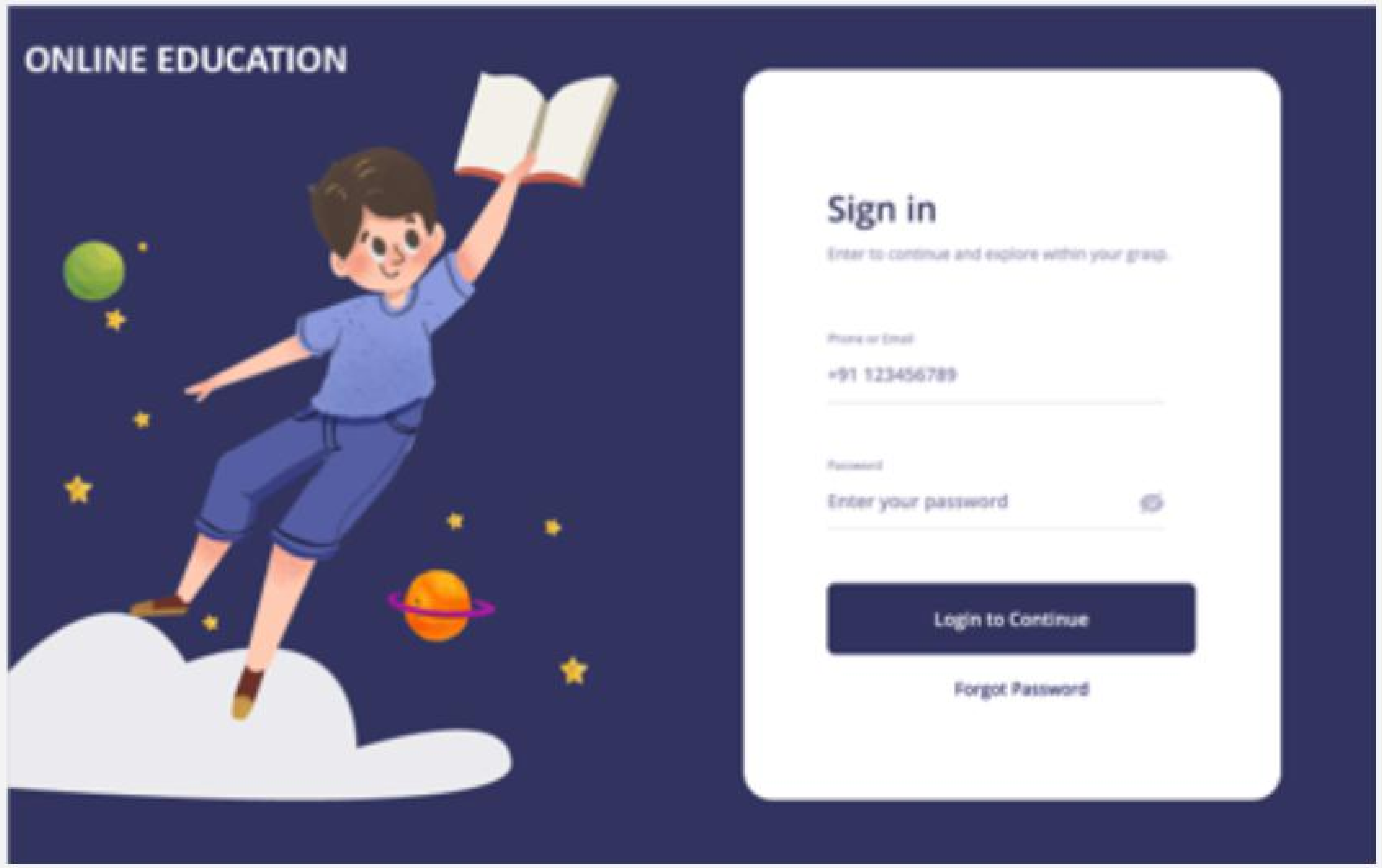
Think and Tell

- When you register on any website for the first time, what happens to the data you enter?

- How does your application identify your username and password every time you log in?



ONLINE EDUCATION

Sign in

Enter to continue and explore within your grasp.

Phone or Email

+91 123456789

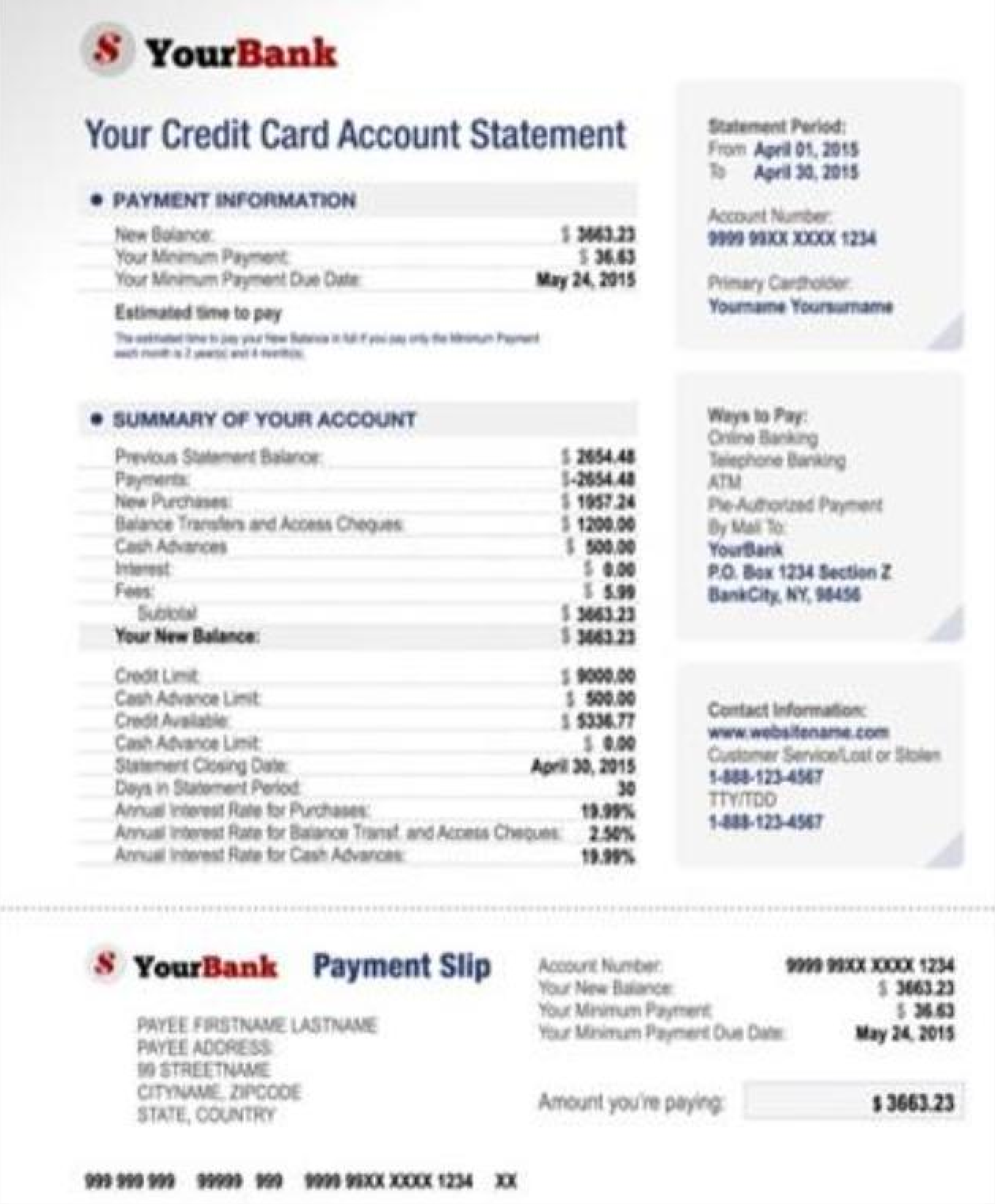
Password

Enter your password

Slide Note

Bank Transactions

- A bank customer wishes to know the details of all his account transactions in the last six months.
- Where do you think the data is stored?
- How will the data be provided to the customer?



YourBank

Your Credit Card Account Statement

• PAYMENT INFORMATION

New Balance:	$ 3663.23
Your Minimum Payment:	$ 36.63
Your Minimum Payment Due Date:	May 24, 2015

Estimated time to pay

The estimated time to pay your New Balance in full if you pay only the Minimum Payment each month is 2 year(s) and 4 month(s).

• SUMMARY OF YOUR ACCOUNT

Previous Statement Balance:	$ 2654.48
Payments:	$-2654.48
New Purchases:	$ 1957.24
Balance Transfers and Access Cheques:	$ 1200.00
Cash Advances	$ 500.00
Interest:	$ 0.00
Fees:	$ 5.99
Subtotal	$ 3663.23
Your New Balance:	$ 3663.23
Credit Limit:	$ 9000.00
Cash Advance Limit:	$ 500.00
Credit Available:	$ 5336.77
Cash Advance Limit:	$ 0.00
Statement Closing Date:	April 30, 2015
Days in Statement Period:	30
Annual Interest Rate for Purchases:	19.99%
Annual Interest Rate for Balance Transf. and Access Cheques:	2.50%
Annual Interest Rate for Cash Advances:	19.99%

Statement Period:
From April 01, 2015
To April 30, 2015

Account Number:
9999 99XX XXXX 1234

Primary Cardholder:
Yourname Yoursurname

Ways to Pay:
Online Banking
Telephone Banking
ATM
Pre-Authorized Payment
By Mail To:
YourBank
P.O. Box 1234 Section Z
BankCity, NY, 98456

Contact Information:
www.websitename.com
Customer Service/Lost or Stolen
1-888-123-4567
TTY/TDD
1-888-123-4567

YourBank **Payment Slip**

Account Number:	9999 99XX XXXX 1234
Your New Balance:	$ 3663.23
Your Minimum Payment:	$ 36.63
Your Minimum Payment Due Date:	May 24, 2015

PAYEE FIRSTNAME LASTNAME
PAYEE ADDRESS:
99 STREETNAME
CITYNAME, ZIPCODE
STATE, COUNTRY

Amount you're paying: $ 3663.23

999 999 999 99999 999 9999 99XX XXXX 1234 XX

Menu

00:02 00:00:43

Banking Facilities

- From where does your mobile banking app get the information displayed on your screen?
- Where does your monthly statement displaying the transactions made from your bank account come from?
- When you withdraw or deposit cash, your account balance gets updated automatically. How does this happen?

Locker facilities

Currency exchange

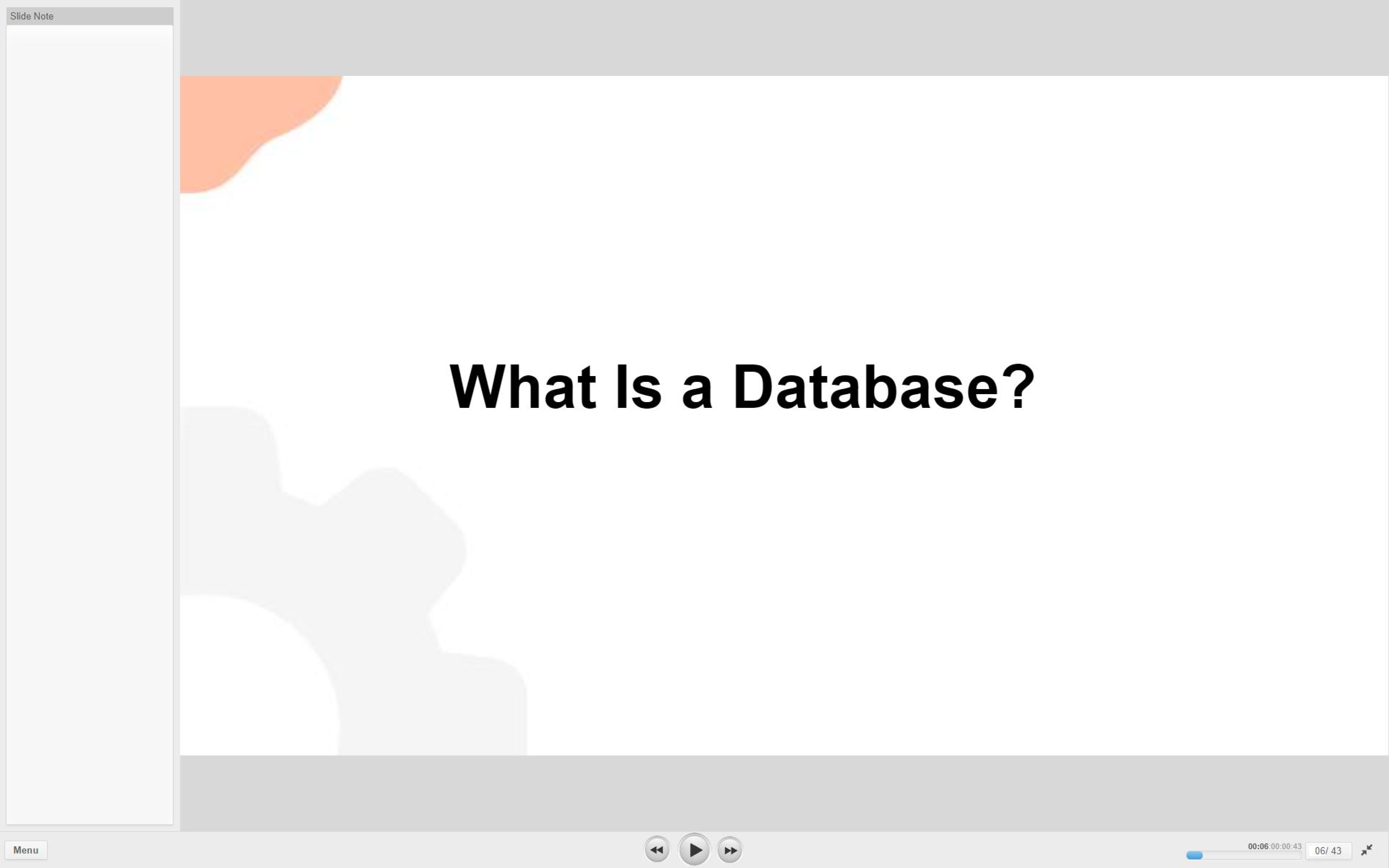
Notary public service

Data Modeling Using RDBMS (DDL & DML)

Slide Note

Learning Objectives

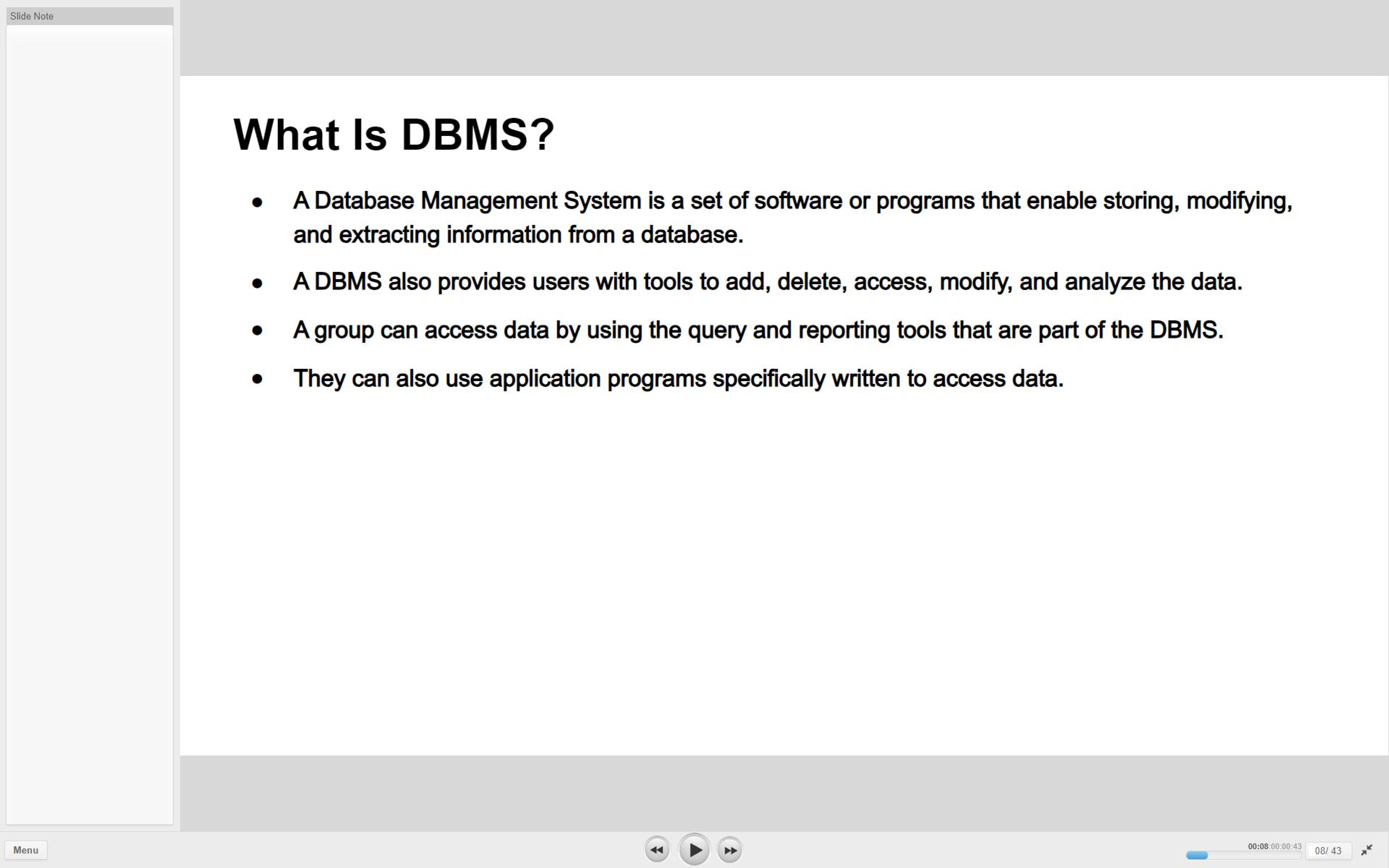
- What is a Database?
- Introduction of SQL
- Introduction to MySQL
- DDL – Data Definition Language
- DML – Data Manipulation Language
- Define Constraints



What Is a Database?

What Is a Database?

- A database is a collection of logically related data. It is used to search data to answer queries.
- A database may be designed for batch, real-time, or online processing.

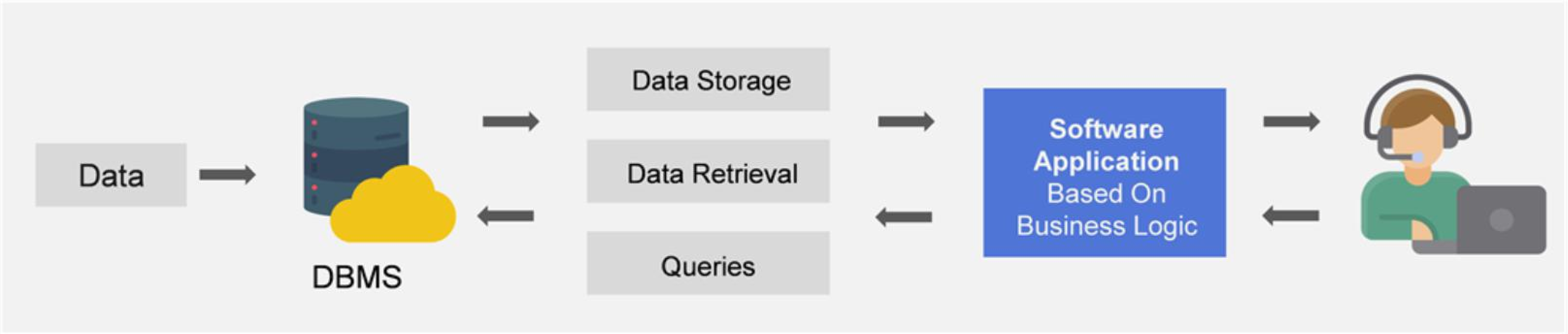


What Is DBMS?

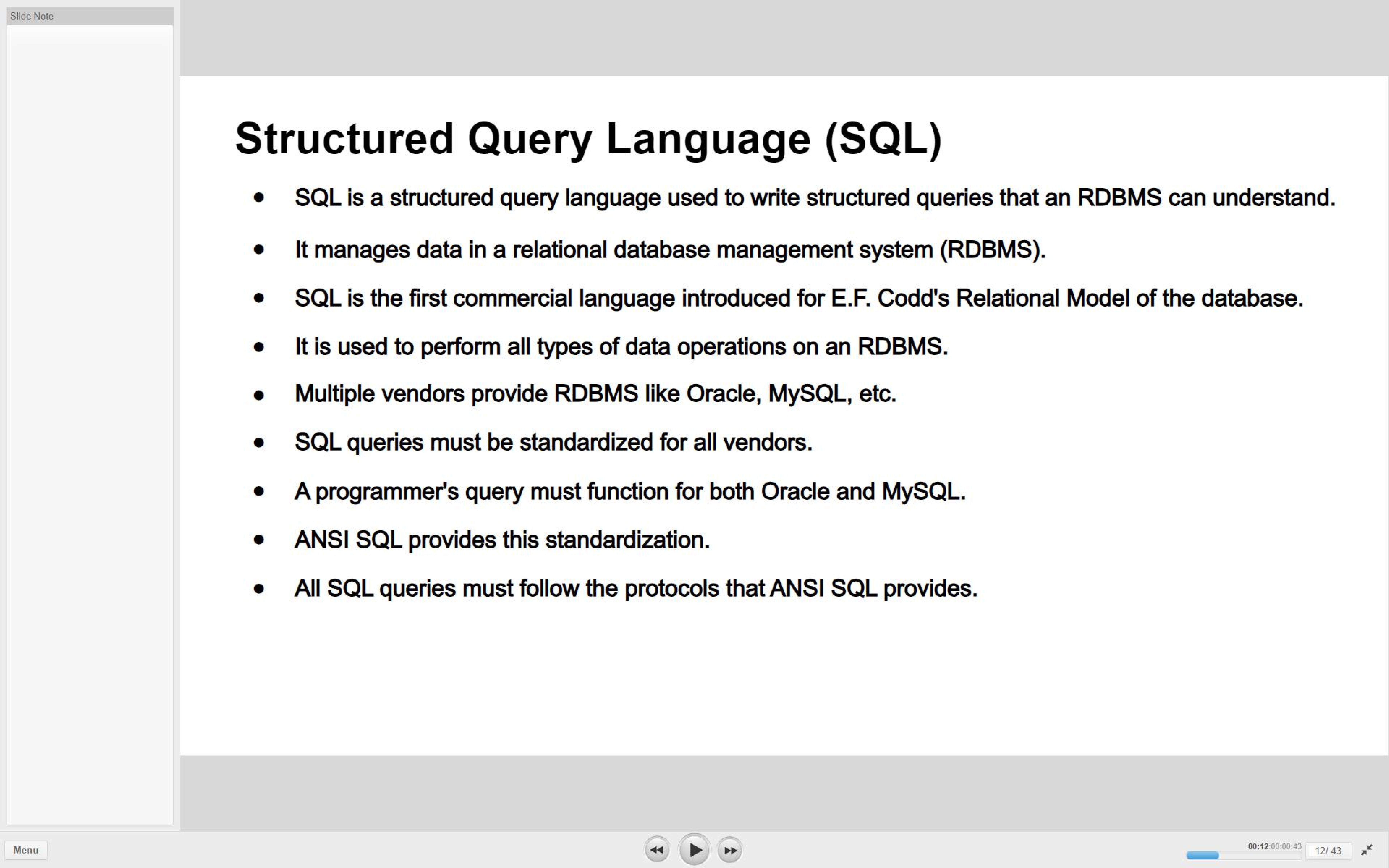
- A Database Management System is a set of software or programs that enable storing, modifying, and extracting information from a database.
- A DBMS also provides users with tools to add, delete, access, modify, and analyze the data.
- A group can access data by using the query and reporting tools that are part of the DBMS.
- They can also use application programs specifically written to access data.

Working of DBMS

- Data – Data is facts and statistics collected for reference or analysis.
- Data Storage – In programming languages, data is stored in a database.
- Data Retrieval – Programming languages use query languages to retrieve data from the database.



Introduction of SQL

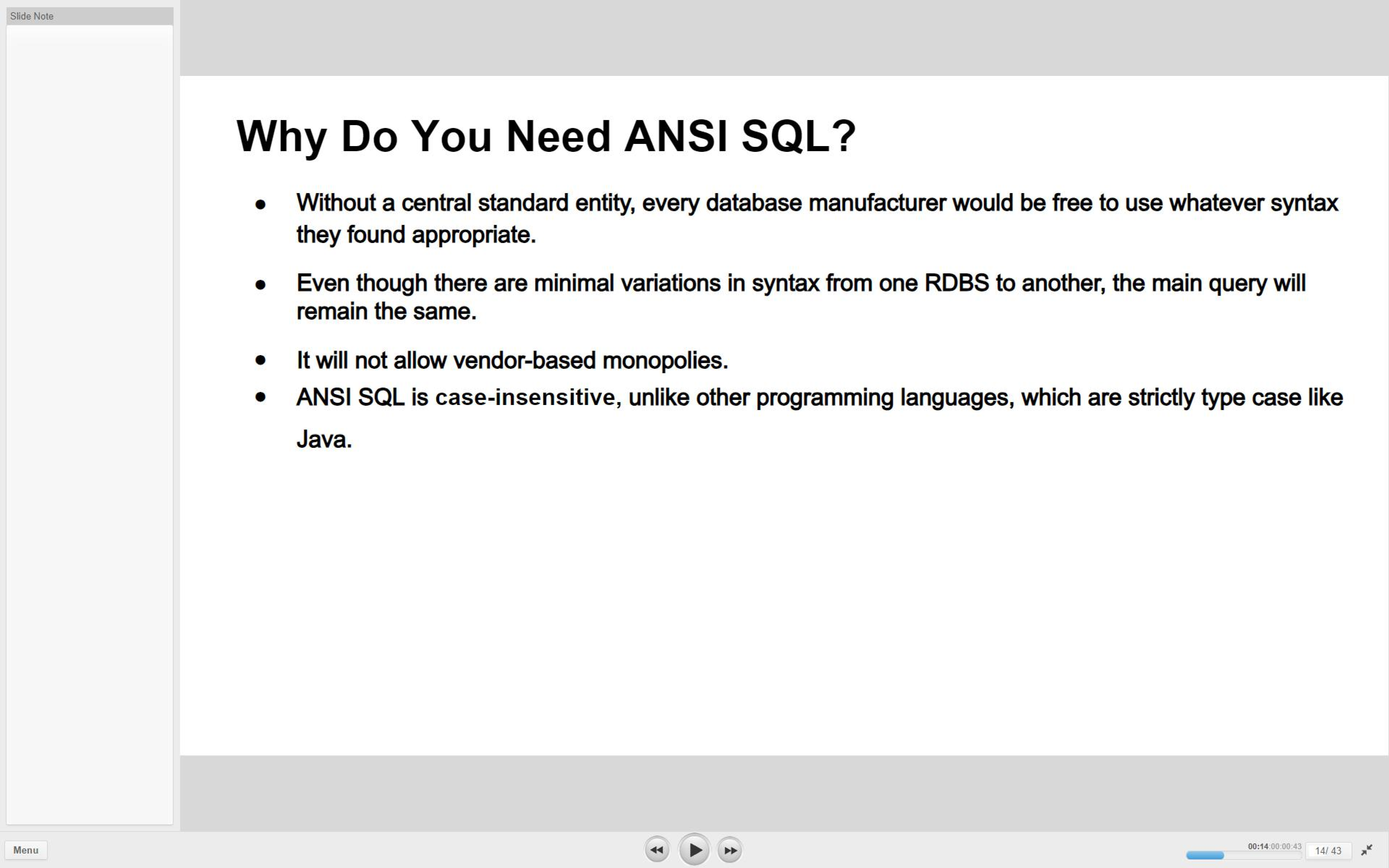


Structured Query Language (SQL)

- SQL is a structured query language used to write structured queries that an RDBMS can understand.
- It manages data in a relational database management system (RDBMS).
- SQL is the first commercial language introduced for E.F. Codd's Relational Model of the database.
- It is used to perform all types of data operations on an RDBMS.
- Multiple vendors provide RDBMS like Oracle, MySQL, etc.
- SQL queries must be standardized for all vendors.
- A programmer's query must function for both Oracle and MySQL.
- ANSI SQL provides this standardization.
- All SQL queries must follow the protocols that ANSI SQL provides.

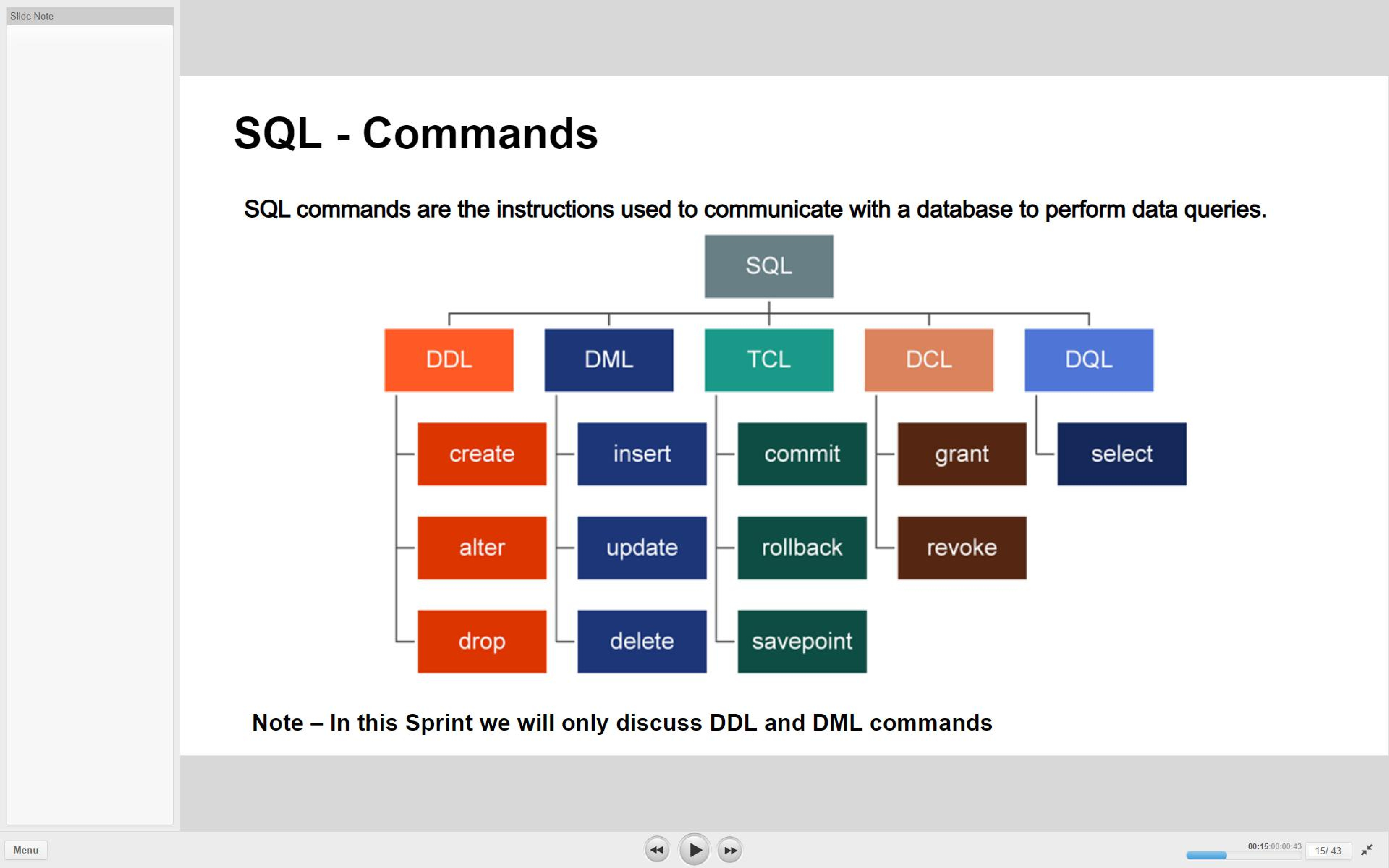
ANSI SQL

- American National Standards Institute (ANSI):
  - ANSI is a private, non-profit organization responsible for developing voluntary standards for products, services, processes, systems, and personnel in the United States.
  - SQL was adopted as a standard by the American National Standards Institute (ANSI) in 1986 as SQL- 86 and by the International Organization for Standardization (ISO) in 1987.



Why Do You Need ANSI SQL?

- Without a central standard entity, every database manufacturer would be free to use whatever syntax they found appropriate.
- Even though there are minimal variations in syntax from one RDBS to another, the main query will remain the same.
- It will not allow vendor-based monopolies.
- ANSI SQL is case-insensitive, unlike other programming languages, which are strictly type case like Java.



SQL - Commands

SQL commands are the instructions used to communicate with a database to perform data queries.

SQL

DDL
- create
- alter
- drop

DML
- insert
- update
- delete

TCL
- commit
- rollback
- savepoint

DCL
- grant
- revoke

DQL
- select

Note – In this Sprint we will only discuss DDL and DML commands

Quick Check

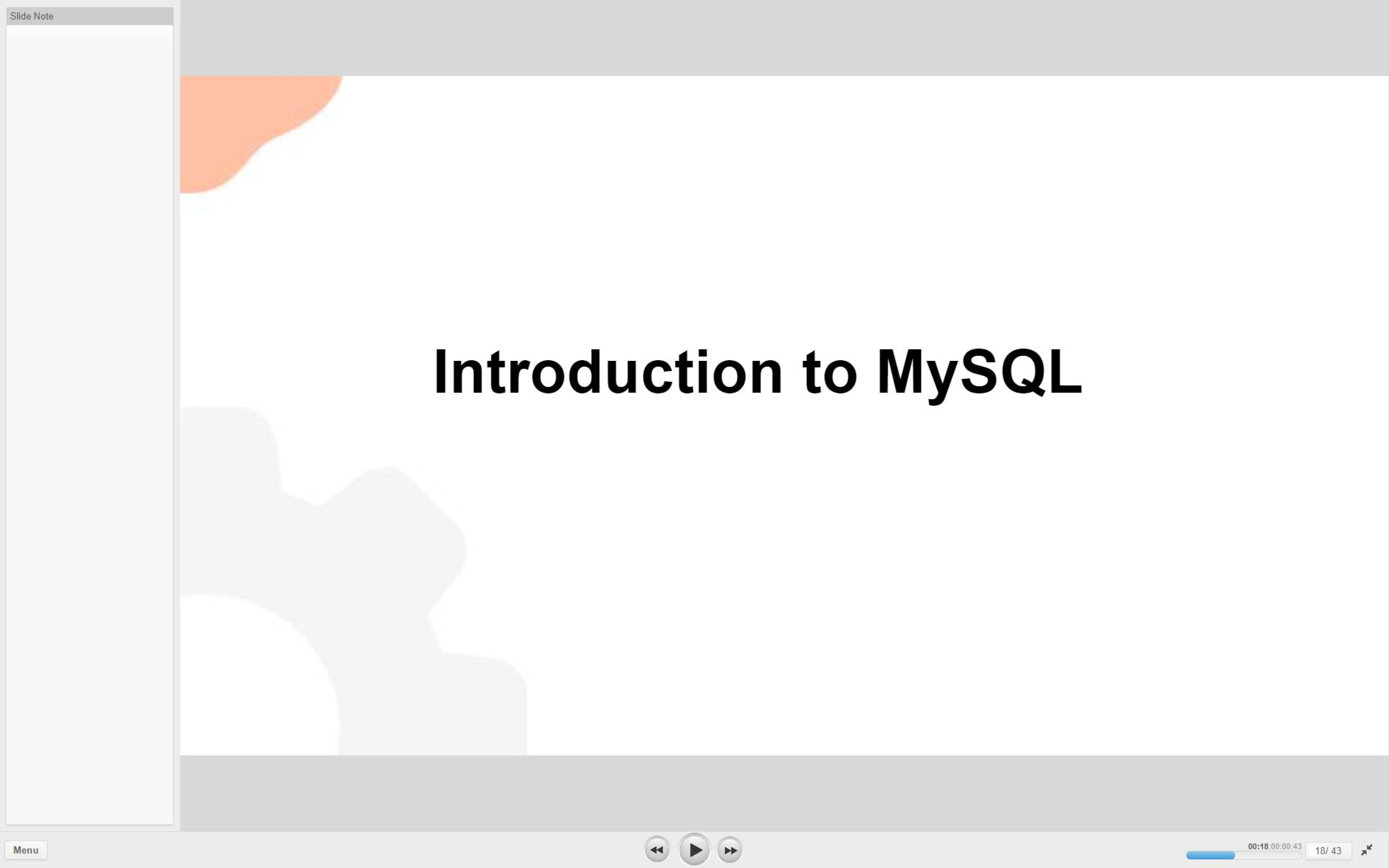
What is the full form of SQL?

1. Structured Query Language
2. Structured Query List
3. Simple QueryLanguage
4. None of the above

Quick Check: Solution

What is the full form of SQL?

1. Structured Query Language
2. Structured Query List
3. Simple QueryLanguage
4. None of the above

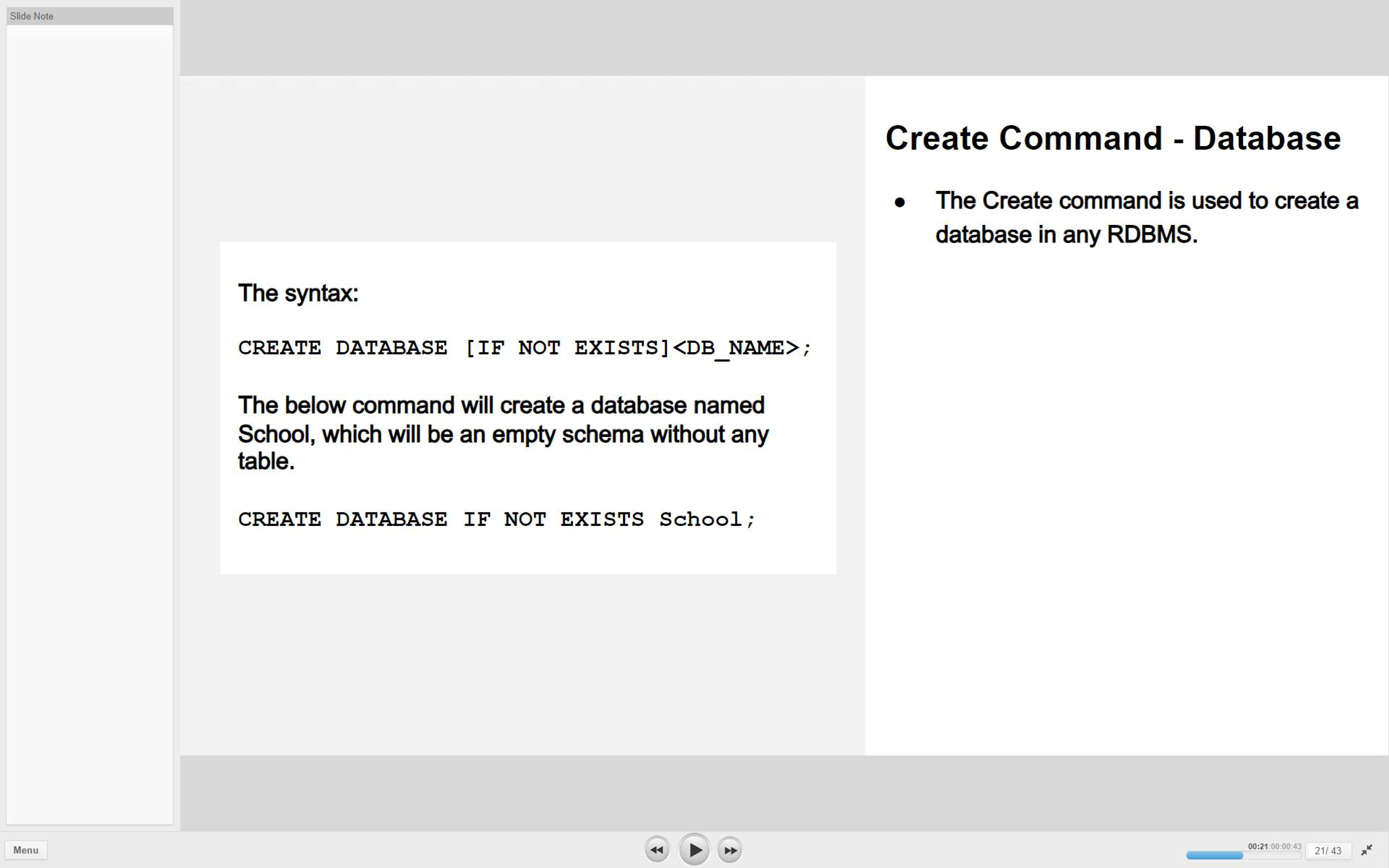


Introduction to MySQL

Introduction to MySQL

- MySQL is an open-source Relational Database Management System (RDBMS)
- One of the most popular open-source RDBMS
- Developed, distributed, and supported by Oracle Corporation
- Runs on a server
- Easy to use

DDL – Data Definition Language



Create Command - Database

- The Create command is used to create a database in any RDBMS.

The syntax:

```
CREATE DATABASE [IF NOT EXISTS]<DB_NAME>;
```

The below command will create a database named School, which will be an empty schema without any table.

```
CREATE DATABASE IF NOT EXISTS School;
```

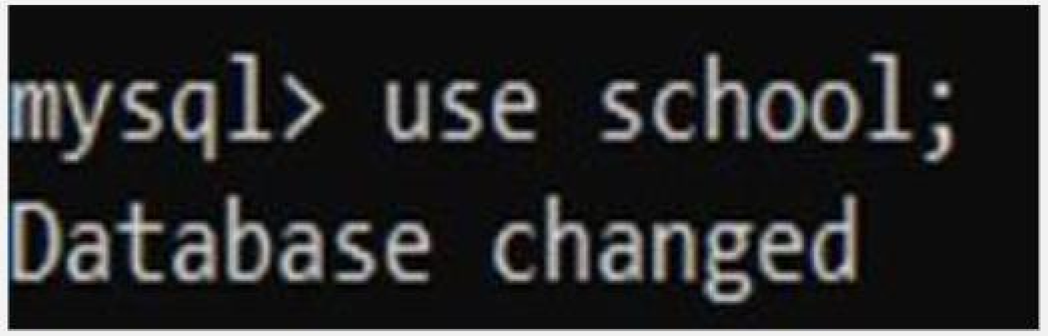
Slide Note

- You can use any database that you have created and work with the tables in the database.

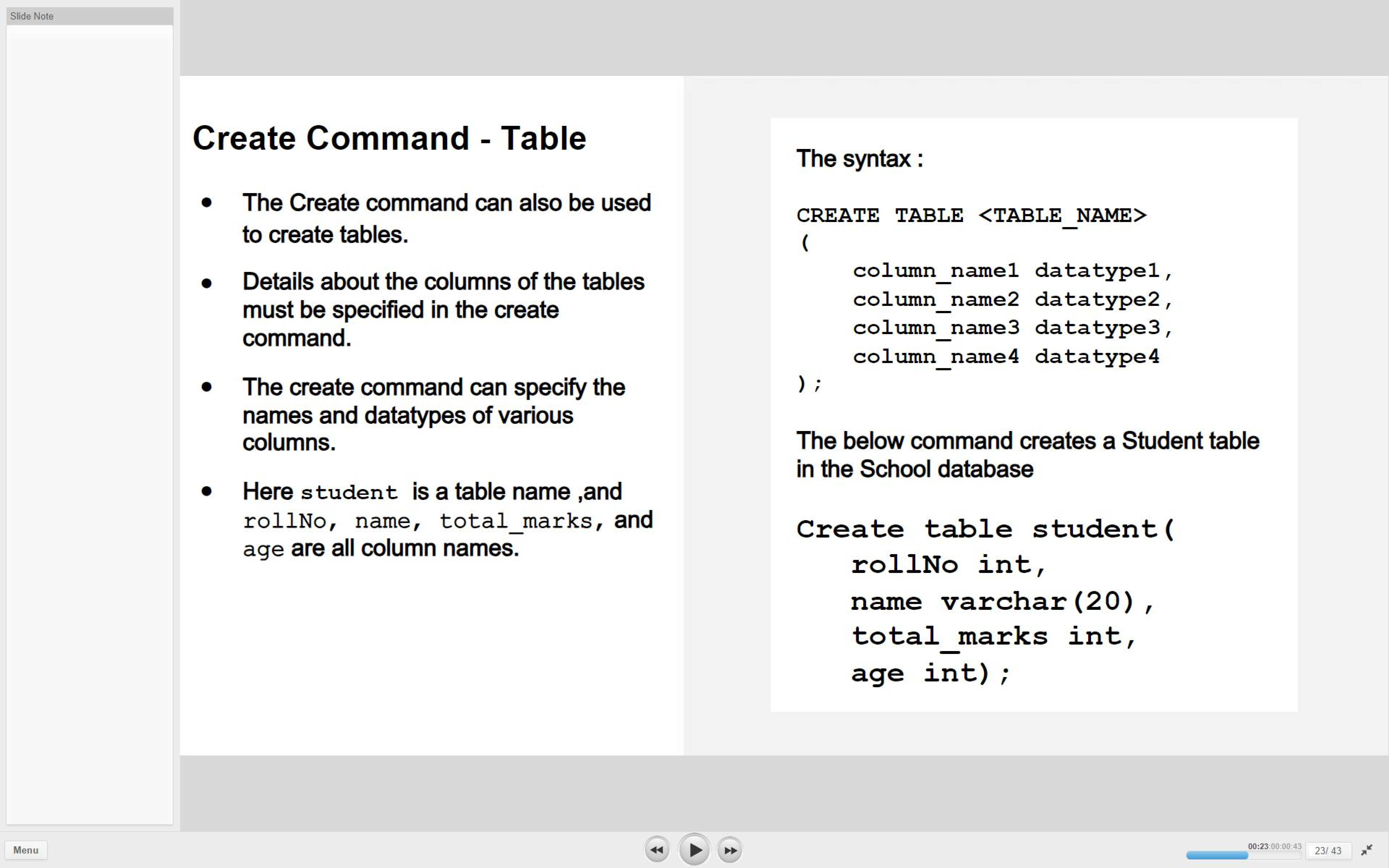
Selecting a Database to Work

- SQL command USE is used to select a particular database.

```
USE <database_name>;
```



```
mysql> use school;
Database changed
```



Create Command - Table

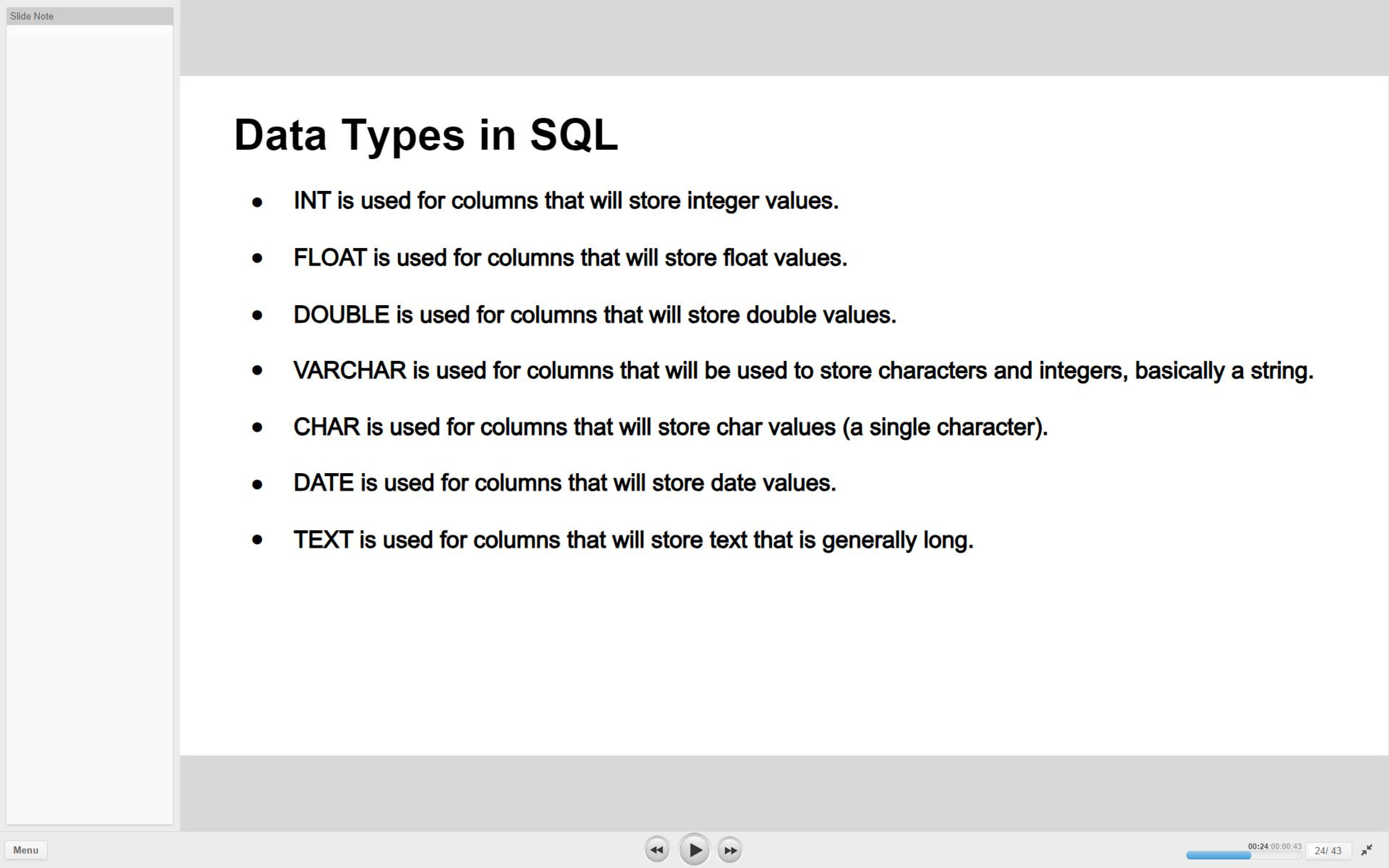
- The Create command can also be used to create tables.
- Details about the columns of the tables must be specified in the create command.
- The create command can specify the names and datatypes of various columns.
- Here `student` is a table name ,and `rollNo`, `name`, `total_marks`, and `age` are all column names.

The syntax :

```
CREATE TABLE <TABLE_NAME>
(
    column_name1 datatype1,
    column_name2 datatype2,
    column_name3 datatype3,
    column_name4 datatype4
);
```

The below command creates a Student table in the School database

```
Create table student(
   rollNo int,
   name varchar(20),
   total_marks int,
   age int);
```

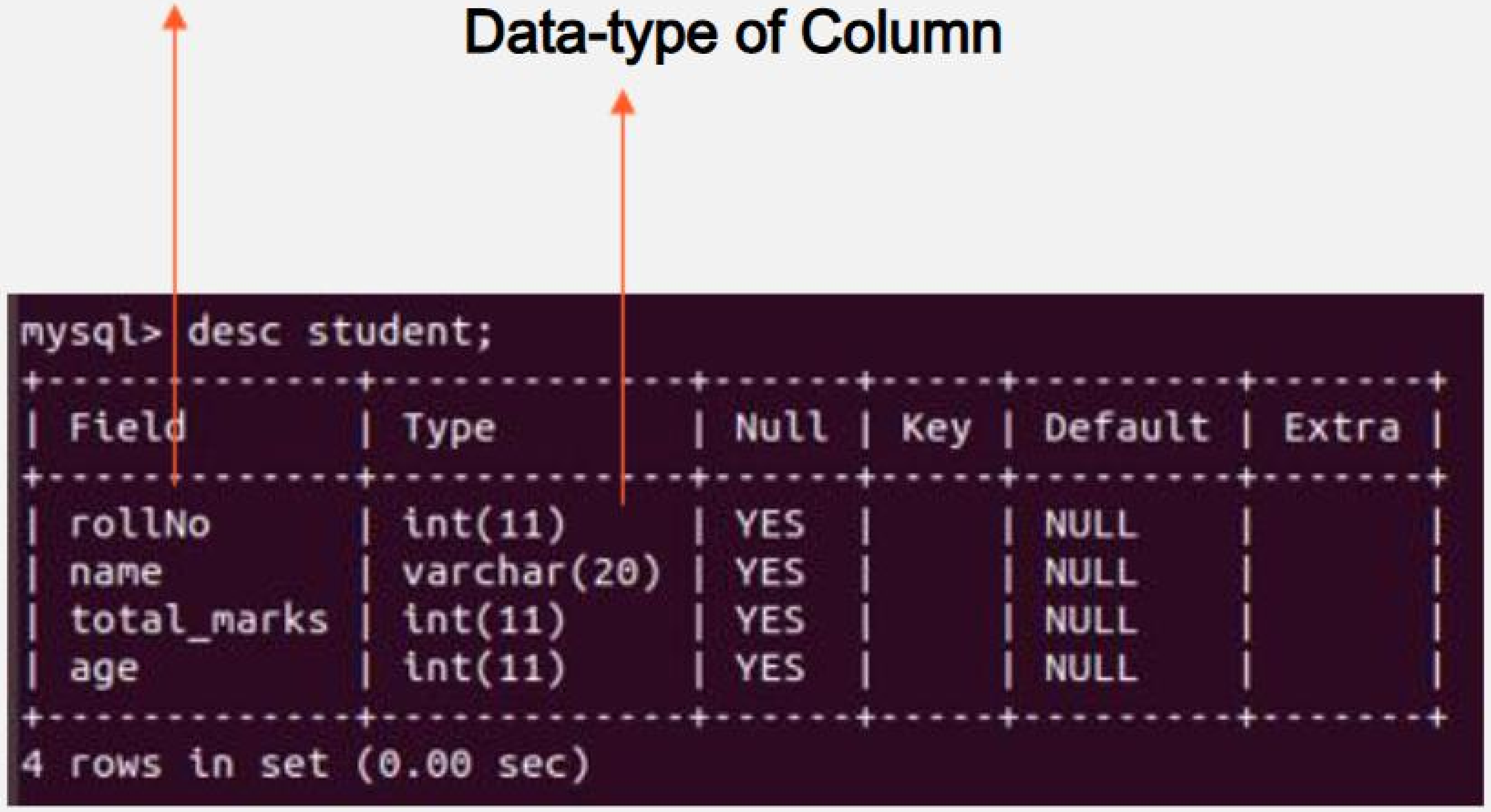


Data Types in SQL

- INT is used for columns that will store integer values.
- FLOAT is used for columns that will store float values.
- DOUBLE is used for columns that will store double values.
- VARCHAR is used for columns that will be used to store characters and integers, basically a string.
- CHAR is used for columns that will store char values (a single character).
- DATE is used for columns that will store date values.
- TEXT is used for columns that will store text that is generally long.

Describe Command

- Describe (or desc) is a command that shows the structure of a table, which includes:
  - name of the column
  - data-type of the column
  - The nullability of the column means that it can contain null values or not.

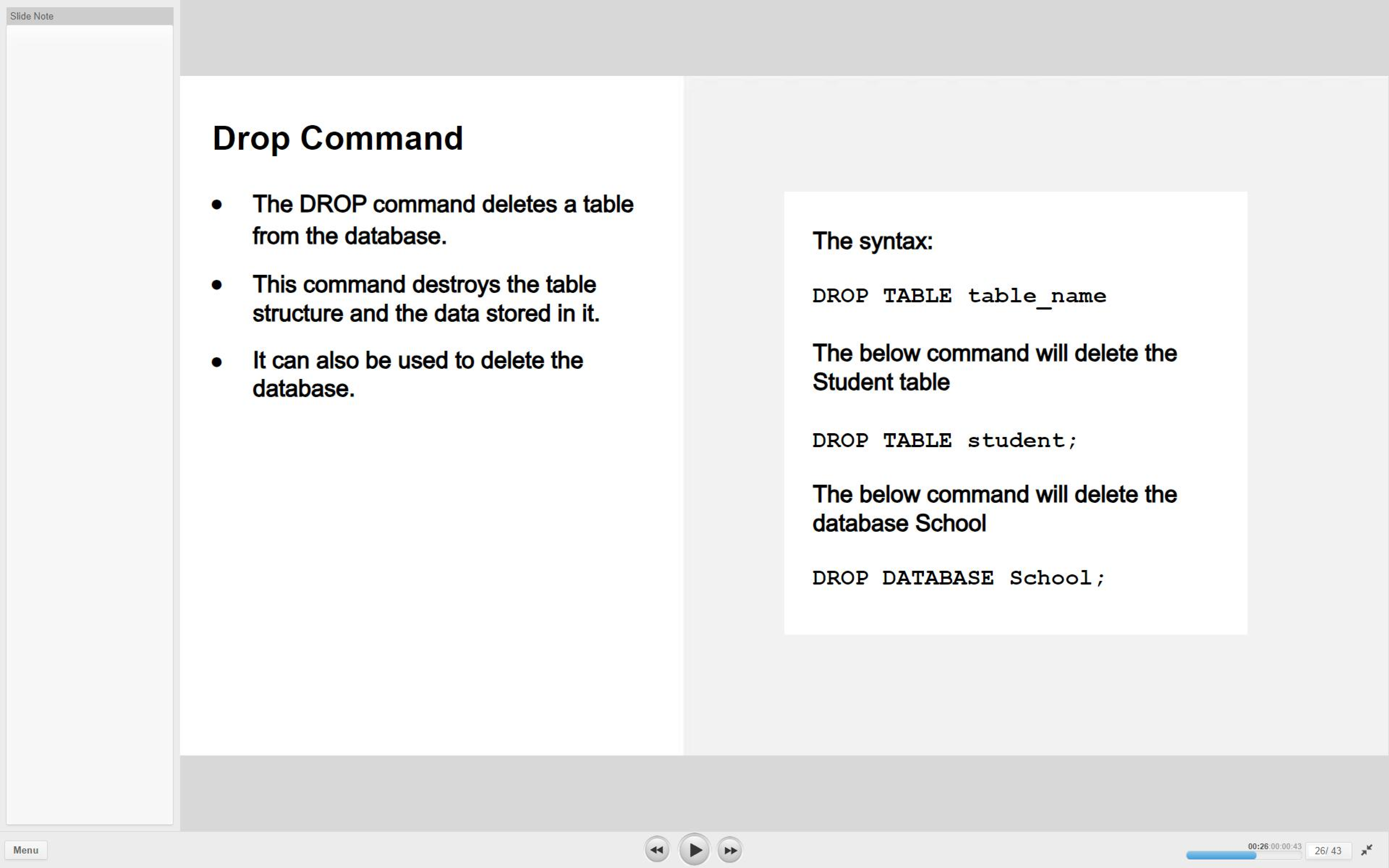


Name of Column

Data-type of Column

```
mysql> desc student;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| rollNo      | int(11)     | YES  |     | NULL    |       |
| name        | varchar(20) | YES  |     | NULL    |       |
| total_marks | int(11)     | YES  |     | NULL    |       |
| age         | int(11)     | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

```
DESCRIBE <table_name>;
```



Drop Command

- The DROP command deletes a table from the database.
- This command destroys the table structure and the data stored in it.
- It can also be used to delete the database.

The syntax:

```
DROP TABLE table_name
```

The below command will delete the Student table

```
DROP TABLE student;
```

The below command will delete the database School

```
DROP DATABASE School;
```

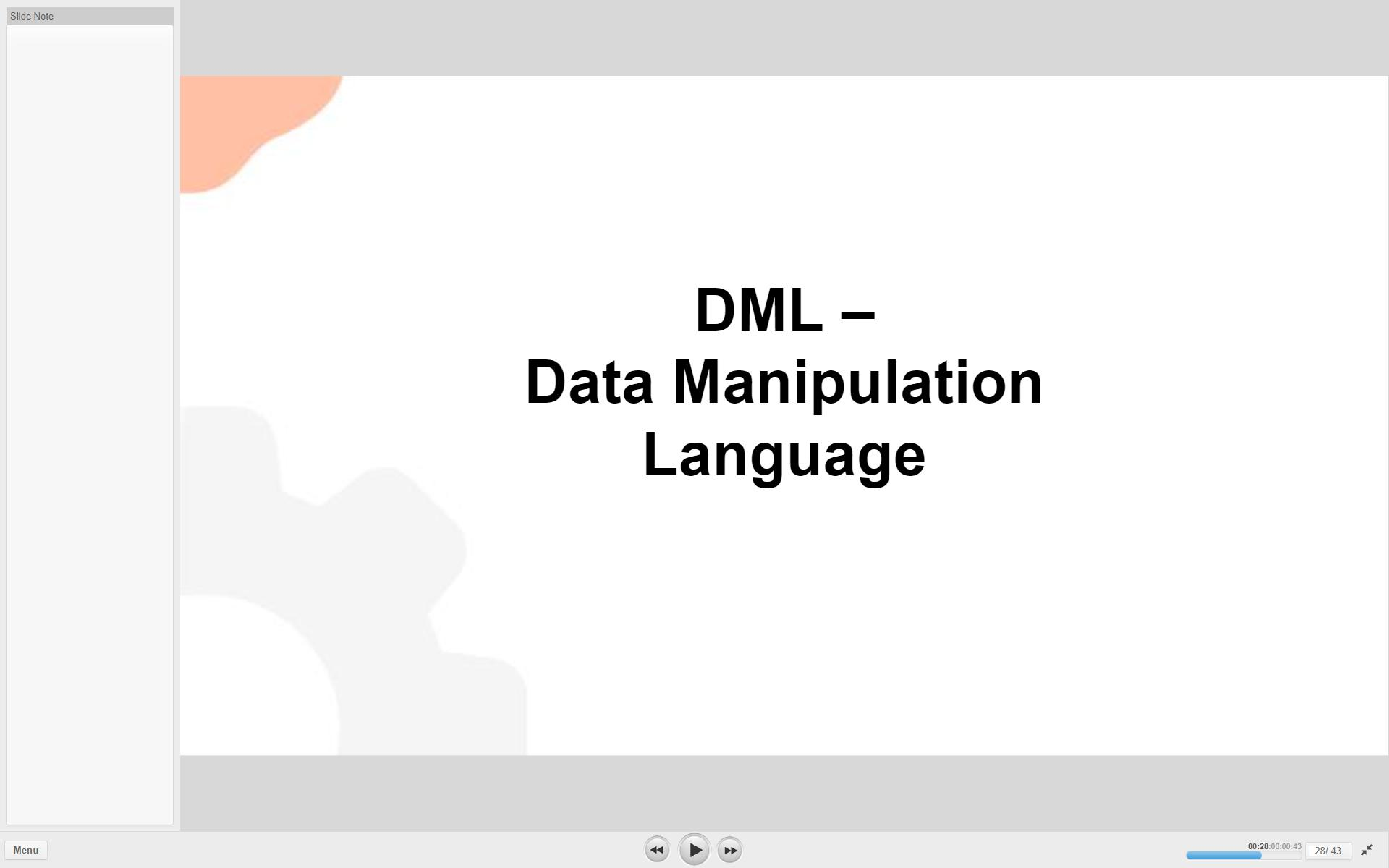
Employee Database

Create a database with the name EmployeeDatabase. This database creates an employee table with the following column names: empNo, dob, firstname, lastname, gender, hiredate, designation, deptName).

Here, empNo will be of type int, and all other attributes will be of type varchar.

Click here for the solution.
MySQL must be used for the demonstration.

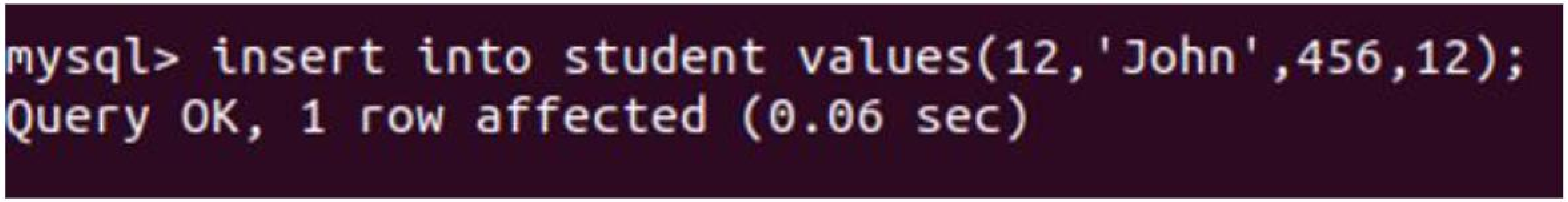
DEMO



DML –
Data Manipulation Language

INSERT Command

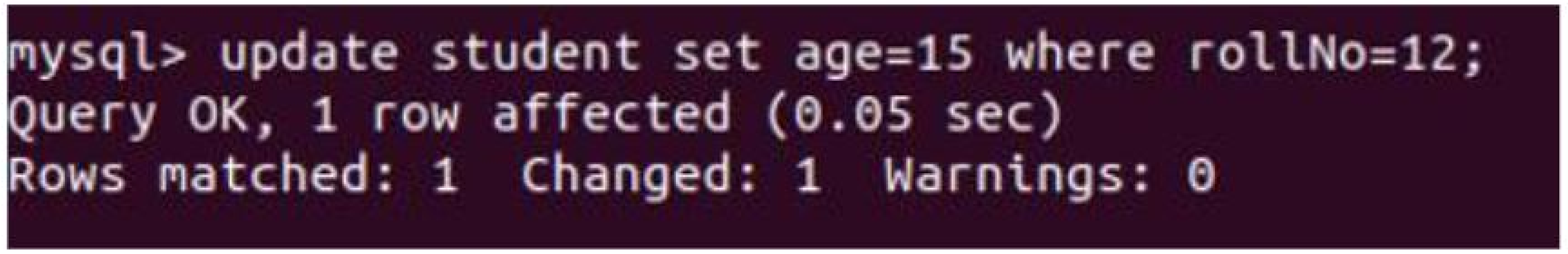
- The INSERT command is used to insert data into a table.
- The syntax:
    - `INSERT INTO table_name VALUES (data1, data2, ...)`
- To insert values into the Student table, use the below command:
    - `INSERT INTO student VALUES (12, 'John', 456, 12);`



```
mysql> insert into student values(12,'John',456,12);
Query OK, 1 row affected (0.06 sec)
```

UPDATE Command

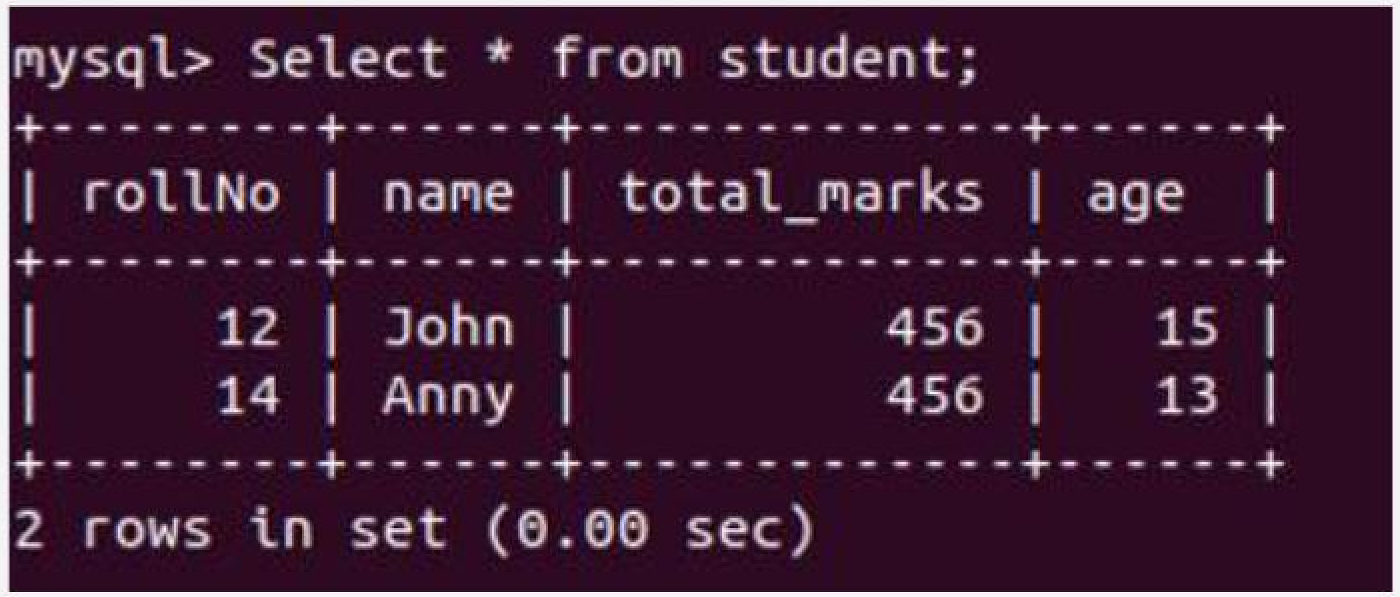
- The UPDATE command is used to update the data of a particular row in a table.
  - The syntax:
  - `UPDATE table_name SET column_name = new_value WHERE some_condition;`
  - The `WHERE` clause is used to add a condition to any SQL query.
  - To update values in the Student table for a student with roll number 102.
  - `UPDATE student SET age=15 WHERE rollNo=12;`
- If you do not use the WHERE clause, then the update query will update the age for all the columns of the student table to 16.



```
mysql> update student set age=15 where rollNo=12;
Query OK, 1 row affected (0.05 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

Note – Where clause will be discussed more in the next Sprint.

Fetching Data from the Tables



```
mysql> Select * from student;
+--------+------+-------------+------+
| rollNo | name | total_marks | age  |
+--------+------+-------------+------+
|     12 | John |         456 |   15 |
|     14 | Anny |         456 |   13 |
+--------+------+-------------+------+
2 rows in set (0.00 sec)
```

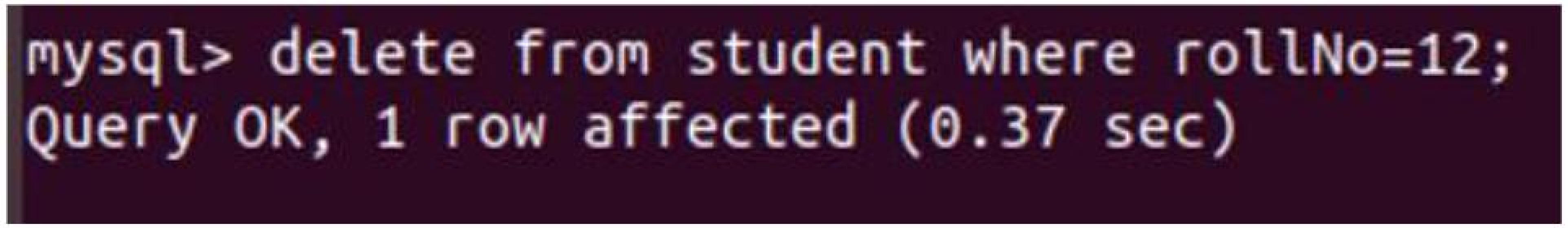
- After inserting values into a table, the data can be viewed through the select command.
- Thus, you can verify if the data is inserted appropriately into the table using a select query.

```
SELECT * from <table_name>;
```

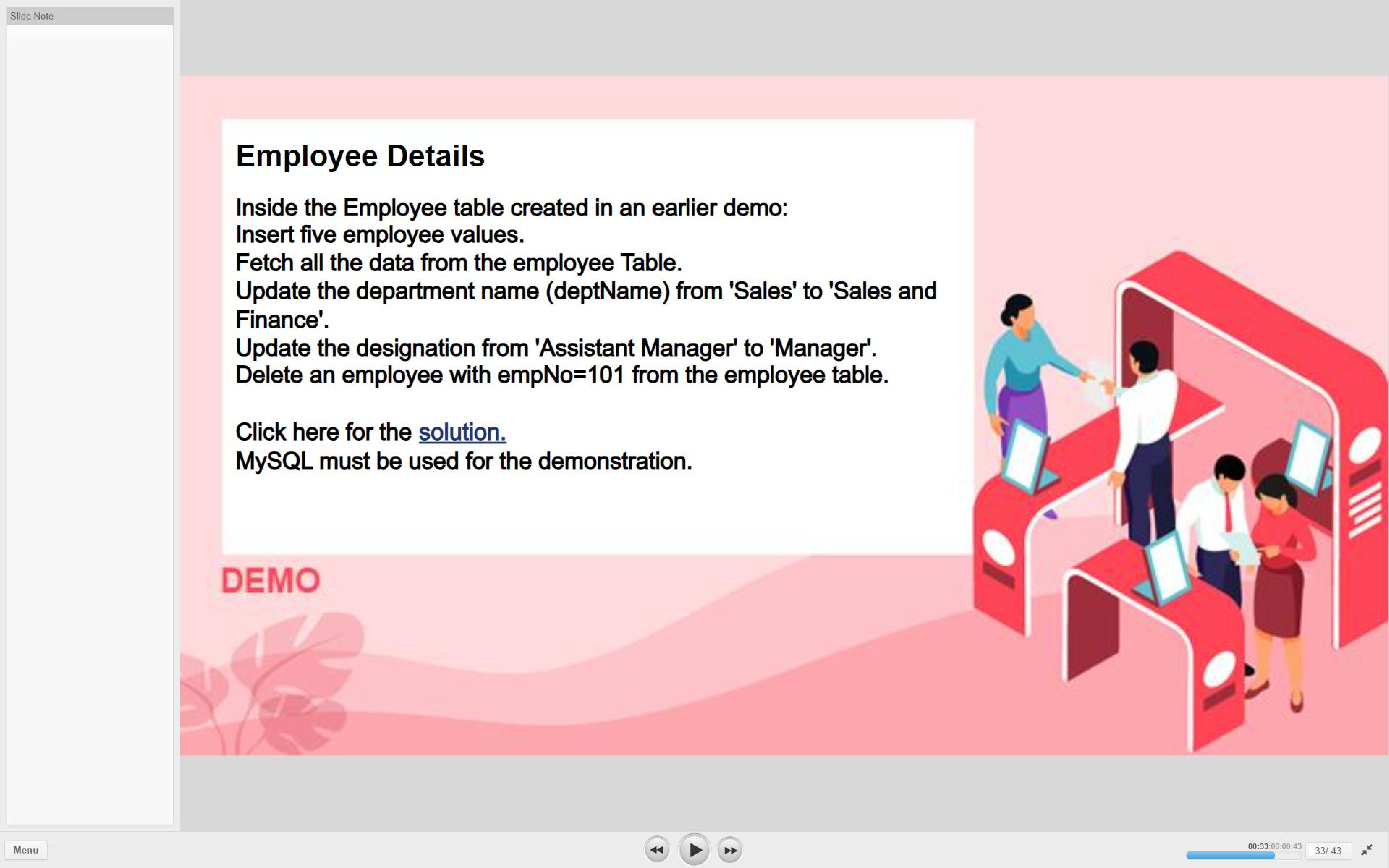
Note – Select Query will be discussed in more detail in the next Sprint.

DELETE Command

- The DELETE command is used to delete all rows from a table or a specific row from the table.
- The syntax:
  - `DELETE FROM table_name;`
- To delete all rows from the Student table, use the below command:
  - `DELETE FROM student;`
- To delete a specific row from the Student table, use the `WHERE` clause:
  - `DELETE FROM student WHERE rollNo=12;`



```
mysql> delete from student where rollNo=12;
Query OK, 1 row affected (0.37 sec)
```

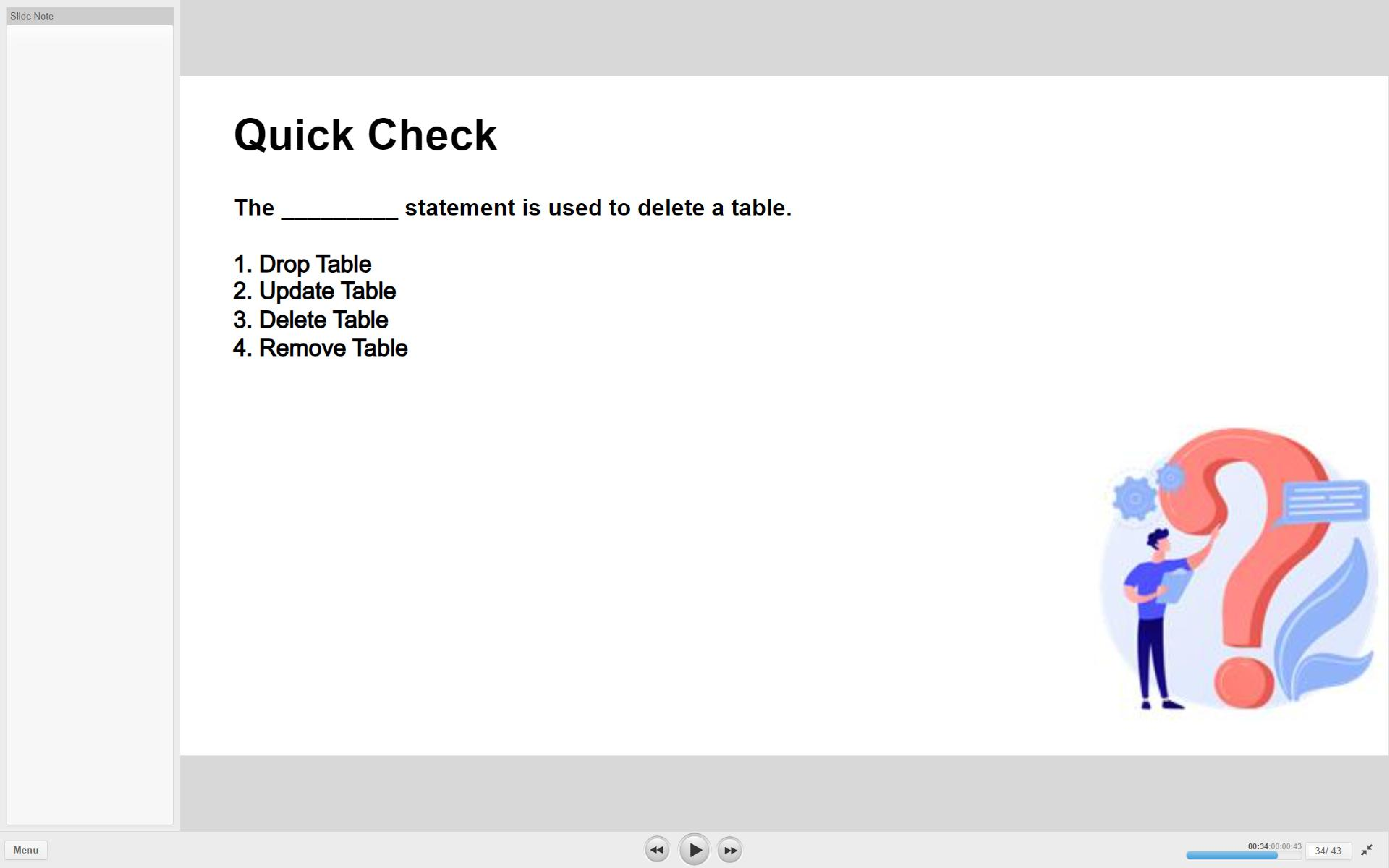


Employee Details

Inside the Employee table created in an earlier demo:
Insert five employee values.
Fetch all the data from the employee Table.
Update the department name (deptName) from 'Sales' to 'Sales and Finance'.
Update the designation from 'Assistant Manager' to 'Manager'.
Delete an employee with empNo=101 from the employee table.

Click here for the solution.
MySQL must be used for the demonstration.

DEMO



Quick Check

The _________ statement is used to delete a table.

1. Drop Table
2. Update Table
3. Delete Table
4. Remove Table

Quick Check: Solution

The _________ statement is used to delete a table.

1. Drop Table
2. Update Table
3. Delete Table
4. Remove Table

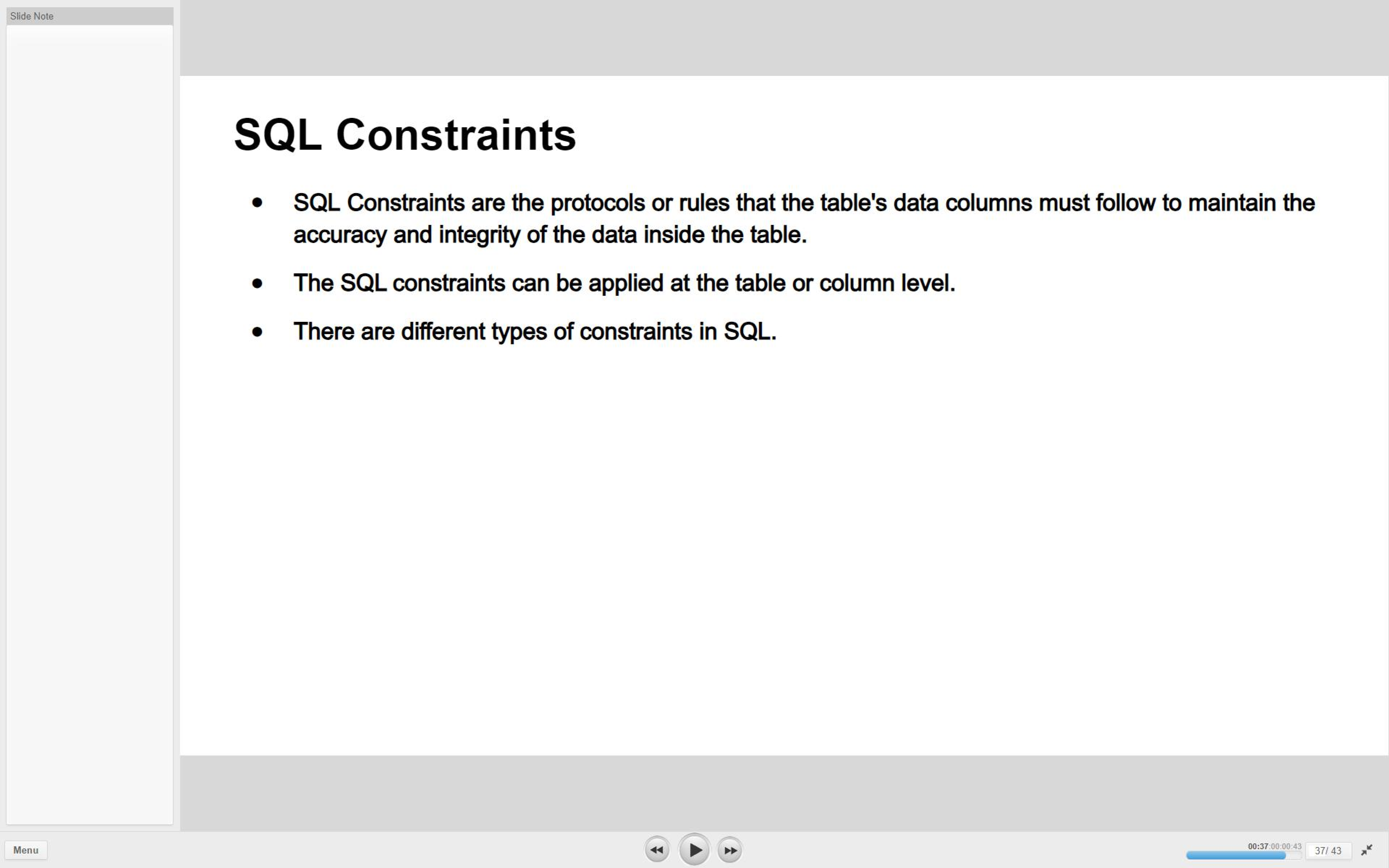
Slide Note

Define Constraints

Menu

00:36 00:00:43

36/ 43

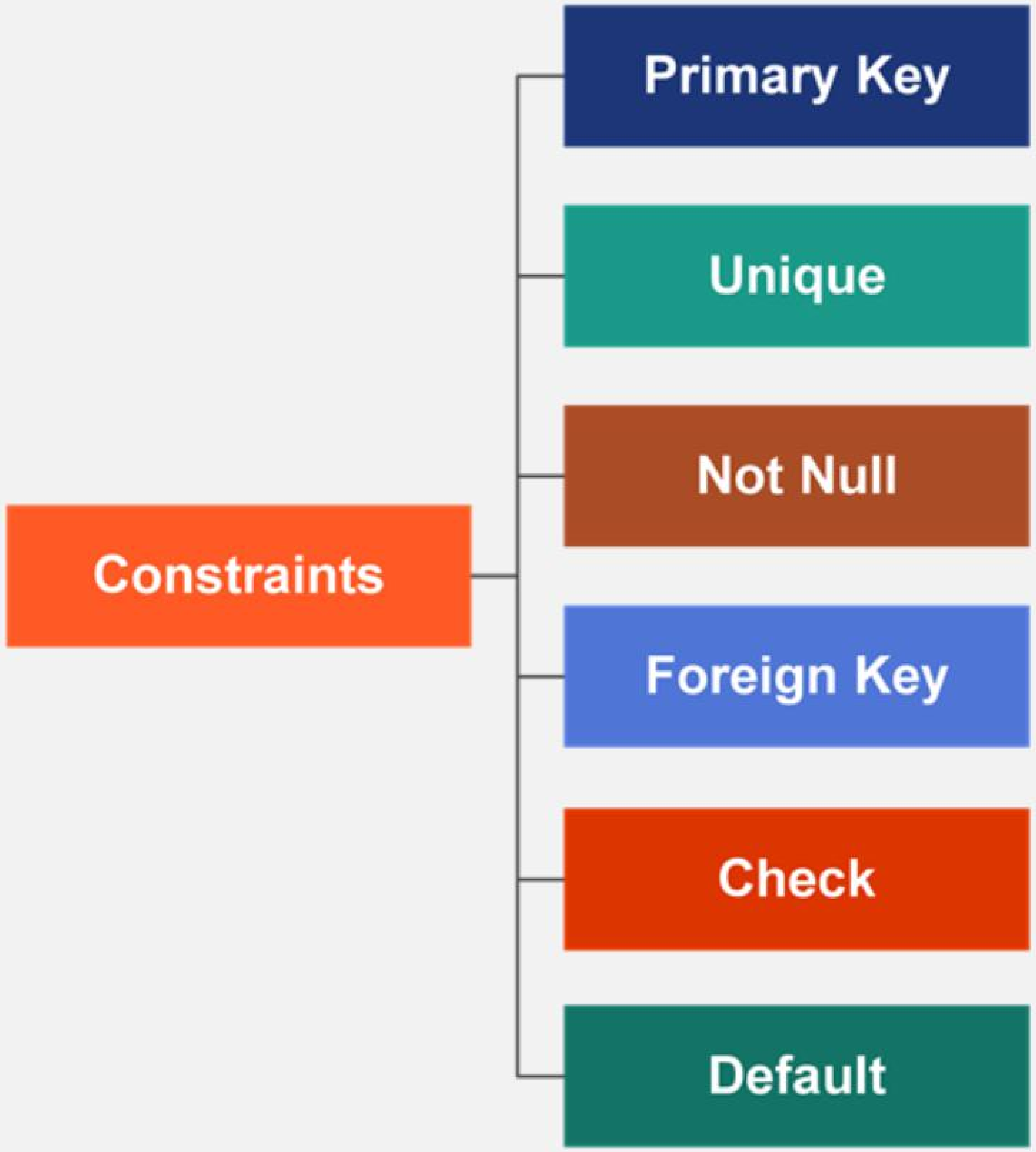


SQL Constraints

- SQL Constraints are the protocols or rules that the table's data columns must follow to maintain the accuracy and integrity of the data inside the table.
- The SQL constraints can be applied at the table or column level.
- There are different types of constraints in SQL.

Types of SQL Constraints

- There are six different types of SQL constraints.
- Note – We will only discuss about Primary and Foreign Key Constraints only



Constraints

Primary Key

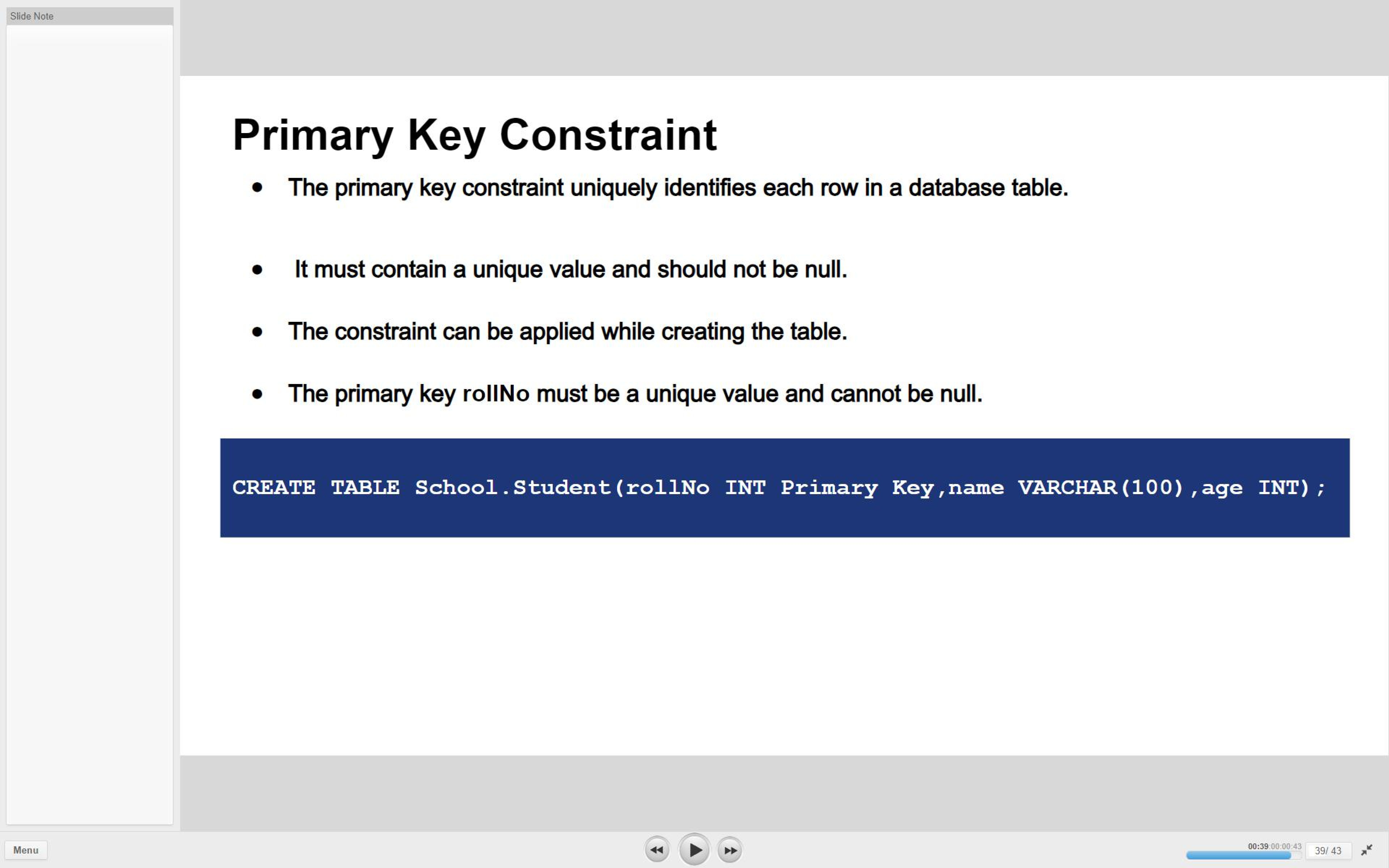
Unique

Not Null

Foreign Key

Check

Default



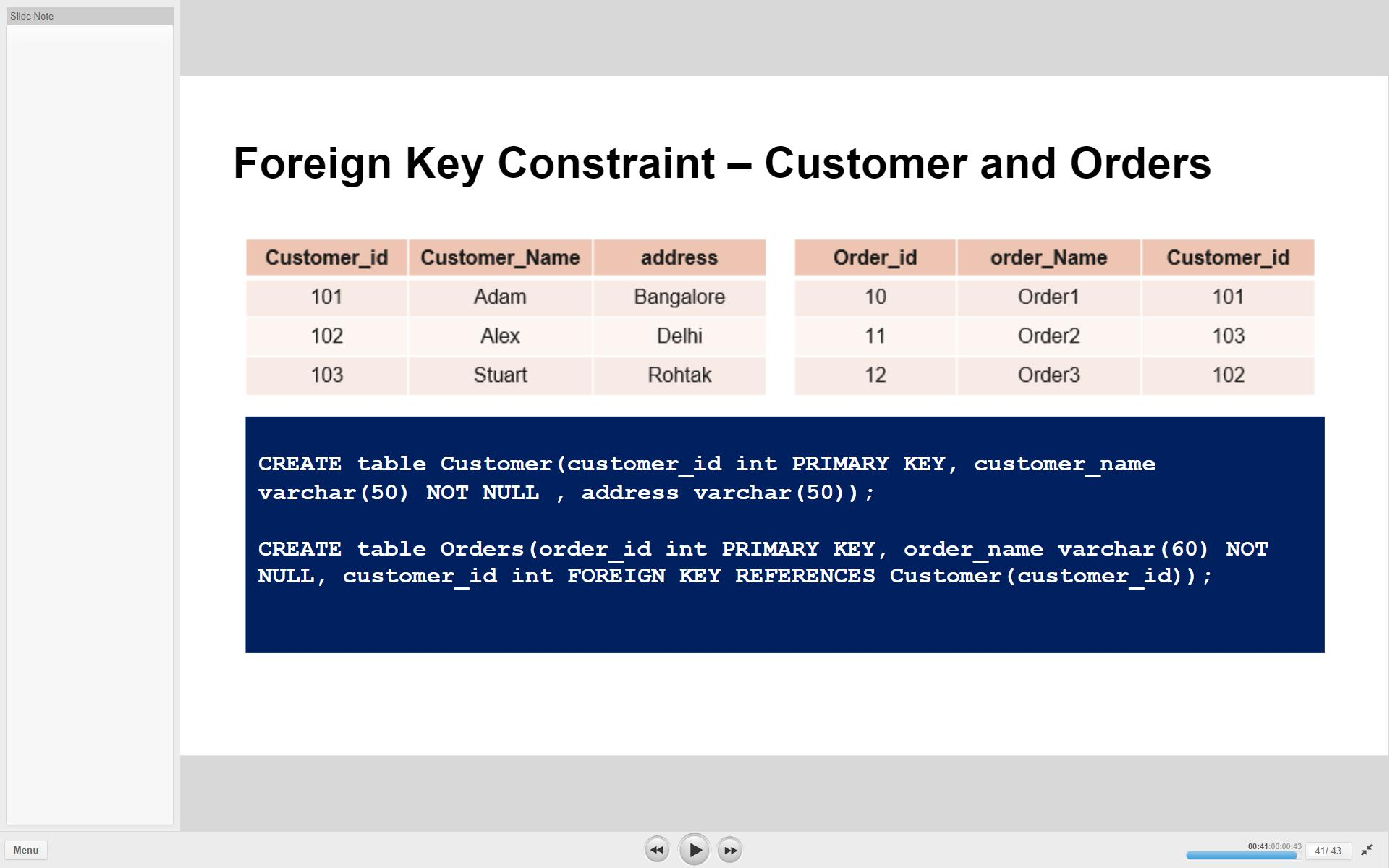
Primary Key Constraint

- The primary key constraint uniquely identifies each row in a database table.
- It must contain a unique value and should not be null.
- The constraint can be applied while creating the table.
- The primary key rollNo must be a unique value and cannot be null.

```
CREATE TABLE School.Student(rollNo INT Primary Key,name VARCHAR(100),age INT);
```

Foreign Key Constraint

- A foreign key is a field in a table that refers to the primary key of another table. It links the two tables.
- The table with the foreign key is called the child table, and the table with the primary key is called the referenced or parent table.
- Consider the example below of an online purchase made by a customer.
- If a customer purchases a product, there must also be an entry for the product purchased in the orders table. It will ensure that the data is consistent in both tables. Hence, it would be best to refer to the customerID in the Orders table.



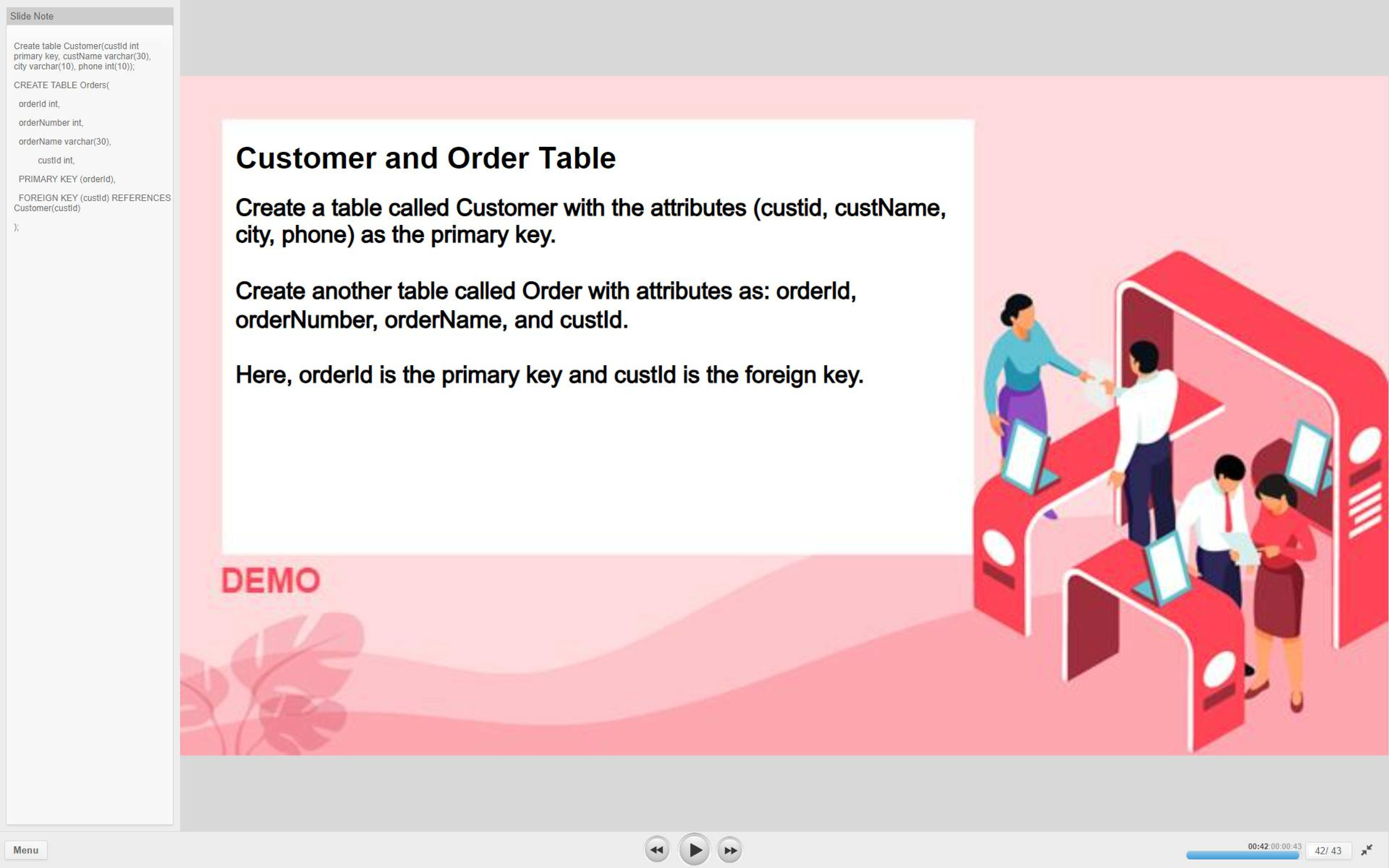
Foreign Key Constraint – Customer and Orders

Customer_id	Customer_Name	address
101	Adam	Bangalore
102	Alex	Delhi
103	Stuart	Rohtak

Order_id	order_Name	Customer_id
10	Order1	101
11	Order2	103
12	Order3	102

```
CREATE table Customer(customer_id int PRIMARY KEY, customer_name
varchar(50) NOT NULL , address varchar(50));

CREATE table Orders(order_id int PRIMARY KEY, order_name varchar(60) NOT
NULL, customer_id int FOREIGN KEY REFERENCES Customer(customer_id));
```



Slide Note

```
Create table Customer(custId int
primary key, custName varchar(30),
city varchar(10), phone int(10));

CREATE TABLE Orders(

  orderId int,

  orderNumber int,

  orderName varchar(30),

        custId int,

  PRIMARY KEY (orderId),

  FOREIGN KEY (custId) REFERENCES
Customer(custId)

);
```

Customer and Order Table

Create a table called Customer with the attributes (custid, custName, city, phone) as the primary key.

Create another table called Order with attributes as: orderId, orderNumber, orderName, and custId.

Here, orderId is the primary key and custId is the foreign key.

DEMO