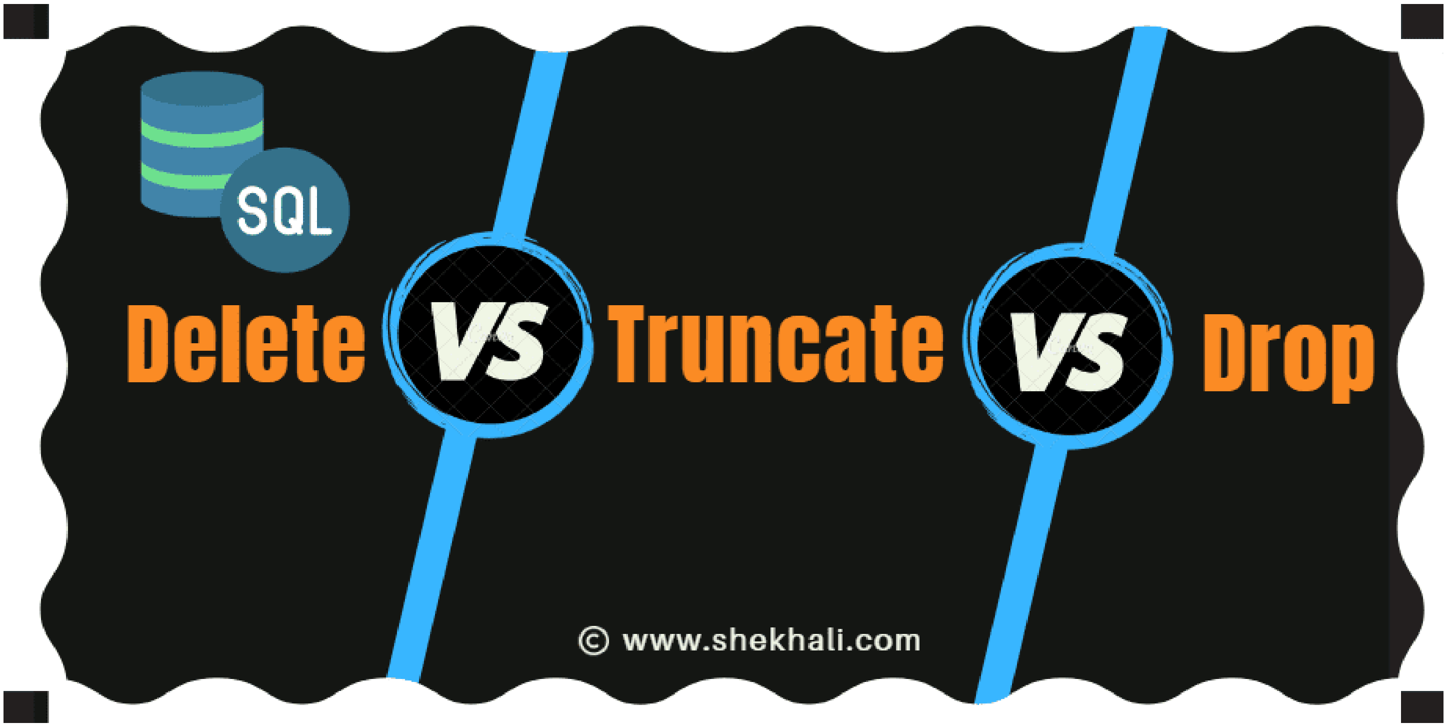
Moaz_Osama_MIP-DA-03_SQL

Mentorness Article / Task 1

When Data Needs to Go: Untangling Truncate, Drop, and Delete



Introduction:

In the intricate world of databases, managing data isn't just about adding and storing information. Sometimes, cleaning house and removing unwanted data becomes essential. That's where our trio of database commands - **Truncate**, **Drop**, and **Delete** - come into play. But choosing the right tool for the job can be tricky! This article unravels the mysteries of these commands, guiding you to make informed decisions when data needs to go.

First, start with an example for creating a database with a table:

```
Create database Mentorness;
```

```
Create table Internship (
        Intern_ID int primary key,
        Name varchar (50) not null,
        Email varchar (50) not null,
        Phone varchar (50) unique,
        Department varchar (50) null
);
```

```
Insert into Internship (Intern_ID, Name, Email, Phone, Department)
Values (1, ‘Moaz’, ‘Moaz @gmail’, ‘+0114’, ‘DA’),
        (2, ‘Jhon’, ‘Jhon @gmail’, ‘+0111’, ‘DA’),
        (3, ‘Mahmoud’, ‘Mahmoud @gmail’, ‘+0121’, ‘DS’),
        (4, ‘Albert’, ‘Albert @gmail’, ‘+9098’, ‘ML’),
        (5, ‘Khan’, ‘Khan @gmail’, ‘+9055’, ‘AI’);
```

```
Select * from Internship;
```

Intern_ID	Name	Email	Phone	Department
1	Moaz	Moaz @gmail	+0114	DA
2	Jhon	Jhon @gmail	+0111	DA
3	Mahmoud	Mahmoud @gmail	+0121	DS
4	Albert	Albert @gmail	+9098	ML
5	Khan	Khan @gmail	+9055	AI

So the previous example is a query for creating a database with a table and insert a records into it to show in the following steps what is the difference between each command that we need to know (“Truncate”, “Drop”, “Delete”)

Let’s start with “Truncate.”

- **Truncate:** Imagine a giant sandbox overflowing with toys. **Truncate** acts like a powerful gust of wind, instantly clearing all the toys away, leaving the sandbox itself (the table structure) pristine and ready for more fun. It's the fastest option, but once the toys are gone, there's no bringing them back.
- So Truncate removes all the record inside the table without the ability of using filters (“Where”).
- Truncate is categorized under “Data Definition Language” (DDL).
- Also Truncate is used for tables only; can’t use it for database.

Let’s take an example by using the previous table that we create it.

```
Truncate Table internship;
```

```
Select * from internship;
```

Intern_ID	Name	Email	Phone	Department

Conclusion: As we notice that after using “Truncate” command the records we inserted before had been removed and when executing the select statement there are no records and the disadvantage of Truncate that we can’t get back those records again.

The second one that we will talk about is “Drop.”

- **Drop:** Think of a dusty old chest filled with outdated belongings. **Drop** is like taking the whole chest and tossing it into a bottomless pit, permanently erasing both the chest (the table) and its contents (the data). It's the quickest way to get rid of things, but remember, once it's gone, it's truly gone forever.
- Drop is categorized under “Data Definition Language” (DDL).
- Drop is used not only for table it also can be used it for database.

Let’s take an example by using the main table that we create it.

```
Drop database mentorness ;
```

*By the previous command we deleted the entire database within its tables.

```
Drop Table internship;
```

*By that command we deleted only that table from the database within its columns.

Conclusion: Drop can’t use filters such as (“Where”) and it’s used for deleting the database structure or the table structure not for deleting a specific record.

The third and the last one we’ll talk about in our article is “Delete” which in my opinion it has many benefits and the best advantage for it; is using filters (“Where”).

- **Delete:** Picture a messy desk cluttered with papers. **Delete** is like carefully sorting through them, picking out specific ones you no longer need and filing them away (in the metaphorical trash bin). It takes more time and effort than just throwing everything away, but it allows you to keep what's important and discard only what's unnecessary. And if you accidentally discard something valuable, you might still be able to retrieve it (depending on the database).
- Delete is categorized under “Data Manipulation Language” (DML).
- Delete is used for records inside the tables.

Let’s take an example by using the main table that we create it.

```
Delete * from internship;
```

That is similar to Truncate

```
Delete from internship
Where Department = “DA”
Select  * from internship;
```

Intern_ID	Name	Email	Phone	Department
3	Mahmoud	Mahmoud @gmail	+0121	DS
4	Albert	Albert @gmail	+9098	ML
5	Khan	Khan @gmail	+9055	AI

So we delete all records that contains department with the value “DA” so row 1,2 had been deleted and the select statement show the remained records.

Conclusion: Choosing the Right Tool for Data Removal

Navigating the database landscape can be complex, but understanding the nuances of Truncate, Drop, and Delete empowers you to make informed decisions when removing data. Remember, each command possesses unique strengths and limitations:

- **Truncate:** excels at rapid removal of all rows, but with the caveat of permanent deletion.
- **Drop:** offers the quickest way to eliminate an entire table, but be wary of its irreversible nature.
- **Delete:** provides precision and control, allowing you to target specific rows based on criteria, with the added benefit of potential rollback.

Ultimately, the optimal choice hinges on your specific requirements. Prioritize speed and bulk removal? Truncate might be your answer. Need permanent elimination of an entire table structure? opts for Drop. Seeking granular control over individual rows and potential reversibility? Delete is your go-to option.

As you embark on your data decluttering journey, remember these guiding principles:

- **Always plan ahead and back up your data.**
- **Consider the scope and reversibility of each command.**
- **Choose the tool that aligns best with your specific needs.**