



**Kampus
Merdeka**
INDONESIA JAYA



SQL STATEMENT: **ALTER**

Bachelor of Information Systems

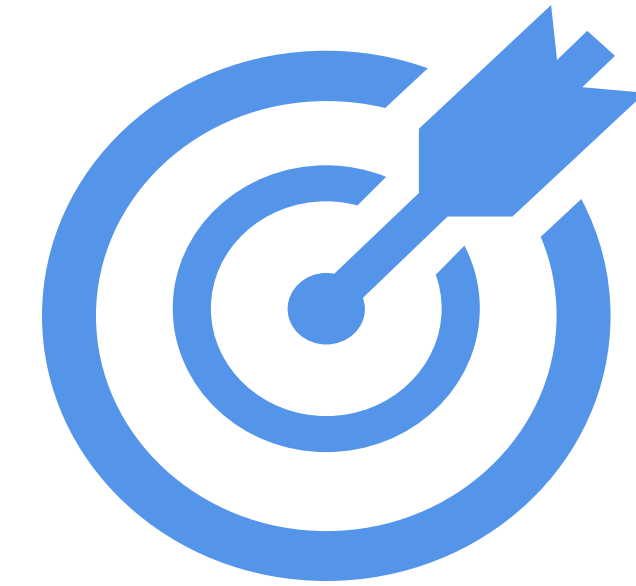


INTRODUCTION TO
DATABASE AND INFORMATION SYSTEM



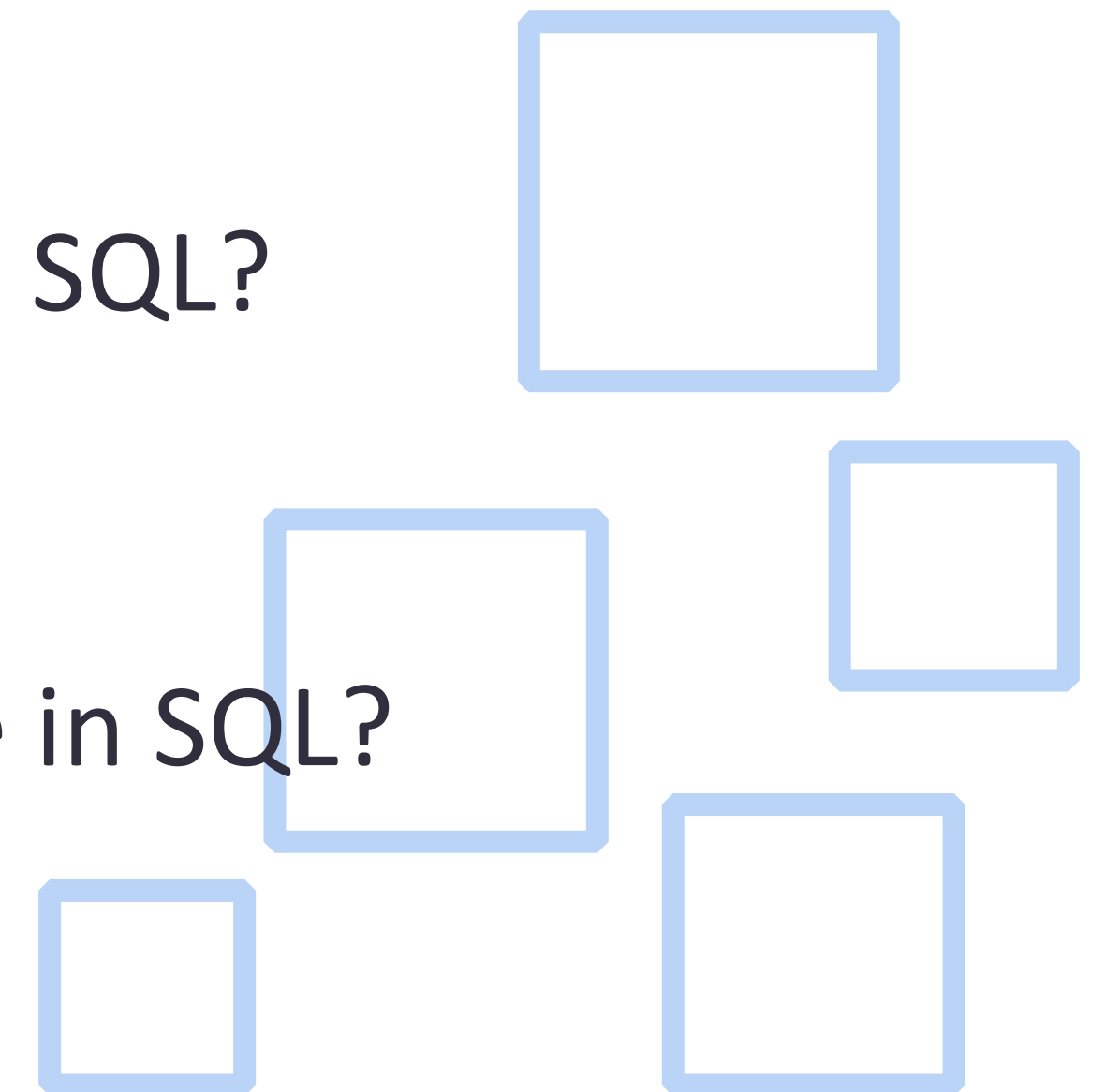
Learning Objective(s)

.....



This material should address the following question(s).

- ? How do we add columns in an existing table in SQL?
- ? How do we change columns definition of an existing table in SQL?
- ? How do we delete columns from an existing table in SQL?
- ? How do we change the name of columns of an existing table in SQL?



Question



How do we **add columns** in an existing table in SQL?

ALTER TABLE ADD

- You may require to add the columns in the existing table.
- Instead of creating a whole table or database again you can easily add single and multiple columns using the **ADD** clause.
- Syntax:
ALTER TABLE TableName
ADD column_name column-definition;

Example:

Student					
FIELD	TYPE	NULL	KEY	DEFAULT	EXTRA
id	int(11)	NO	PRI		
name	varchar(50)	NO			
phone_number	int(12)	YES		NULL	
email	varchar(50)	YES		NULL	

Add **birth_place** with data type VARCHAR and the size is 50 to the **Student** table

```
ALTER TABLE Student  
ADD birth_place VARCHAR(50);
```

```
DESC Student;
```

Output:

Student					
FIELD	TYPE	NULL	KEY	DEFAULT	EXTRA
id	int(11)	NO	PRI		
name	varchar(50)	NO			
phone_number	int(12)	YES		NULL	
email	varchar(50)	YES		NULL	
birth_place	varchar(50)	YES		NULL	

ALTER TABLE ADD

- You may require to add the columns in the existing table.
- Instead of creating a whole table or database again you can easily add single and multiple columns using the **ADD** clause.
- Syntax:
ALTER TABLE TableName
ADD column_name column-definition;

Example:

Student					
FIELD	TYPE	NULL	KEY	DEFAULT	EXTRA
id	int(11)	NO	PRI		
name	varchar(50)	NO			
phone_number	int(12)	YES		NULL	
email	varchar(50)	YES		NULL	

Add **birth_place** VARCHAR(50) and
date_of_birth DATE to the **Student** table

```
ALTER TABLE Student
ADD (
    birth_place VARCHAR(50),
    date_of_birth DATE
);
```

```
DESC Student;
```

Output:

Student					
FIELD	TYPE	NULL	KEY	DEFAULT	EXTRA
id	int(11)	NO	PRI		
name	varchar(50)	NO			
phone_number	int(12)	YES		NULL	
email	varchar(50)	YES		NULL	
birth_place	varchar(50)	YES		NULL	
date_of_birth	date	YES		NULL	

Question



How do we **change columns definition** of an existing table in SQL?

ALTER TABLE MODIFY

- The **MODIFY** clause is used for changing the column definition of the existing table.
- Syntax:
ALTER TABLE TableName
MODIFY column_name column-definition;

Example:

Student					
FIELD	TYPE	NULL	KEY	DEFAULT	EXTRA
id	int(11)	NO	PRI		
name	varchar(50)	NO			
phone_number	int(12)	YES		NULL	
email	varchar(50)	YES		NULL	

Modify the data type of **phone_number** to VARCHAR and the size to 15

```
ALTER TABLE Student  
MODIFY phone_number VARCHAR(15);
```

```
DESC Student;
```

Output:

Student					
FIELD	TYPE	NULL	KEY	DEFAULT	EXTRA
id	int(11)	NO	PRI		
name	varchar(50)	NO			
phone_number	varchar(15)	YES		NULL	
email	varchar(50)	YES		NULL	

Question



How do we **delete columns** from an existing table in SQL?

ALTER TABLE DROP

- Instead of deleting the whole table or database, you can use **DROP** clause for deleting columns.
- Syntax:
ALTER TABLE TableName
DROP COLUMN column_name;

Example:

Student					
FIELD	TYPE	NULL	KEY	DEFAULT	EXTRA
id	int(11)	NO	PRI		
name	varchar(50)	NO			
phone_number	int(12)	YES		NULL	
email	varchar(50)	YES		NULL	

Delete **phone_number** from the **Student** table

```
ALTER TABLE Student  
DROP COLUMN phone_number;
```

```
DESC Student;
```

Output:

Student					
FIELD	TYPE	NULL	KEY	DEFAULT	EXTRA
id	int(11)	NO	PRI		
name	varchar(50)	NO			
email	varchar(50)	YES		NULL	

Question



How do we **change the name of columns** of an existing table in SQL?

ALTER TABLE RENAME

- The **RENAME** clause is used for changing the name of columns or fields of the existing table.
- Syntax:
ALTER TABLE TableName
RENAME COLUMN old_name **TO**
new_name;

Example:

Student					
FIELD	TYPE	NULL	KEY	DEFAULT	EXTRA
id	int(11)	NO	PRI		
name	varchar(50)	NO			
phone_number	int(12)	YES		NULL	
email	varchar(50)	YES		NULL	

Change **phone_number** to **phone**

```
ALTER TABLE Student  
RENAME COLUMN phone_number TO phone;
```

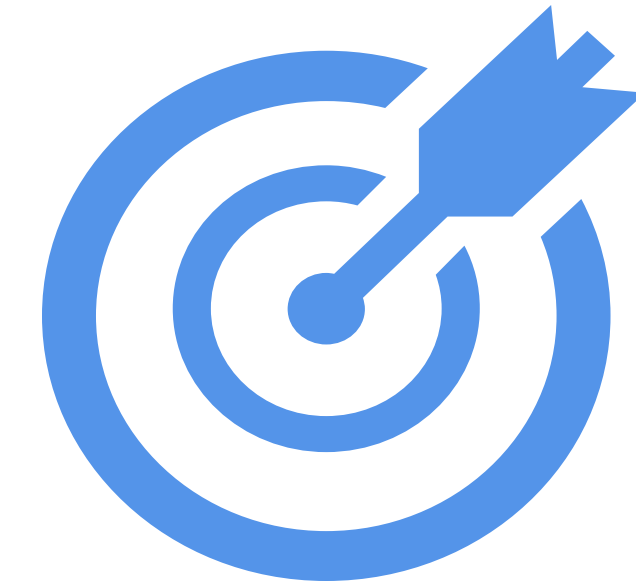
```
DESC Student;
```

Output:

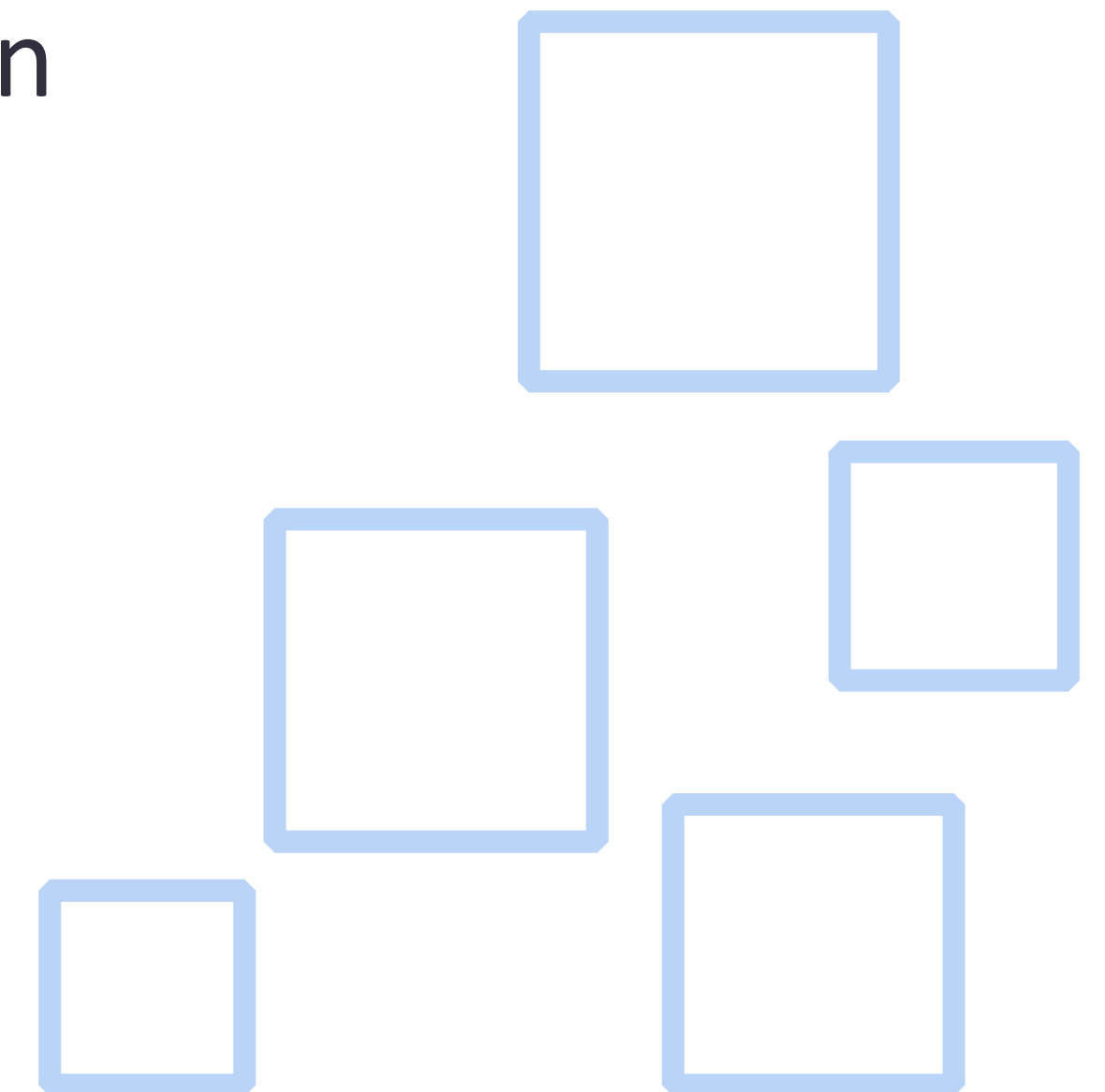
Student					
FIELD	TYPE	NULL	KEY	DEFAULT	EXTRA
id	int(11)	NO	PRI		
name	varchar(50)	NO			
phone	int(12)	YES		NULL	
email	varchar(50)	YES		NULL	

Conclusion

.....



- ✓ Use **ALTER TABLE ADD** to add columns in an existing table in SQL.
- ✓ Use **ALTER TABLE MODIFY** to change columns definition of an existing table in SQL.
- ✓ Use **ALTER TABLE DROP** to delete columns from an existing table in SQL.
- ✓ Use **ALTER TABLE RENAME** to change the name of columns of an existing table in SQL.



References

.....



- R, Elmasri, et. al., Fundamentals of Database Systems.
- A. Silberschatz, et. al., Database System Concepts.
- R. K. Rainer, et. al., Introduction to Information Systems.
- G. M. Marakas et. al., Introduction to Information Systems: Essentials for The e-Business Enterprise.

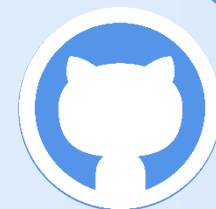


Course



Mario E. S. Simaremare

@simaremare



Lecturer



Samuel I. G. Situmeang

@exemuel

