

ALTER TABLE to Change Columns: Data Characteristics

by Sophia Tutorial



WHAT'S COVERED

This tutorial explores using the ALTER TABLE command to change the data type and size of a column:

1. Changing Columns: Data Characteristics
2. Handling New Cases

1. Changing Columns: Data Characteristics

Beyond simply changing the data type, we may need to change other aspects of a table, like the size of a column. This can be useful to optimize the size of a table, or to increase the size of a column to fit added data. The structure of the command is the same as changing the data type, which we covered in the prior tutorial:

```
ALTER TABLE <tablename>  
ALTER COLUMN <columnname>  
TYPE <newdatatype>;
```

We can keep the same data type but just change the size of the data, or use a smaller/larger version of the data type if one exists.

For example, let's take a look at a basic registration table used to capture individuals registering for an event:

```
CREATE TABLE registration(  
  registration_id int PRIMARY KEY,  
  first_name VARCHAR(10),  
  last_name VARCHAR(10),  
  email VARCHAR(30),  
  fee NUMERIC(4,2)  
);
```

Let's add some sample data. We will cover this in a later tutorial:

```
INSERT INTO registration VALUES (1,'Michelle','Pippen','mpippen@a.com',9.99);
```

```
INSERT INTO registration VALUES (2, 'Santana','Smith','smith@b.com',9.99);
```

No error should occur and if we query the table, we should see the two rows:

Query Results				
Row count: 2				
registration_id	first_name	last_name	email	fee
1	Michelle	Pippen	mpippen@a.com	9.99
2	Santana	Smith	smith@b.com	9.99

2. Handling New Cases

However, a new individual is trying to register with a last_name that has more than 10 characters. If we tried to insert a record with a value longer than 10 for the last name like this:

```
INSERT INTO registration VALUES (3, 'Joseph','Rudy-Potter','jpotter@c.com',9.99);
```

We should get an error like this:

Query Results
Query failed because of: error: value too long for type character varying(10)

We could simply count the number of characters and set the length of the variable to that value, but it would be better to anticipate future potential sizes to avoid having to make this adjustment again later. It is better to err on the higher side rather than the lower side. We can go ahead and change the length of the last name to 50 characters:

```
ALTER TABLE registration  
ALTER COLUMN last_name  
TYPE VARCHAR(50);
```

Now, if we run the insert statement again, it should run successfully:

Query Results
Query ran successfully. 0 rows to display.

We can make changes to more than one column at a time. For example, the first name and email should probably also have a larger size. We can combine them together by listing each ALTER COLUMN statement separated by a comma:

```
ALTER TABLE registration  
ALTER COLUMN first_name  
TYPE VARCHAR(50),
```

```
ALTER COLUMN email
```

```
TYPE VARCHAR(100);
```

In looking at the fee, we are limited to four digits, with two digits after the decimal point. This means the maximum value that could be inserted is 99.99. To change this, we can use the same statement to increase the precision of the fee to six digits:

```
ALTER TABLE registration
```

```
ALTER COLUMN fee
```

```
TYPE numeric(6,2);
```

This will allow values up to 9999.99 to be inserted into the table.

It is important to note that we cannot alter a column's data type that has a foreign key reference to another table. If there is a foreign key reference, the data type and size have to be the same as the primary key. For example, if we tried to alter the artist_id in the album table, due to the foreign key to the artist_id in the artist table, we would get the following error:

```
ALTER TABLE album
```

```
ALTER COLUMN artist_id
```

```
TYPE VARCHAR (100);
```

Query Results

Query failed because of: error: foreign key constraint "album_artist_id_fkey" cannot be implemented

Video Transcription

[MUSIC PLAYING] With an ALTER TABLE statement, you can also modify the data type as well as the size of a particular column. Let's take a look at the artist table in this case here. That's currently set as of our chart with 120 characters. Typically when you modify the data types or the sizes, it should go larger, depending on the data size.

Being that there's going to be existing data in place, most databases will have a check in this case here. So for example, with a name, we can try to modify this utilizing an ALTER TABLE statement. With the table name, then we're gonna say ALTER column, then the column name, then the type as the keyword. And then we're gonna change it to a varchar, and say we'll try to change it to 10 characters.

In this case here, we can obviously see that there's existing data, in this case, that's larger than 10. If we try to run it, we will typically get an error, being that the value's too small to be able to fit that data type. We can try to modify to something that's like a little bit more reasonable that will fit all the data, like 100 in this case here. Which in this case, it'll make the modification without any issues.

[MUSIC PLAYING]



Your turn! Open the SQL tool by clicking on the LAUNCH DATABASE button below. Then enter in one of the examples above and see how it works. Next, try your own choices for which columns you want the query to

provide.



SUMMARY

The ALTER TABLE command can be used to change the data type and size of a column.

Source: Authored by Vincent Tran