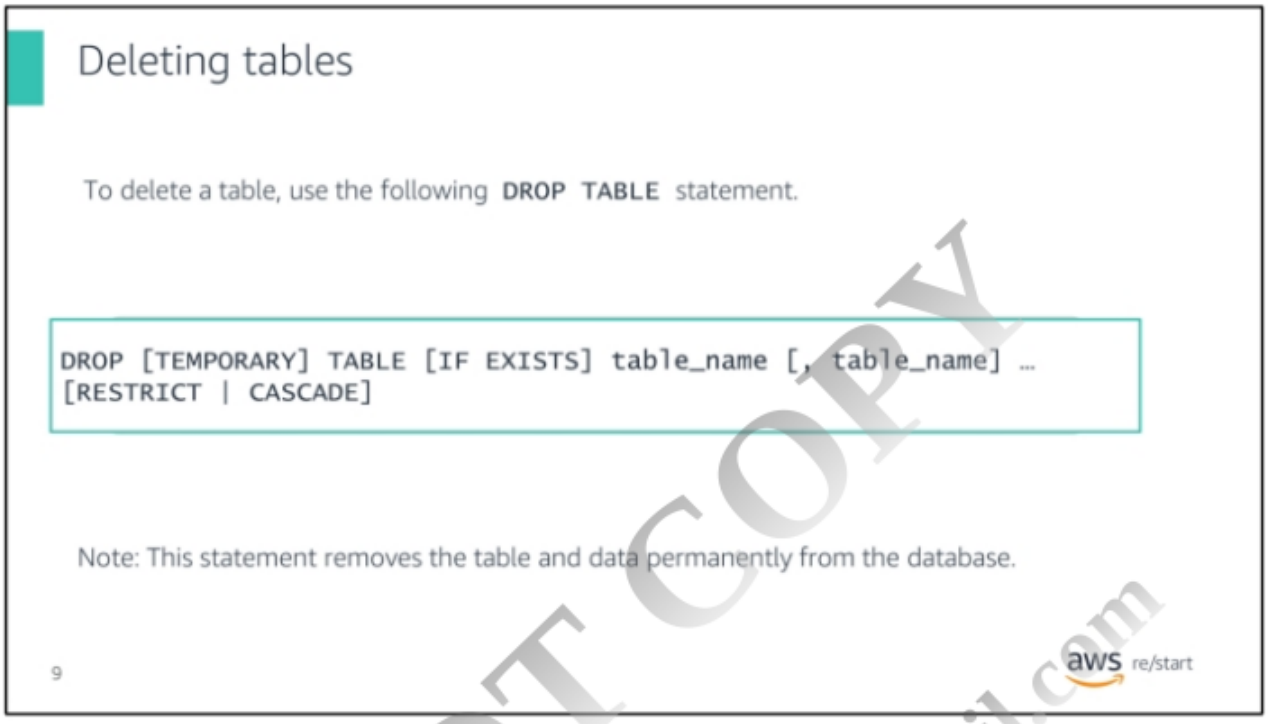
If mistakes happen, or your results are not correct, you can undo any work that is performed since **START TRANSACTION**.

COMMIT;

To commit the transaction, use **COMMIT**

If you are satisfied with the outcome of the transaction or the transaction is successful, use this command to save all changes.

aws re/start



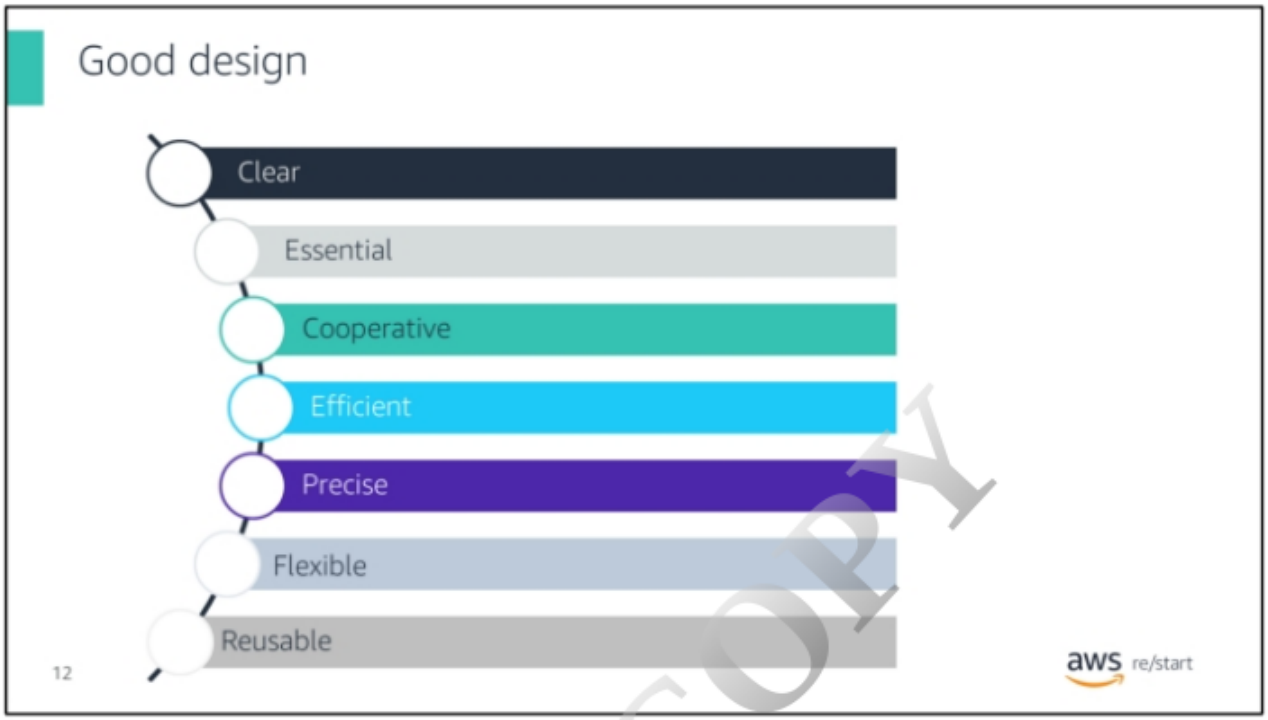
To delete multiple tables, separate each table by a comma in your **DROP TABLE** statement.

Note: When you perform a **DELETE** and **UPDATE**, be careful with delete and update clauses—like COUNTS, SELECT, and others—and verify that the output is as expected.

Identifying database anomalies

Anomalies might occur when database content changes.

- **Insertion anomaly –** Occurs when you cannot insert important data into the database because other data is missing. For example, a clerk cannot add customer information for a new customer who has not made a purchase.
- **Update anomaly –** Occurs when data is stored redundantly within the same table. This anomaly makes it difficult to ensure that all copies of a particular information value are updated when the value must be changed.
- **Deletion anomaly –** Occurs when you delete unwanted information, but the action also results in deleting information that you want to keep.



Good design can prevent anomalies. The first step is to identify where you might have a problem.

Activity: Identifying Data Anomalies

Overview

The following sample has several rows from the Orders table of a database that tracks sales orders. This table contains various anomalies that affect the user's ability to manage the data that it contains.

Instructions

Step one:

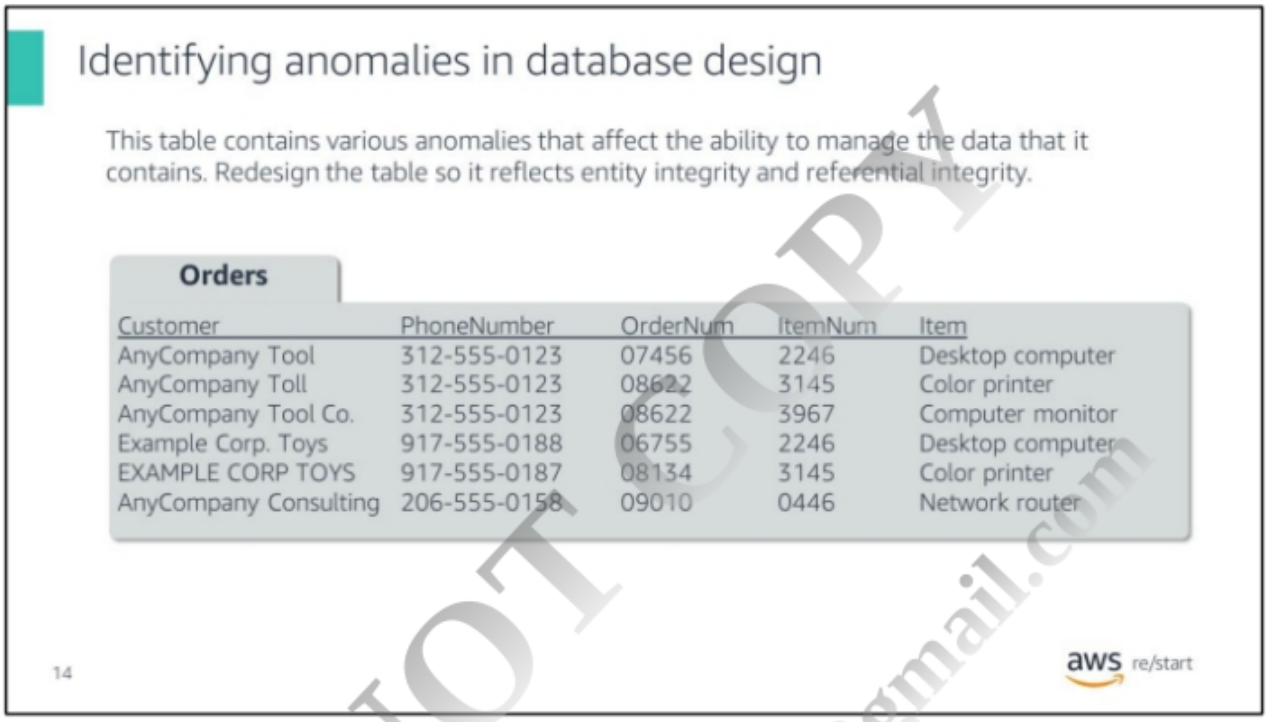
Identify any anomalies in the table.

Step two:

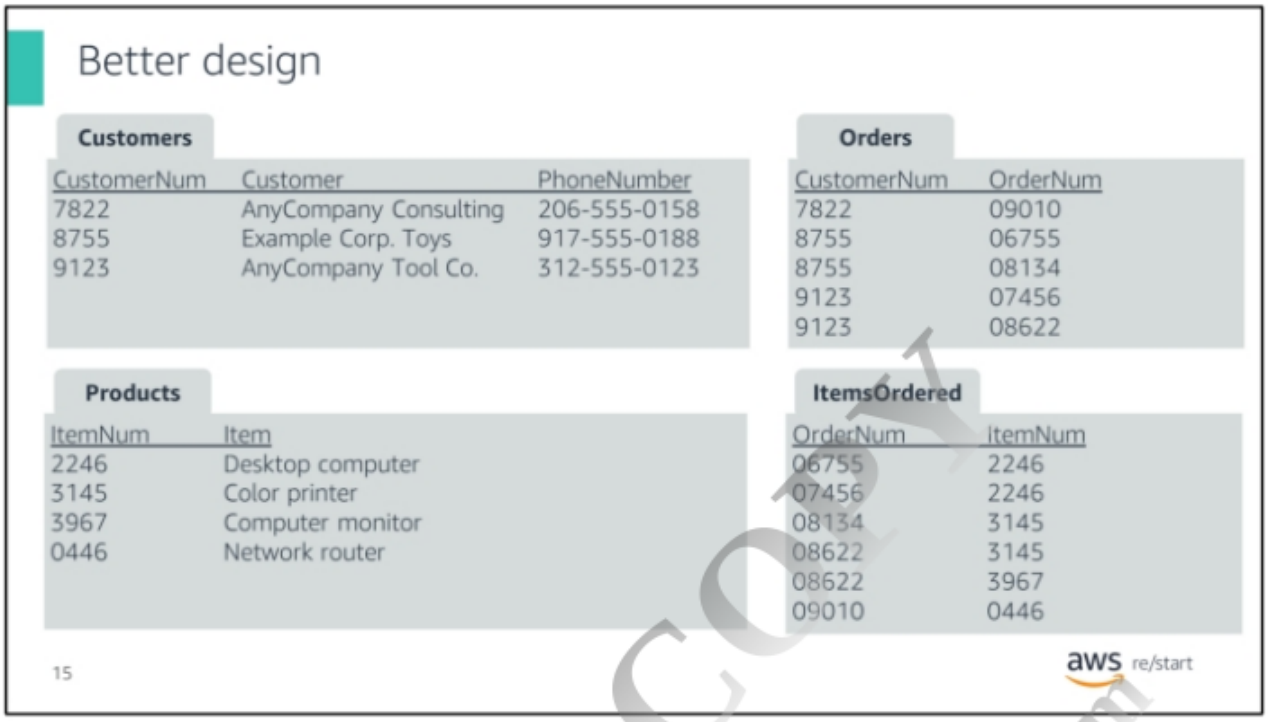
Redesign the following table so it reflects **entity integrity** and **referential integrity**.

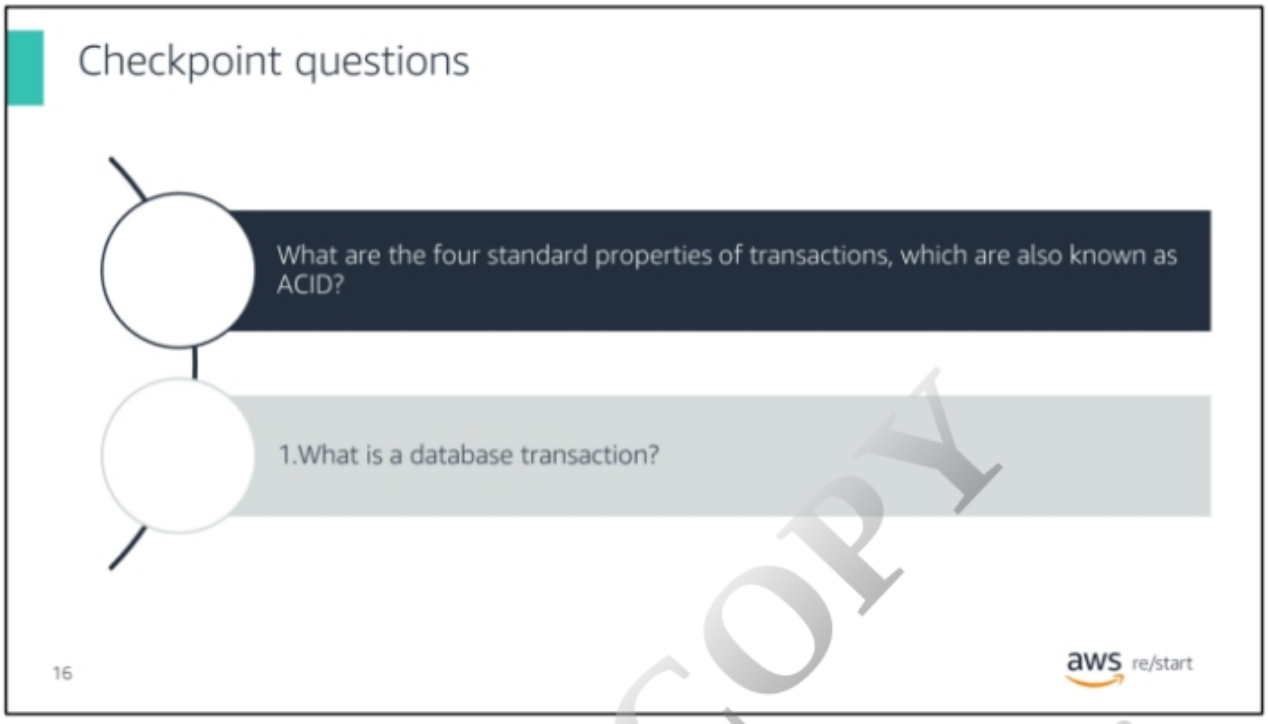
Hint: Consider separating the table.

aws re/start



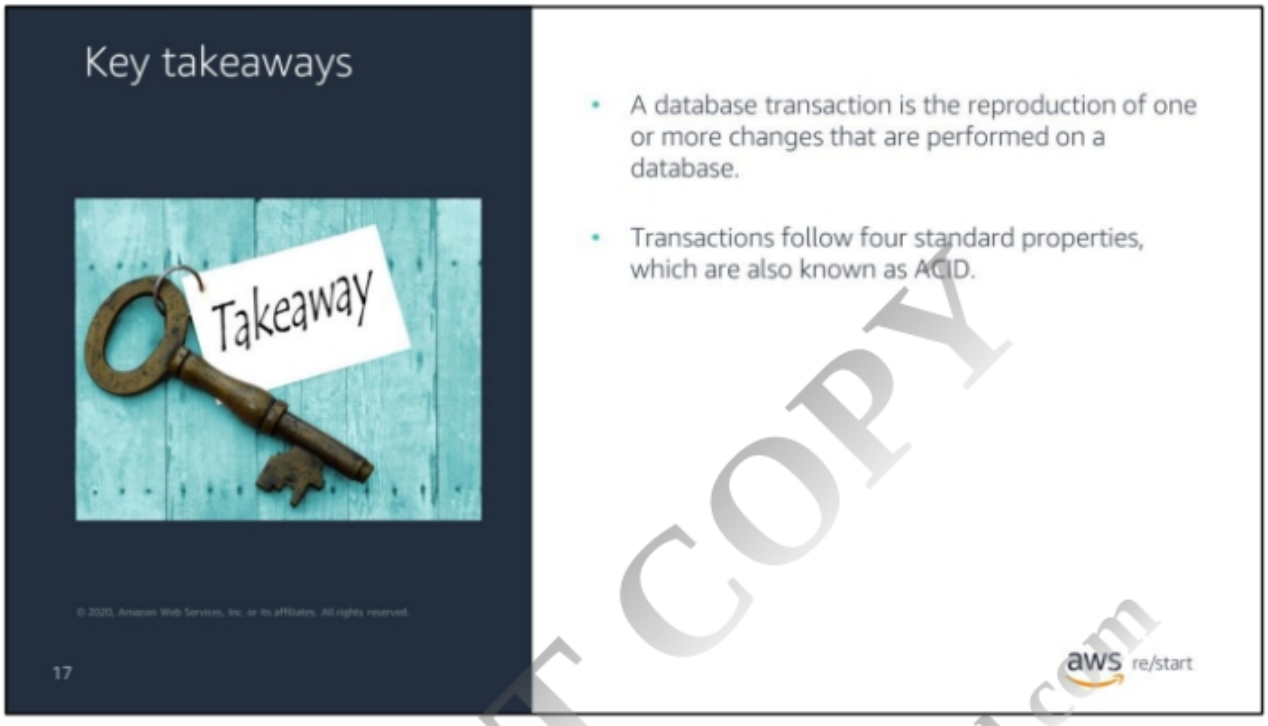
This table contains various anomalies that affect the user's ability to manage the data that the table contains.





Answers:

1. The four standard properties of transactions are **A**tomicity, **C**onsistency, **I**solation, and **D**urability.
2. A database transaction is the reproduction of one or more changes that are performed on a database.



Some key takeaways from this lesson include:

- A database transaction is the reproduction of one or more changes that are performed on a database.
- Transactions follow four standard properties, which are also known as ACID.