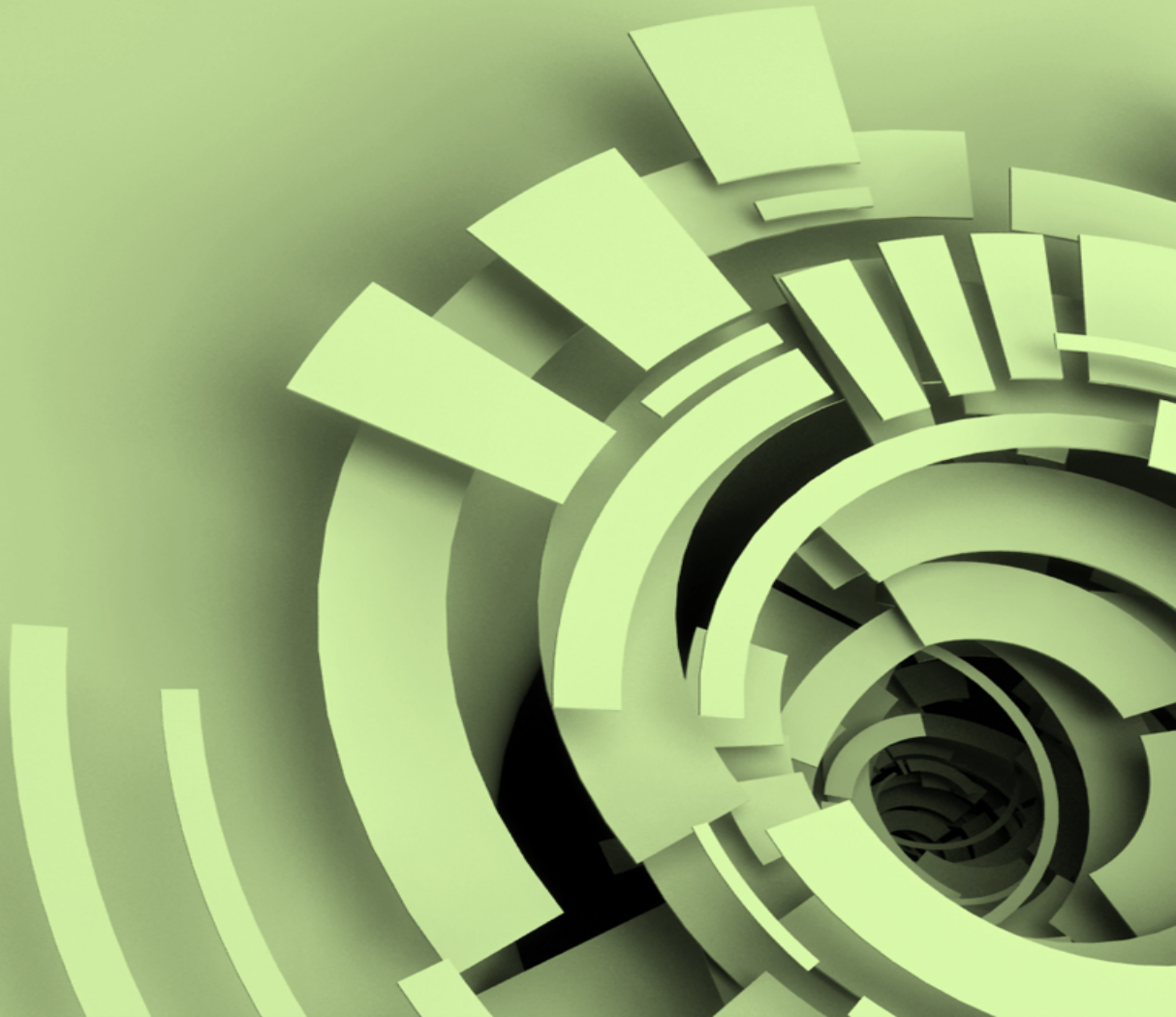
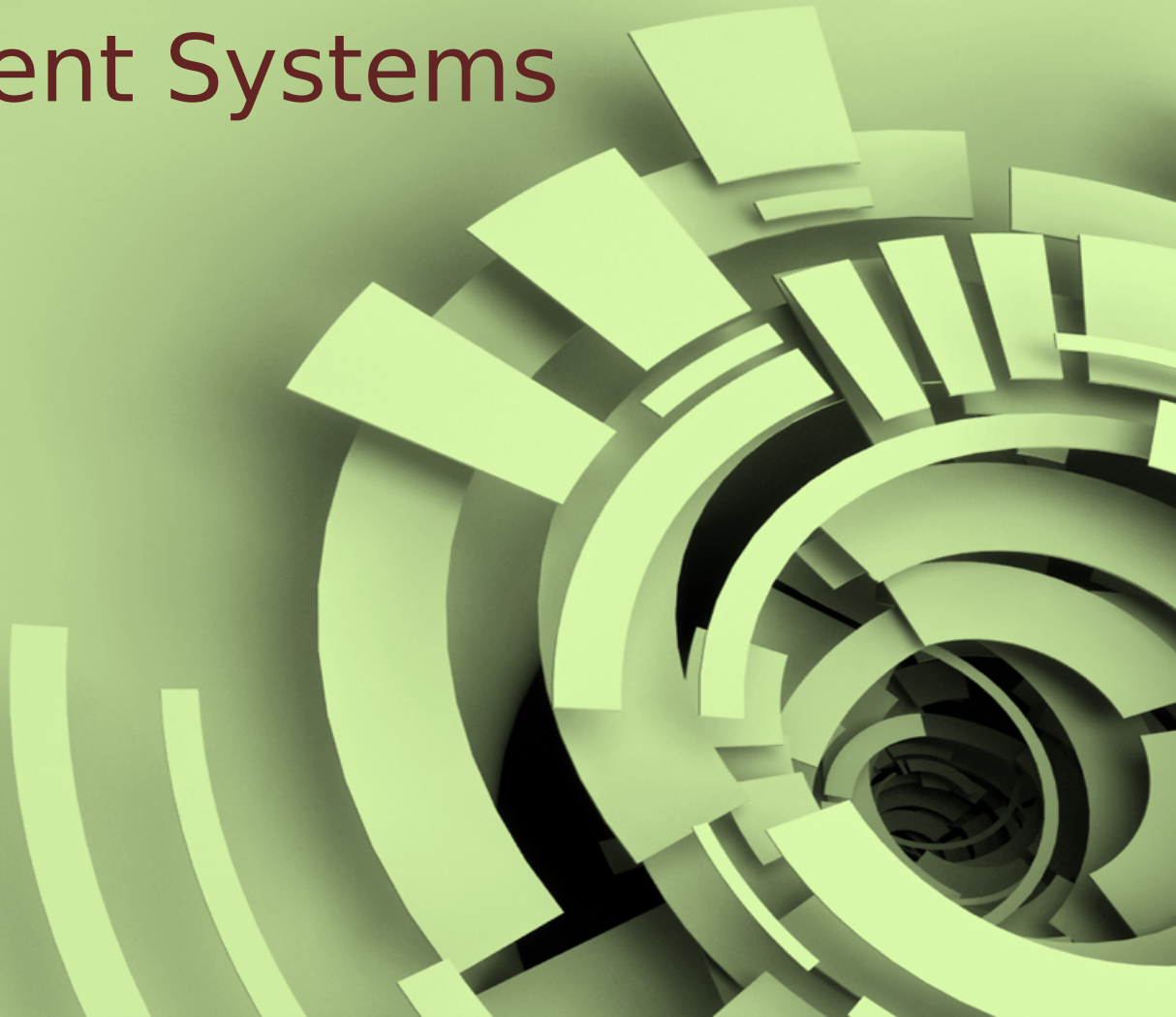
MySQL

Mr. Elvin B. Catantan



Outline

- Database Concepts
- Database Management Systems (DBMS)
- SQL in Action

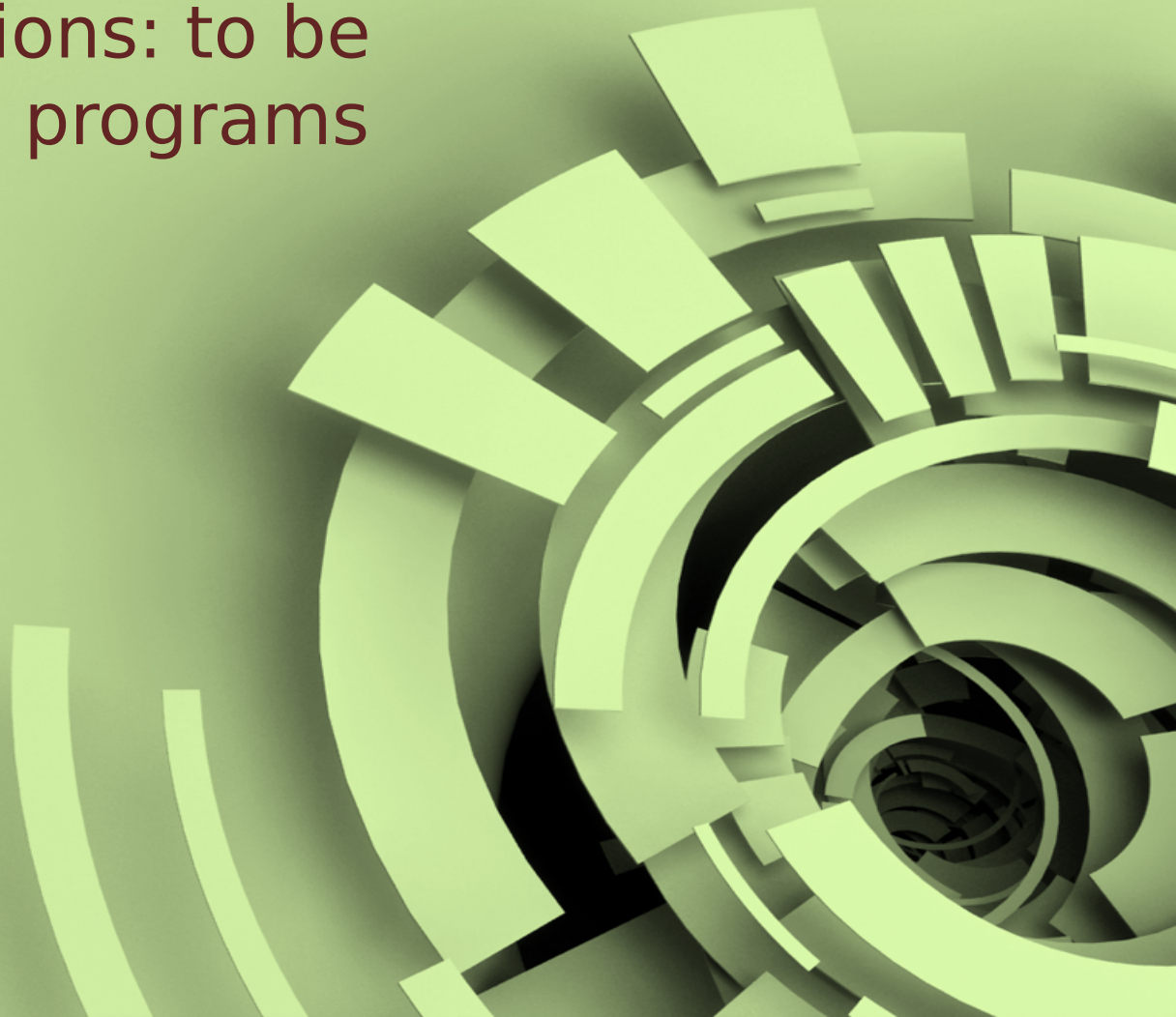


Important Functions on a Database

- Structure definition
- Population
- Querying
- Reporting
- Modification
- Modification of structure, of schema

DBMS Functions

- Database functions or DBMS functions
- Application-program functions: to be programmed in application programs
- Evolution
- Others
  - Concurrency control
  - Backup and recovery
  - Redundancy management
  - Access control
  - Performance optimization
  - Metadata management
  - Active features (rules, triggers)



Relational Database Model

-What is a relational database?

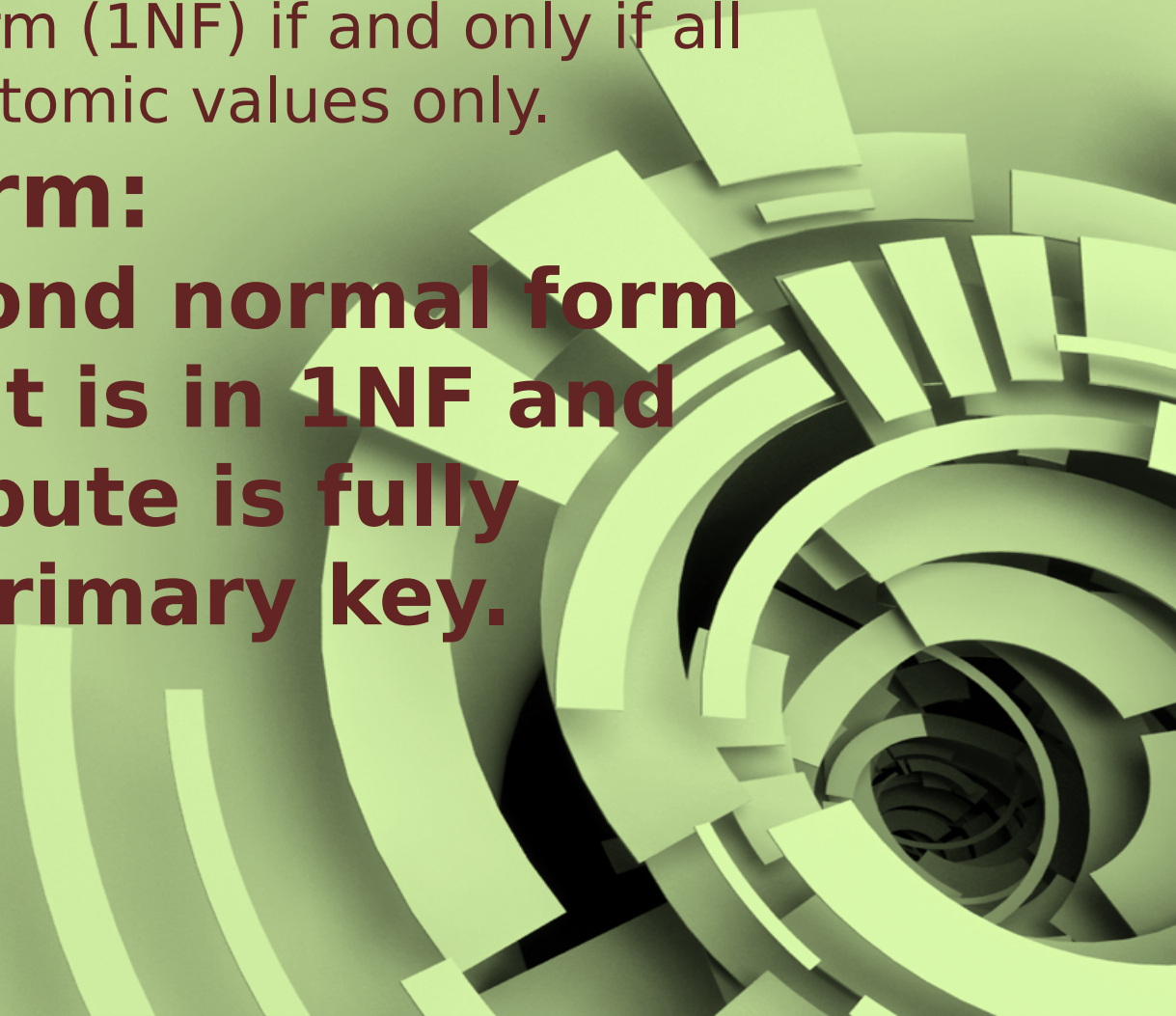
-a database that treats all of its data as a collection of relations

Relational Database Normal Forms

- **First Normal Form:**
  - **A relation is in first normal form (1NF) if and only if all underlying domains contain atomic values only.**

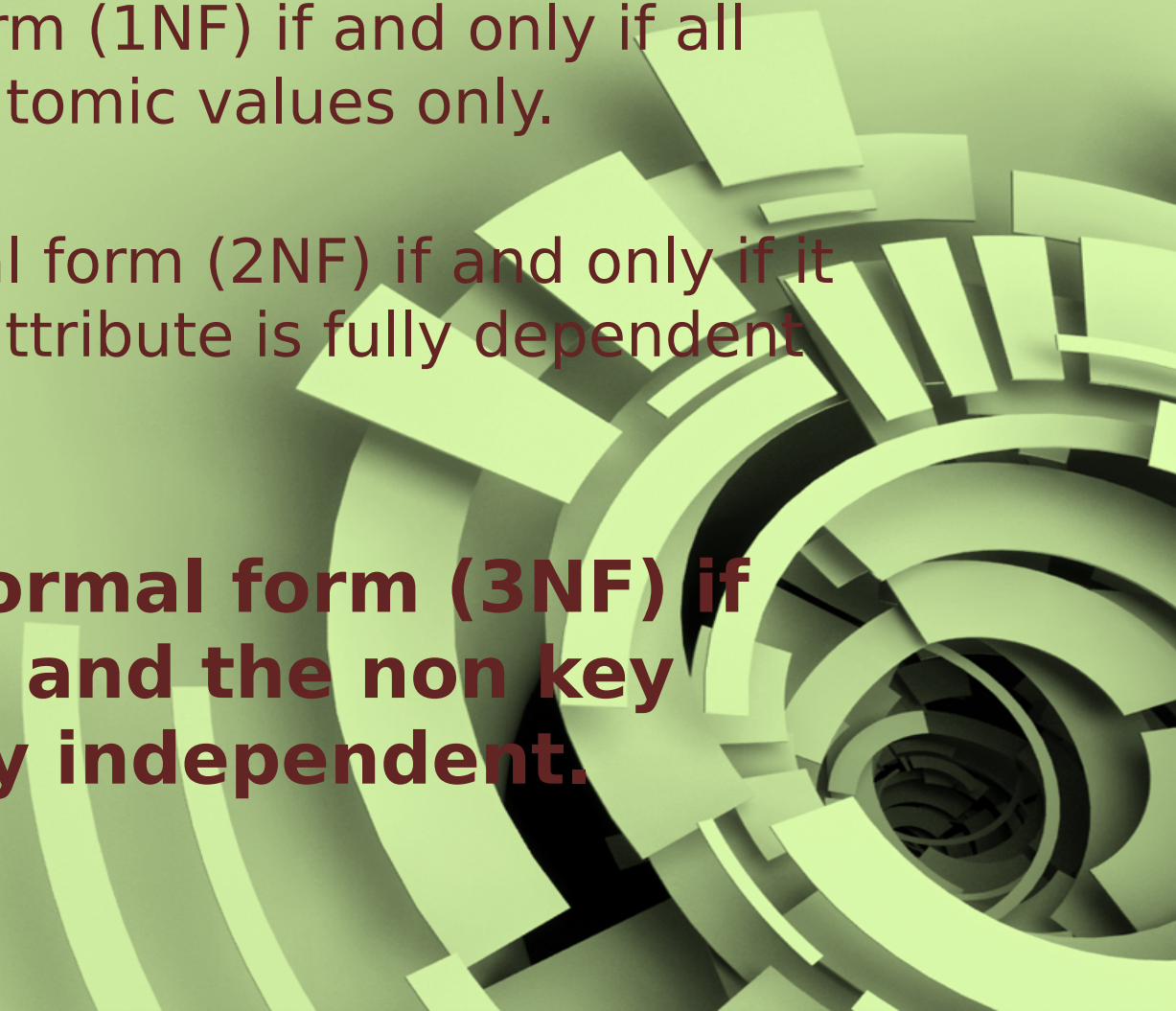
Relational Database Normal Forms

- First Normal Form:
  - A relation is in first normal form (1NF) if and only if all underlying domains contain atomic values only.
- **Second Normal Form:**
  - **A relation is in second normal form (2NF) if and only if it is in 1NF and every non-key attribute is fully dependent on the primary key.**



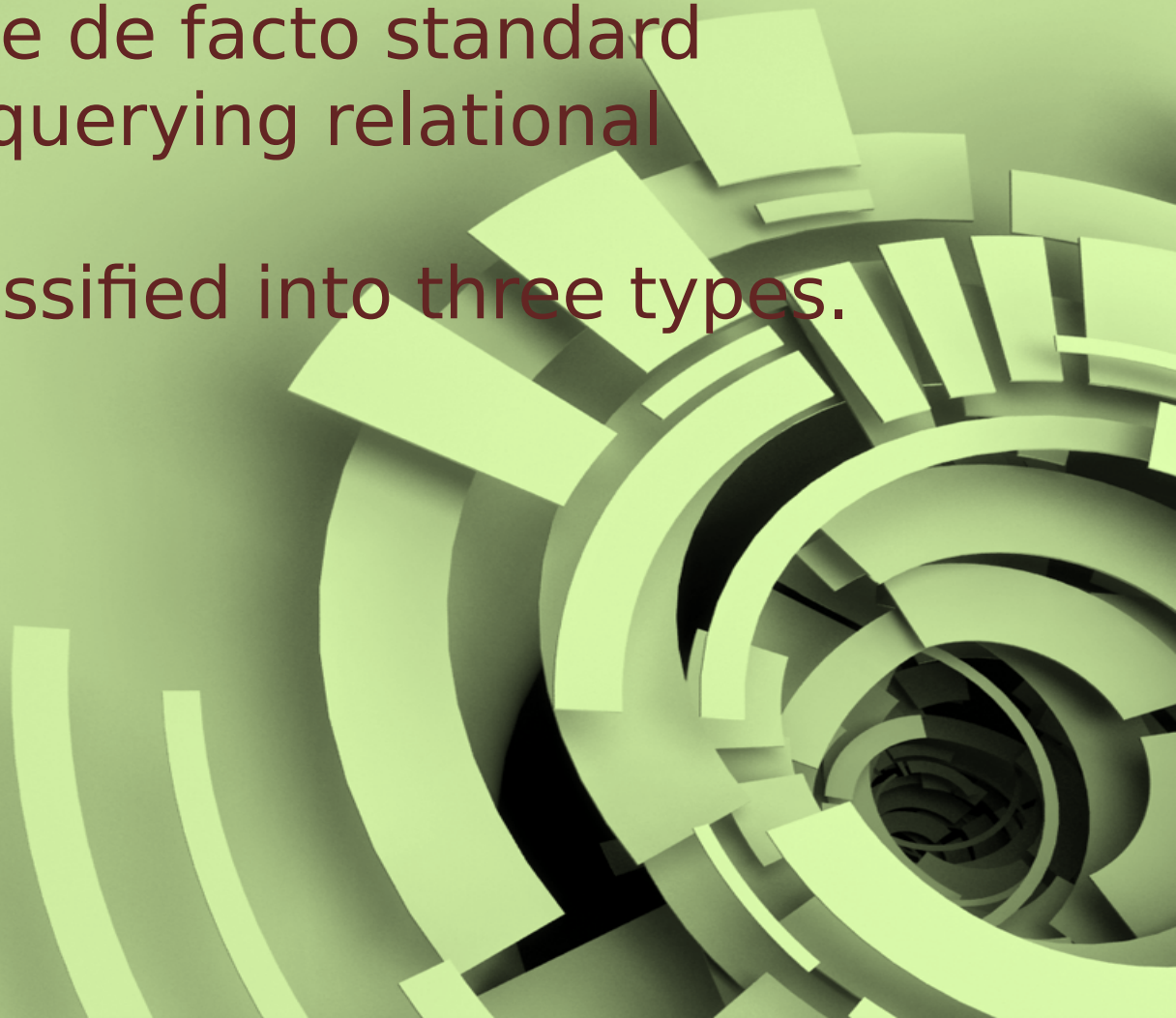
Relational Database Normal Forms

- First Normal Form:
  - A relation is in first normal form (1NF) if and only if all underlying domains contain atomic values only.
- Second Normal Form:
  - A relation is in second normal form (2NF) if and only if it is in 1NF and every non-key attribute is fully dependent on the primary key.
- **Third Normal Form:**
  - **A relation is in third normal form (3NF) if and only if it is in 2 NF and the non key attributes are mutually independent.**



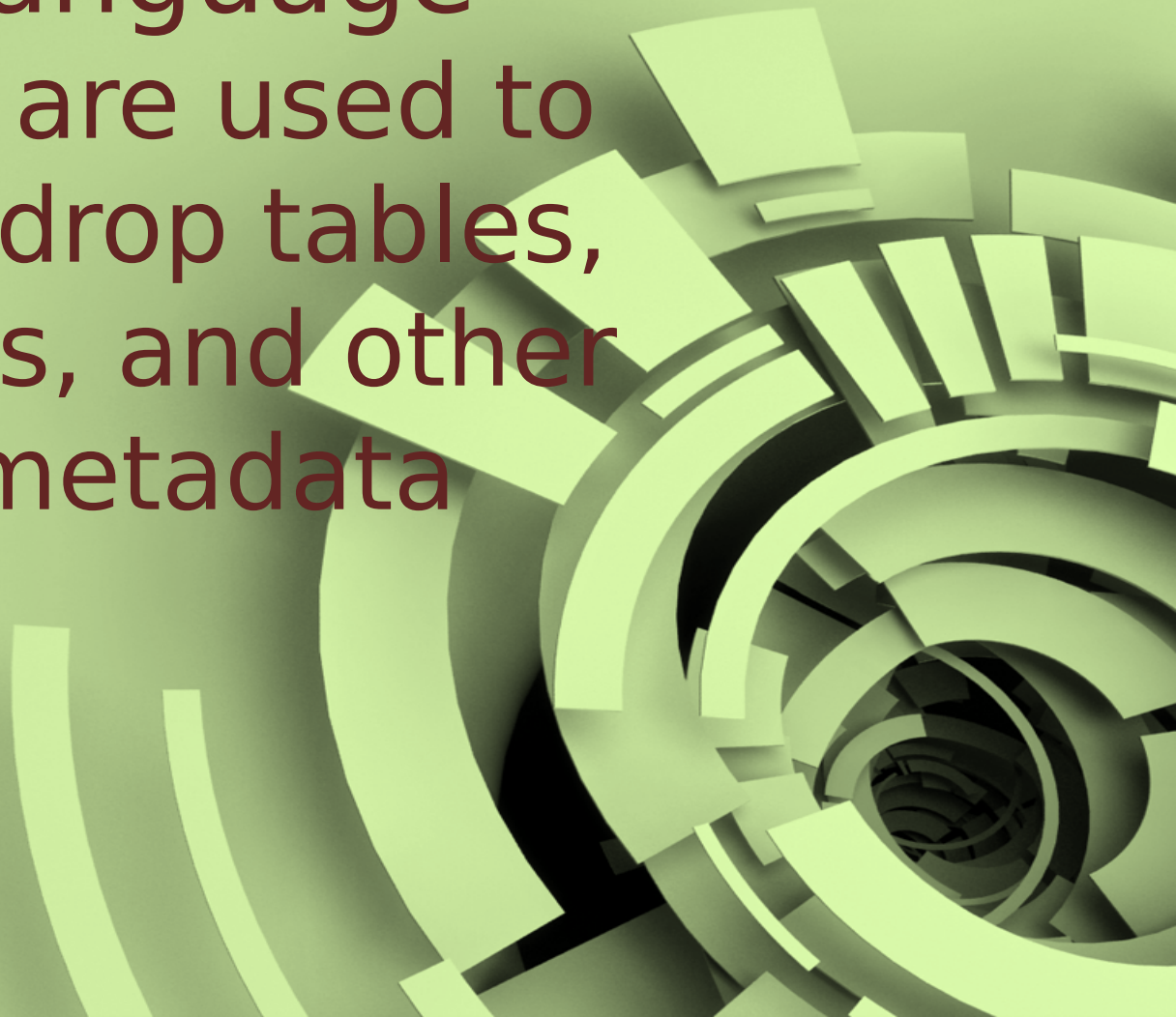
Structured Query Language (SQL)

- Pronounced “S-Q-L” by some and “sequel” by others, SQL has become the de facto standard language for creating and querying relational databases.
- SQL commands can be classified into three types.
  - Data Definition Language
  - Data Manipulation Language
  - Data Control Language



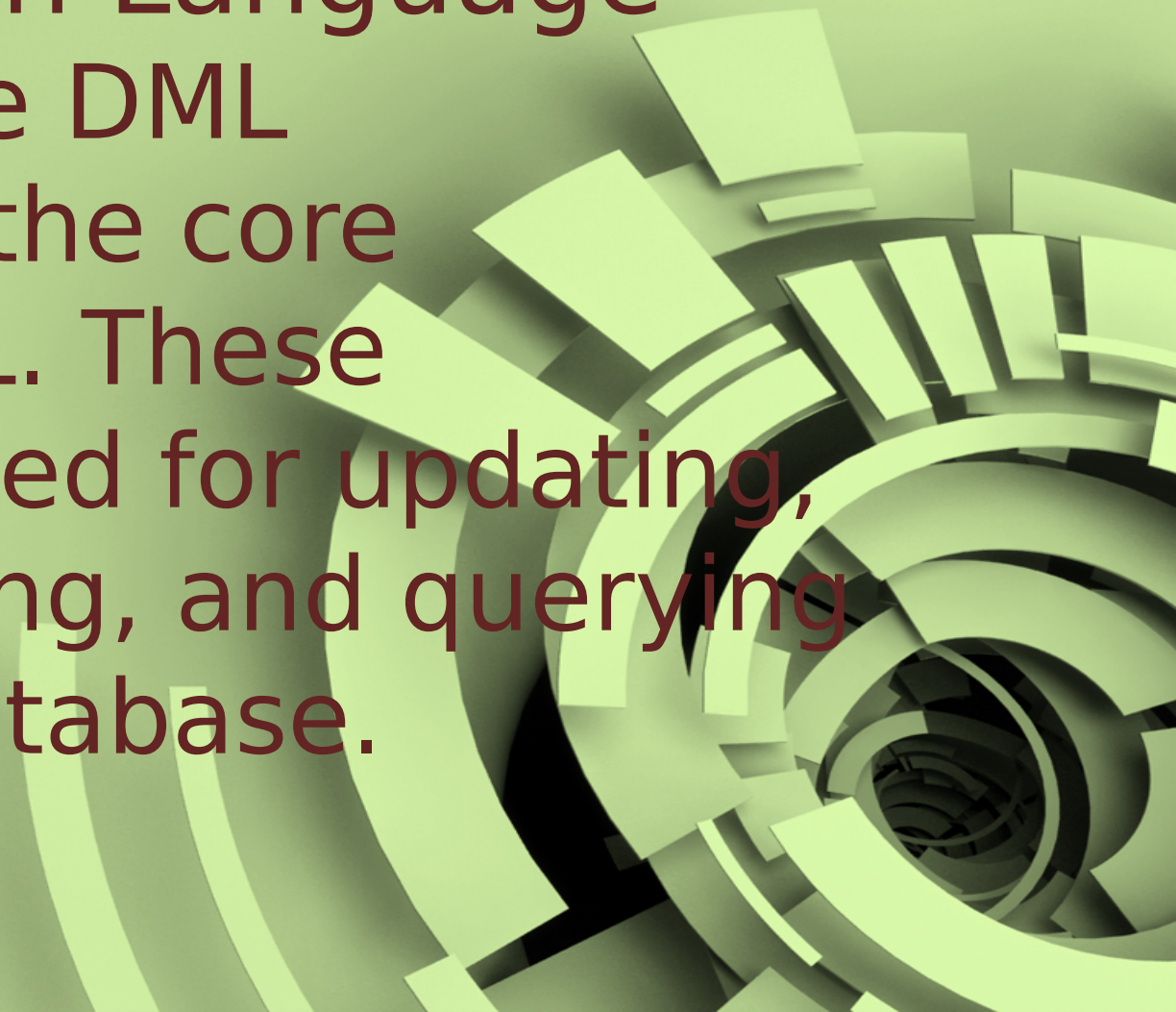
Structured Query Language (SQL)

- Data Definition Language
  - These commands are used to create, alter, and drop tables, views, and indexes, and other things related to metadata



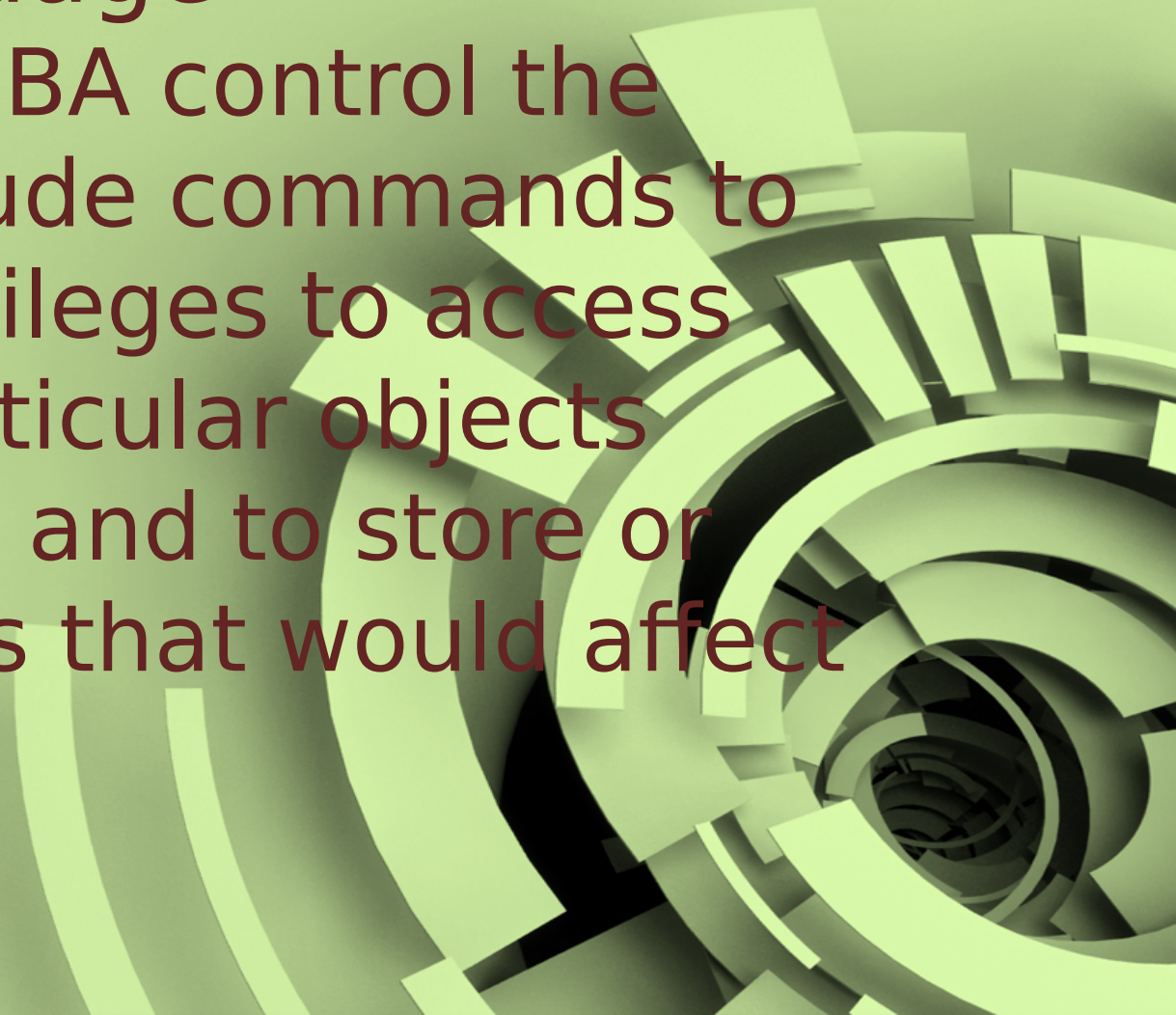
Structured Query Language (SQL)

- Data Manipulation Language
  - Many consider the DML commands to be the core commands of SQL. These commands are used for updating, inserting, modifying, and querying the data in the database.



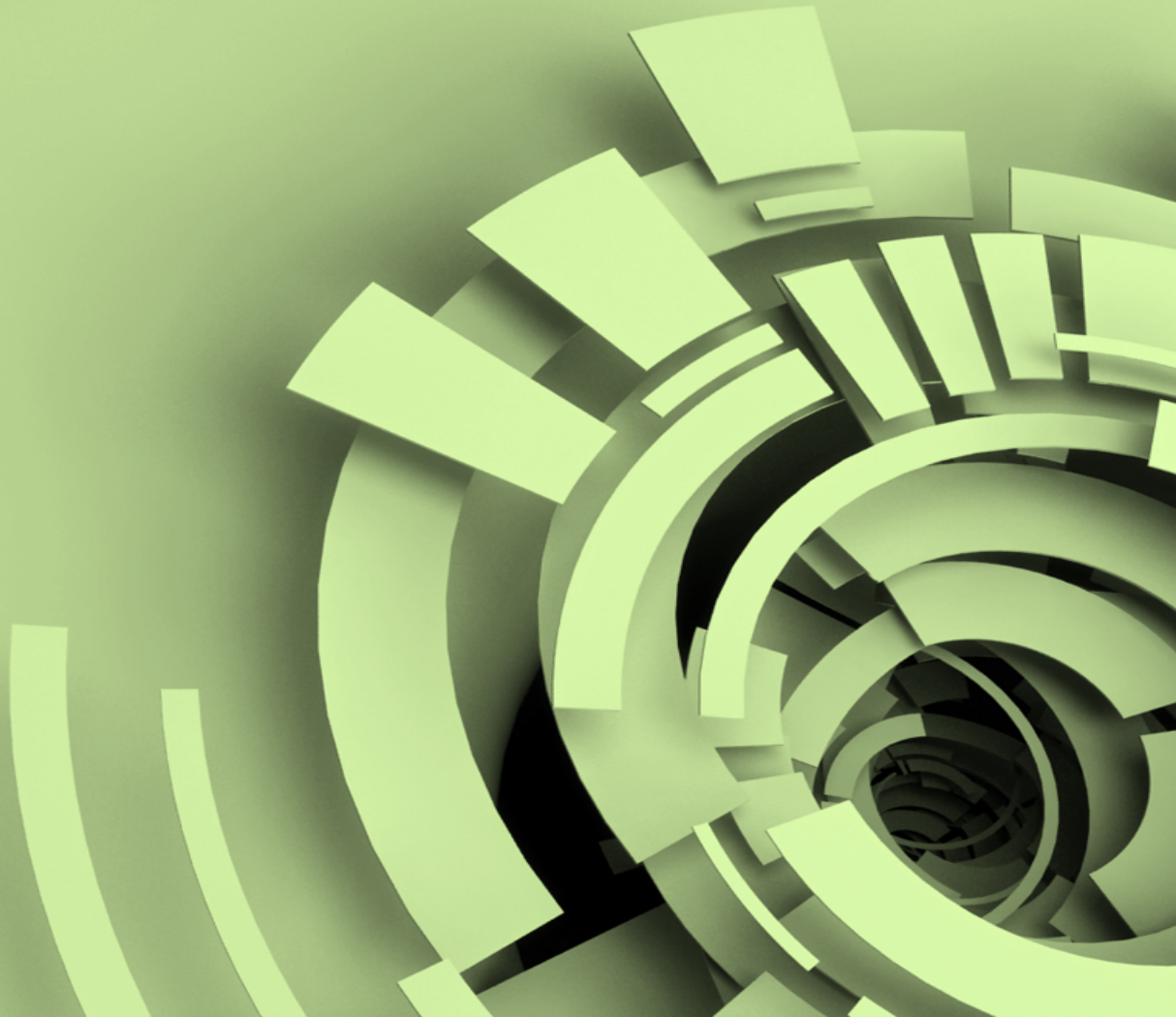
Structured Query Language (SQL)

- Data Control Language
  - commands help a DBA control the database; they include commands to grant or revoke privileges to access the database or particular objects within the database and to store or remove transactions that would affect the database.



SQL in Action

- Database Queries
- Table Queries
- Retrieving Data
- Inserting Data
- Modifying Data
- Pattern Matching
- Sorting
- Limiting
- Grouping

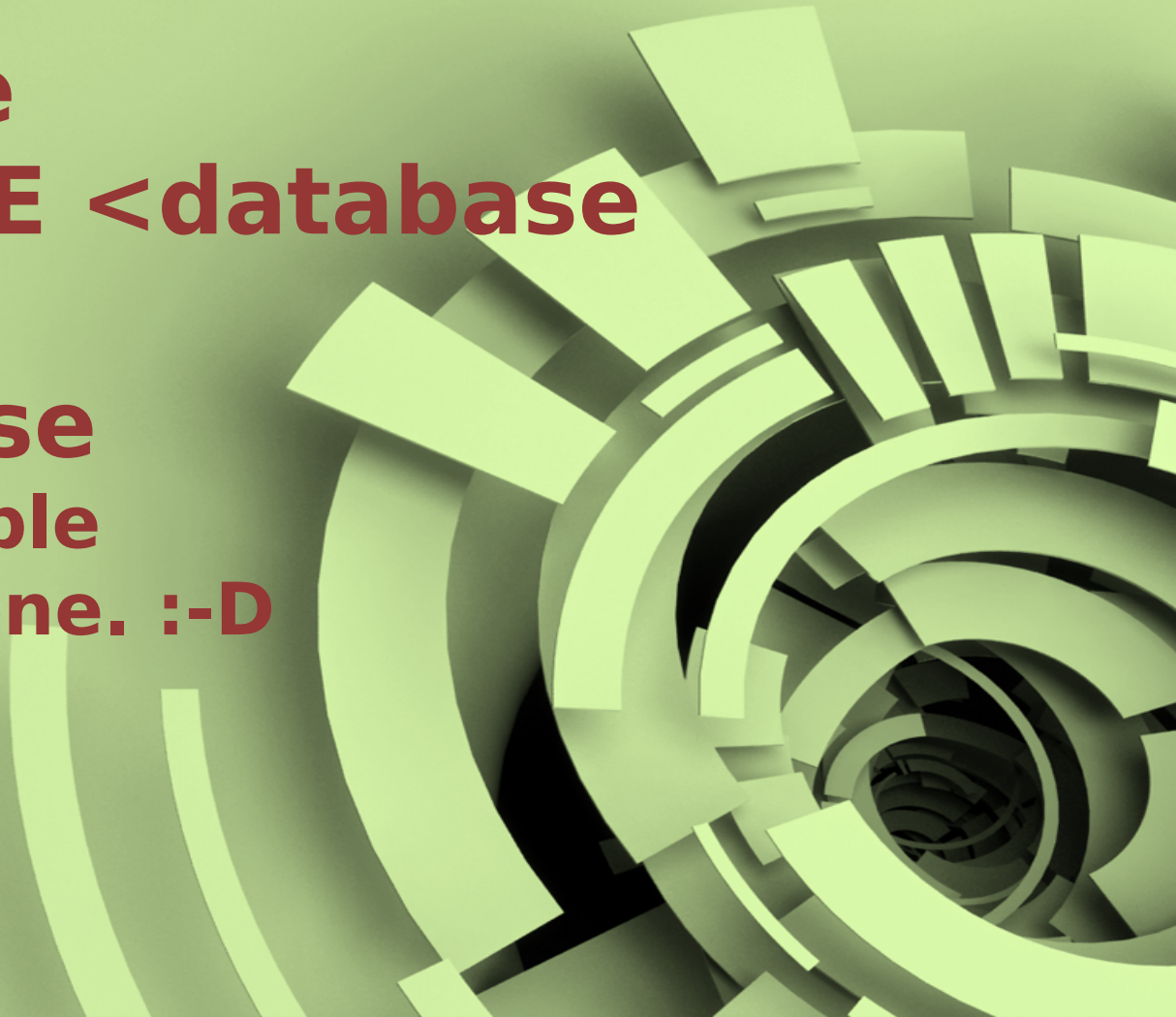


SQL in Action

- Database Queries
  - **List all databases**
    - **>SHOW databases;**
  - **Select the database**
    - **>USE <database name>**
  - **Create a database**
    - **>CREATE DATABASE <database name>**

SQL in Action

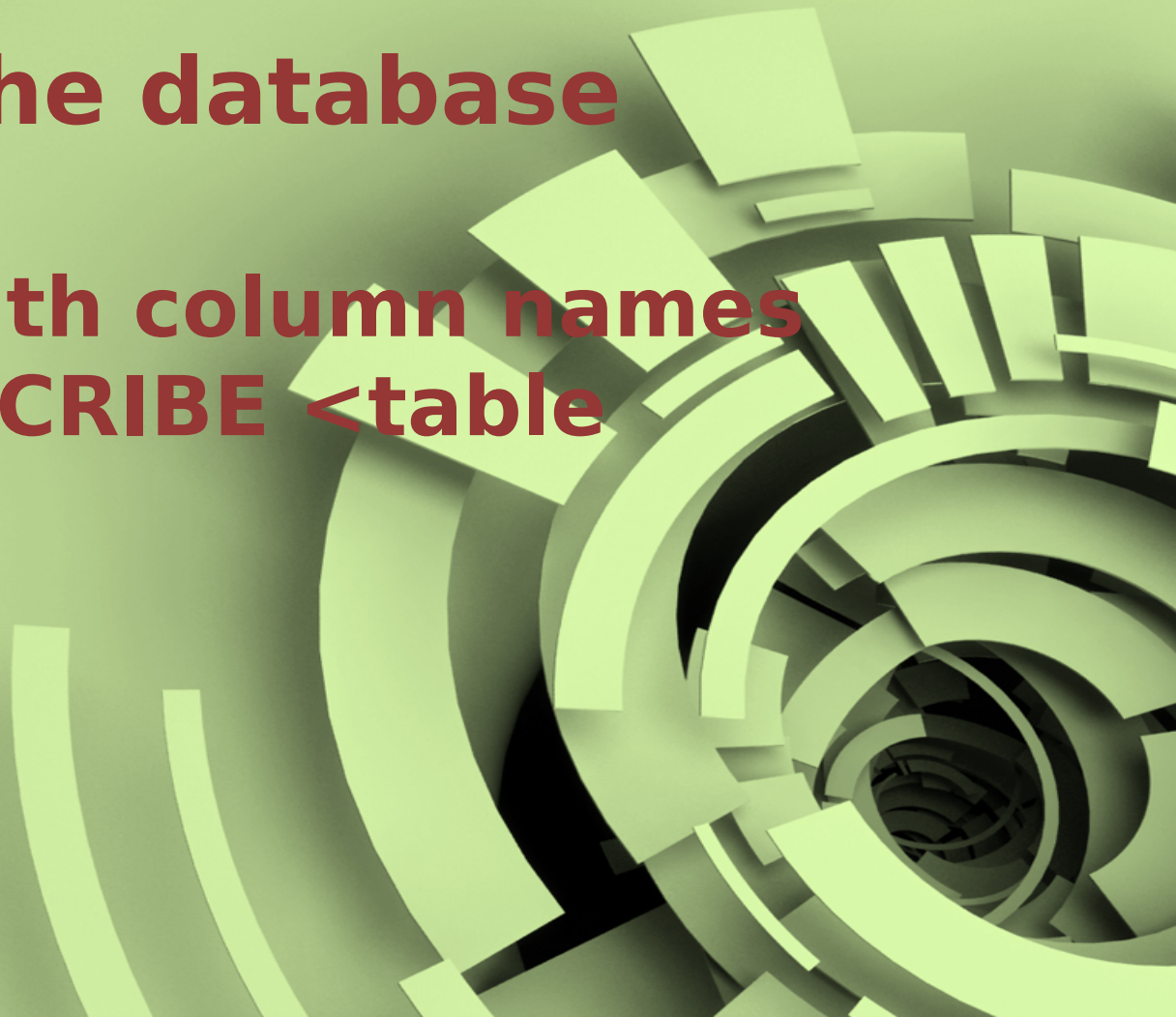
- Database Queries
  - **Delete a database**
    - **>DROP DATABASE <database name>**
  - **Rename a database**
    - **DROP the previous table then create a new one. :-D**



SQL in Action

- Table Queries
  - **List all tables in the database**
    >SHOW tables;

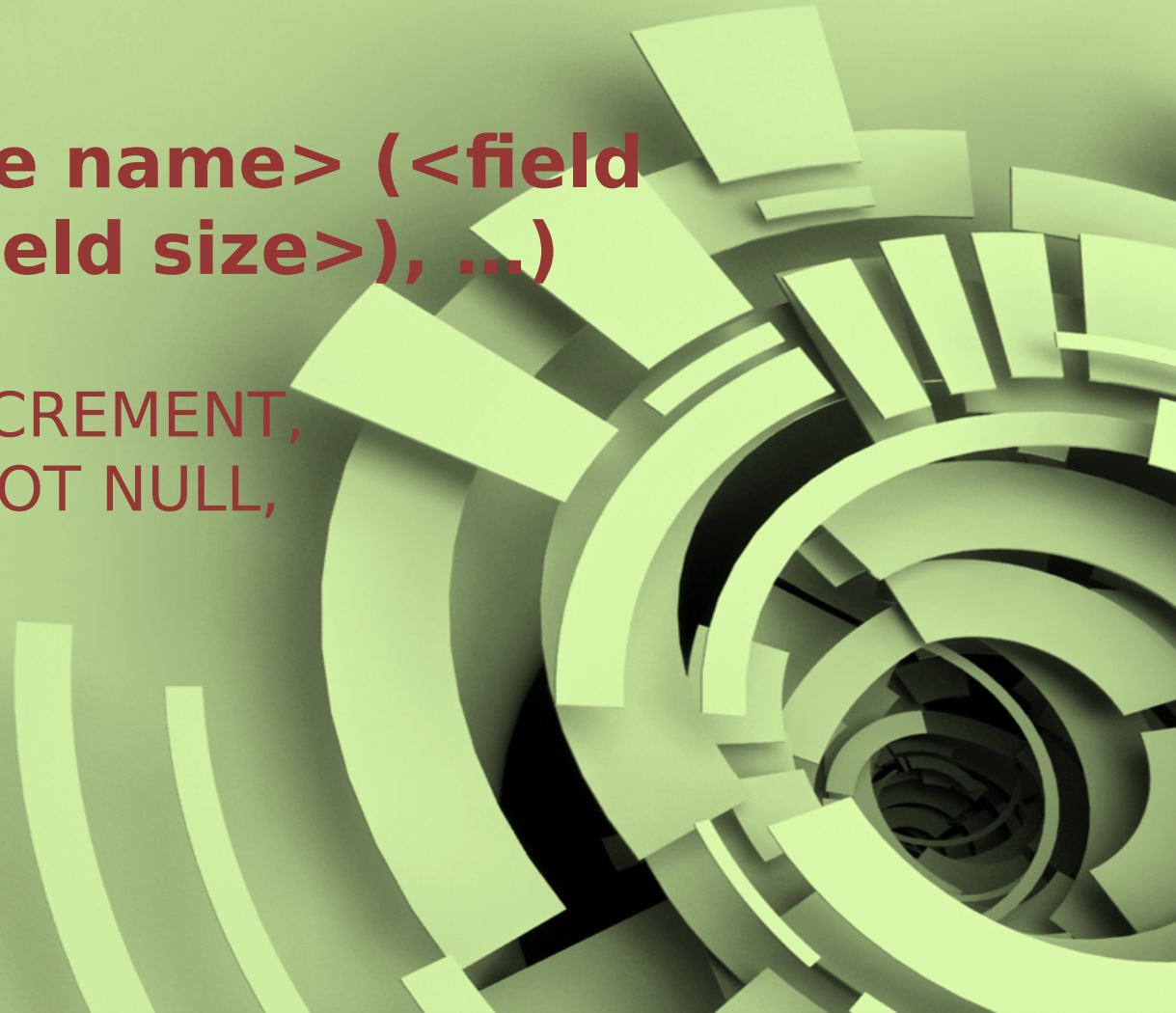
Show table format with column names and data types >DESCRIBE <table name>



SQL in Action

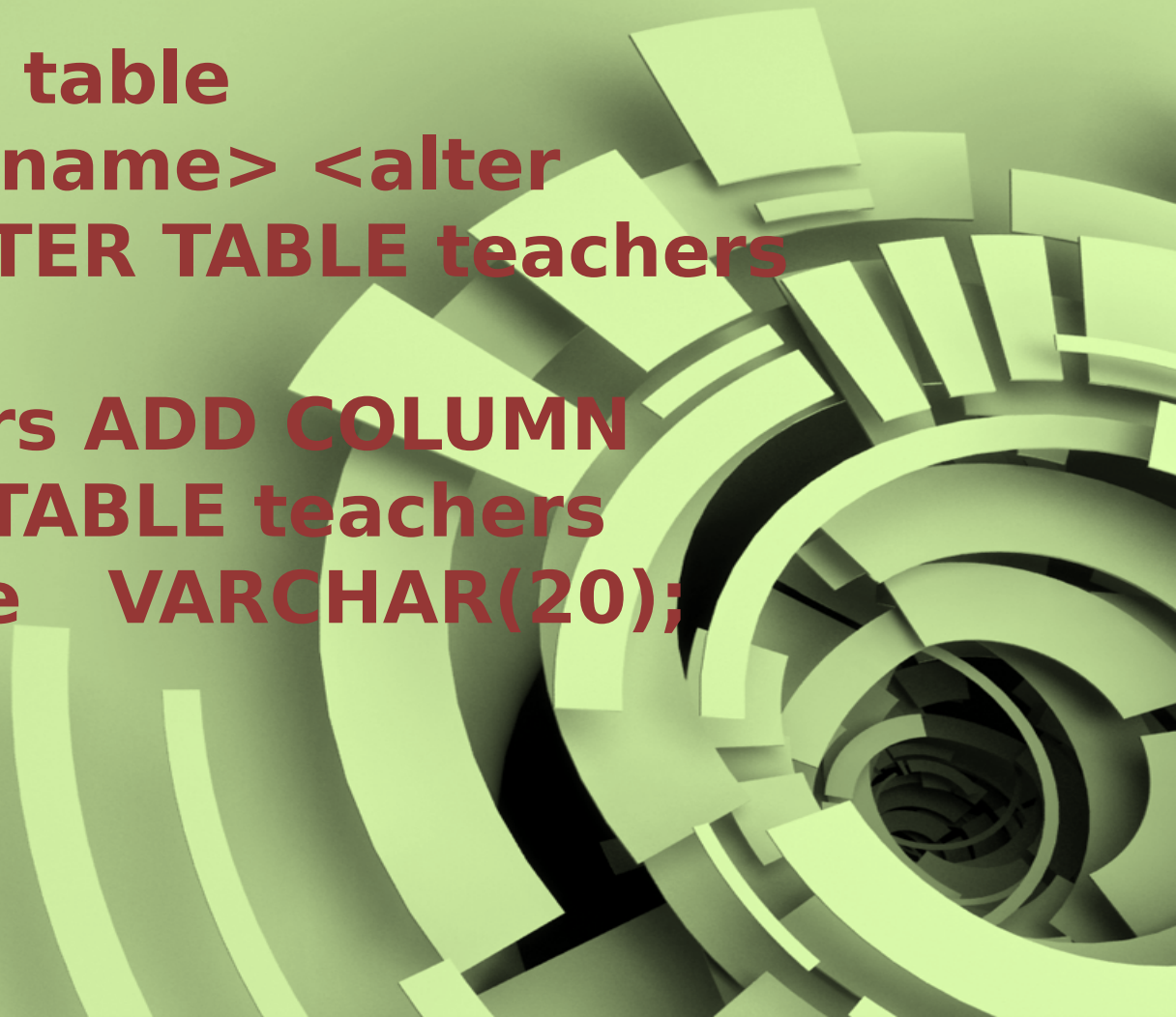
- Table Queries
  - **Create a table**
    >CREATE TABLE <table name> (<field name> <field type> (<field size>), ...)

```
CREATE TABLE Persons(
    ID int NOT NULL AUTO_INCREMENT,
    LastName varchar(255) NOT NULL,
    FirstName varchar(255),
    Address varchar(255),
    City varchar(255),
    PRIMARY KEY (ID)
    )
```



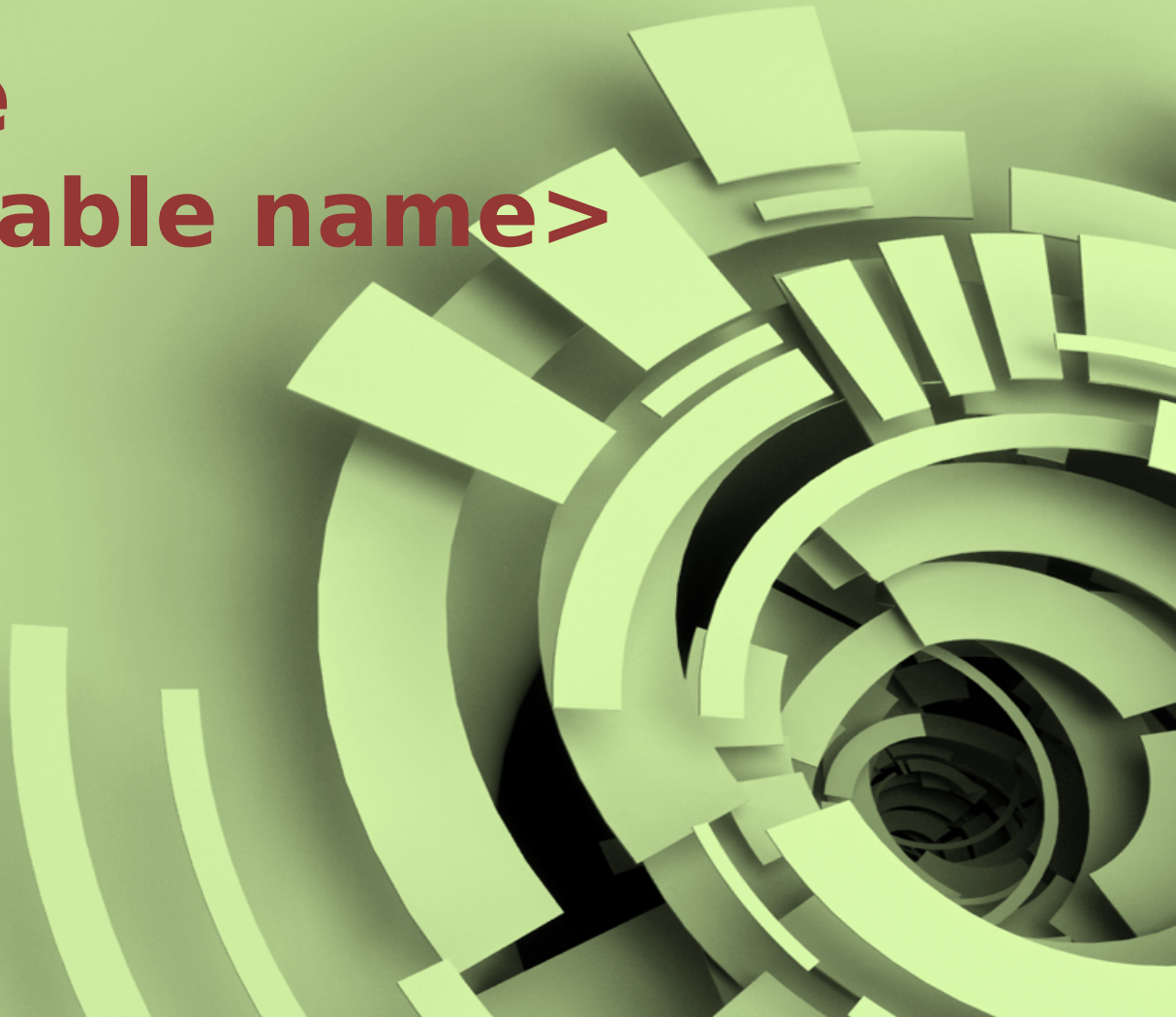
SQL in Action

- Table Queries
  - **Modify the structure of table**
    >ALTER TABLE <table name> <alter specifications>
    >ALTER TABLE teachers DROP COLUMN salary;
    >ALTER TABLE teachers ADD COLUMN salary INT(5);
    >ALTER TABLE teachers CHANGE firstName name VARCHAR(20);



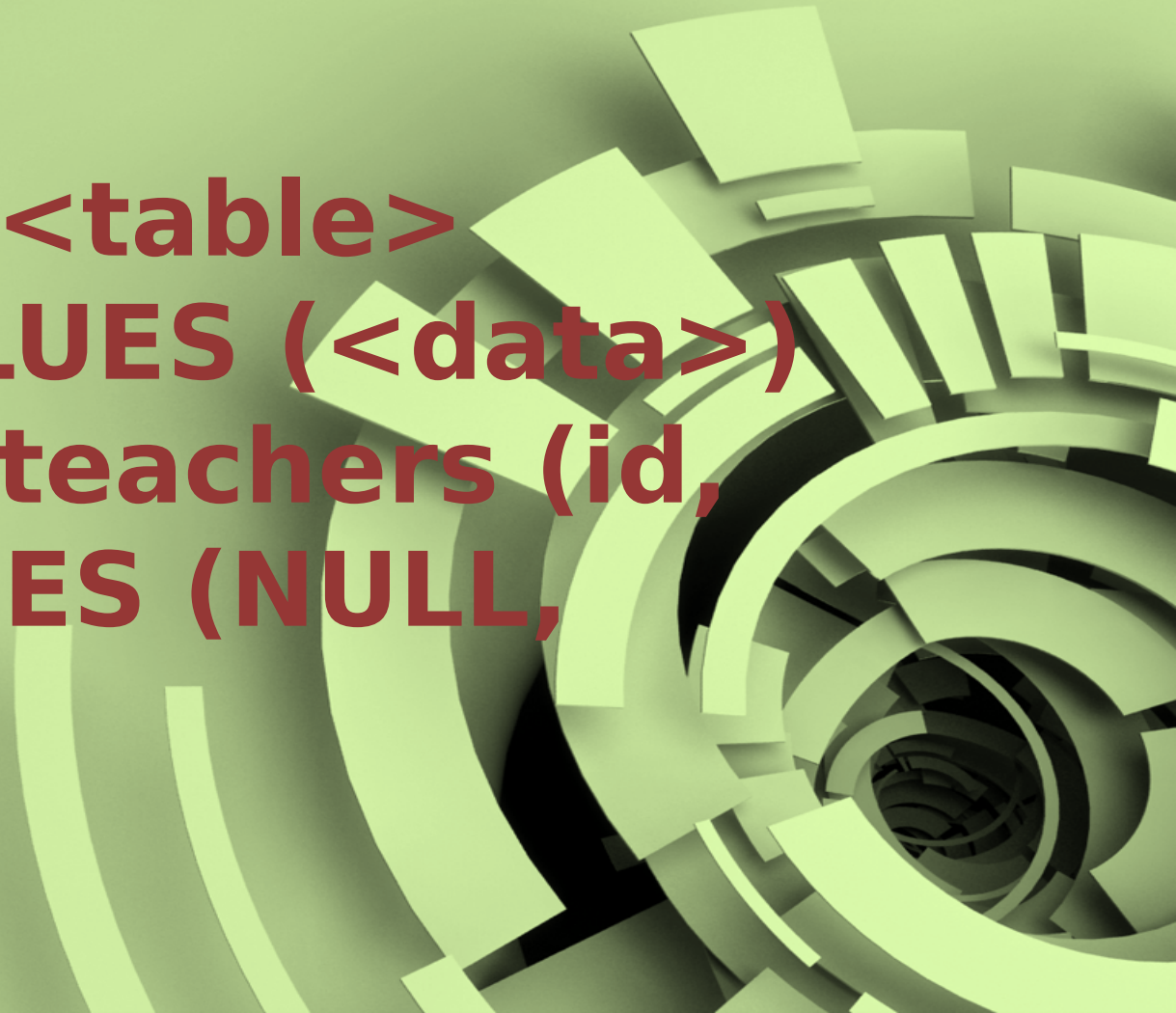
SQL in Action

- Table Queries
  - **Delete the table**
    - **>DROP TABLE <table name>**



SQL in Action

- Inserting Data
  - **Inserting Data:**
    > INSERT INTO <table> (<columns>) VALUES (<data>)
    > INSERT INTO teachers (id, name, age) VALUES (NULL, 'John', '12');

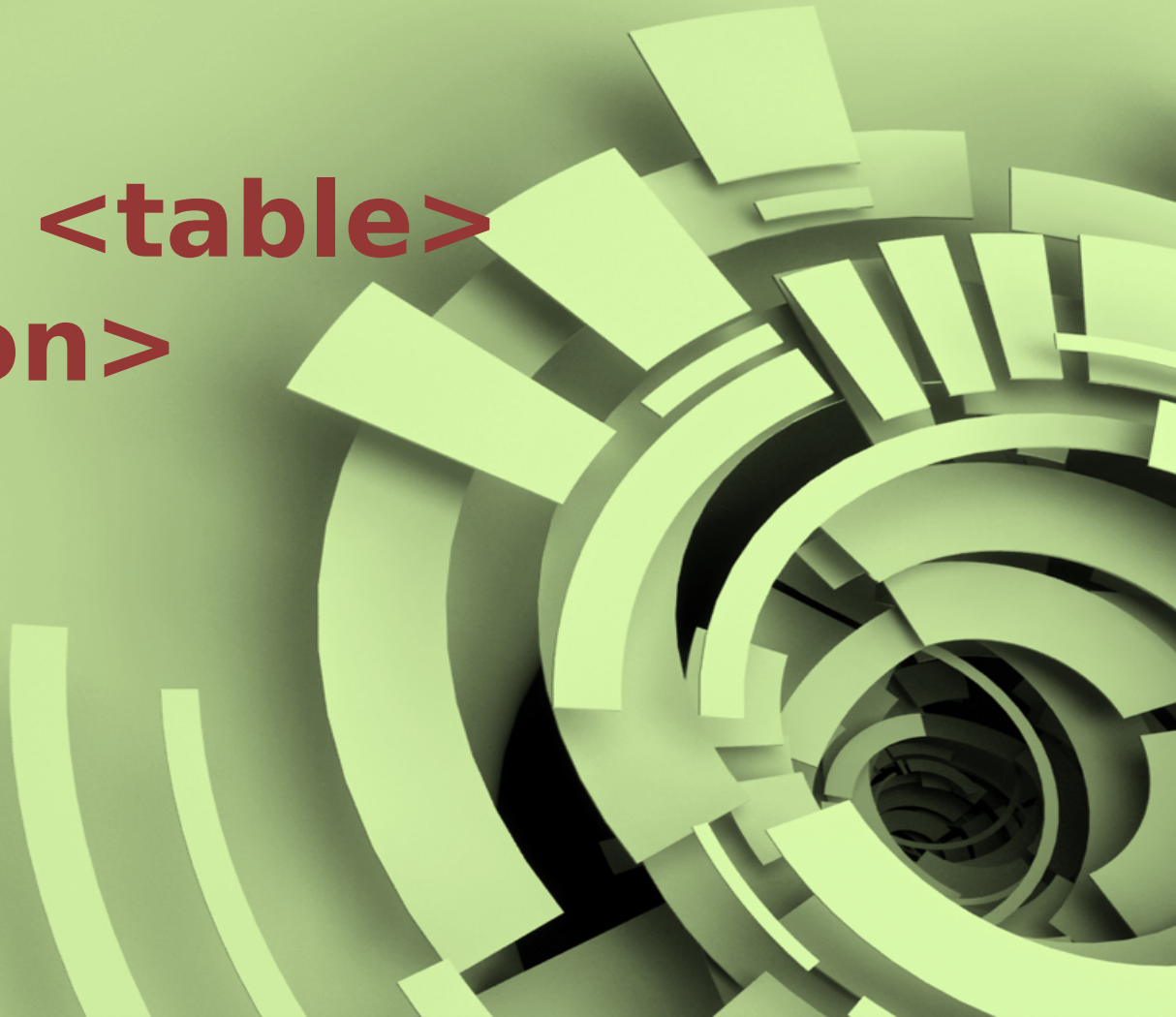


SQL in Action

- Modifying Data:
  - **Modifying Data:**
    - **>UPDATE <table> SET <field1> = <value1> AND <field2> = <value2> WHERE <conditions>**

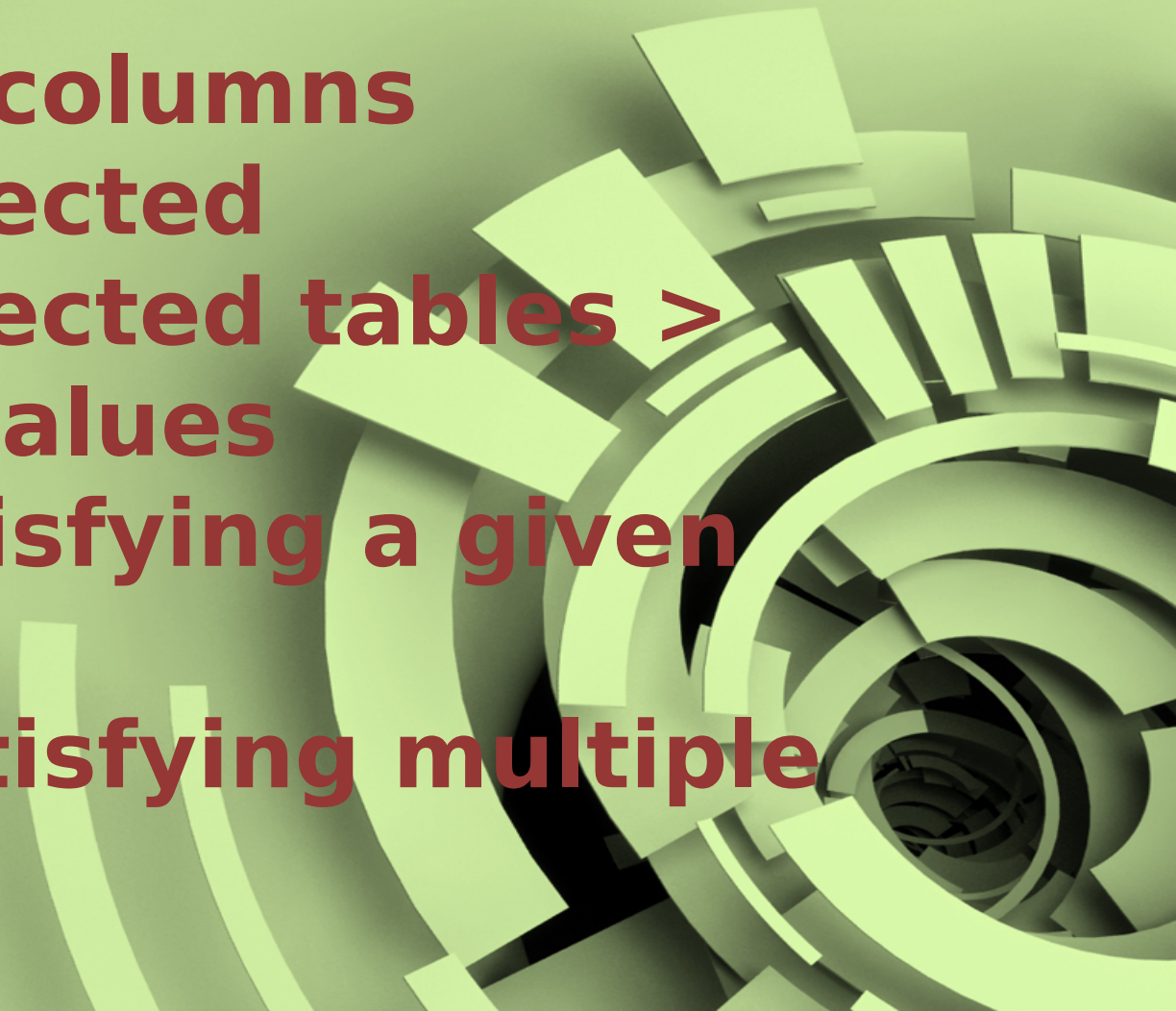
SQL in Action

- Deleting Data:
  - **Deleting Data:**
    >DELETE FROM <table>
    WHERE <condition>



SQL in Action

- Retrieving Data:
  - **Retrieve from all columns**
  - **Retrieve from selected**
  - **Retrieve from selected tables >**
  - **Retrieve unique values**
  - **Retrieve data satisfying a given condition**
  - **Retrieve data satisfying multiple conditions**

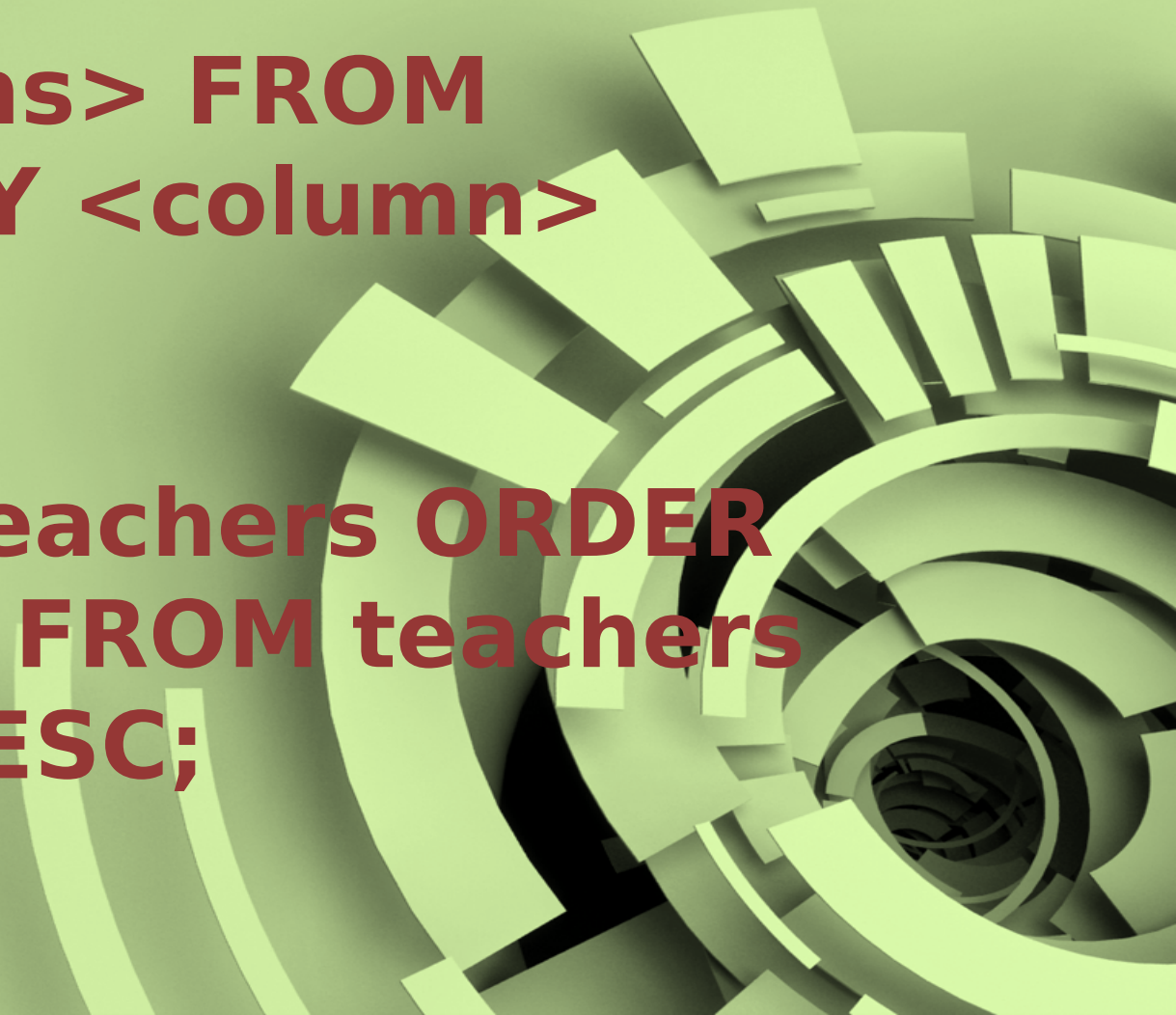


SQL in Action

- Sorting:

o>**SELECT <columns> FROM <table> ORDER BY <column> <ASC or DESC> >**

>SELECT * FROM teachers ORDER BY age; >SELECT * FROM teachers ORDER BY name DESC;

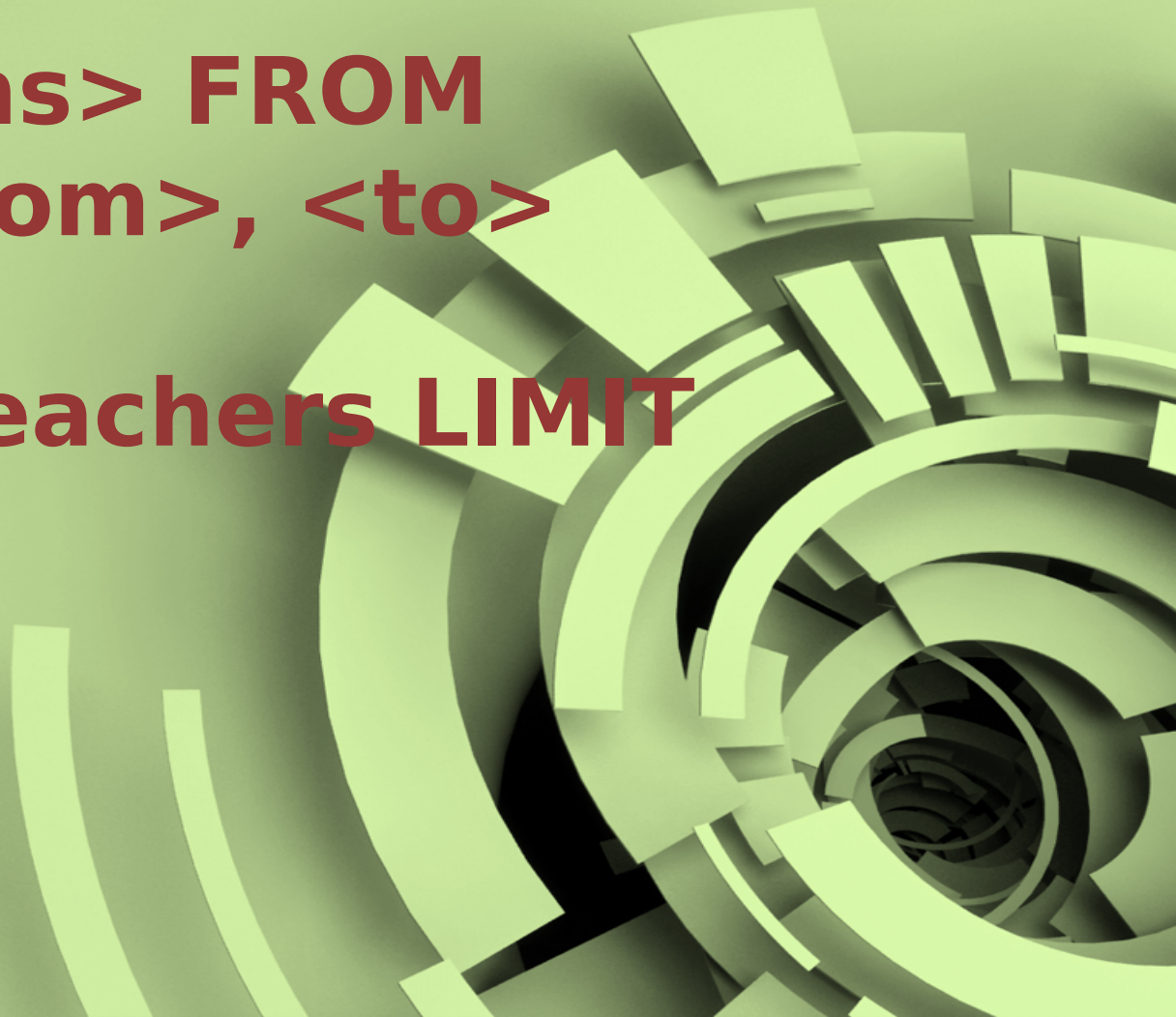


SQL in Action

- Limiting:

o>**SELECT <columns> FROM <table> LIMIT <from>, <to>**

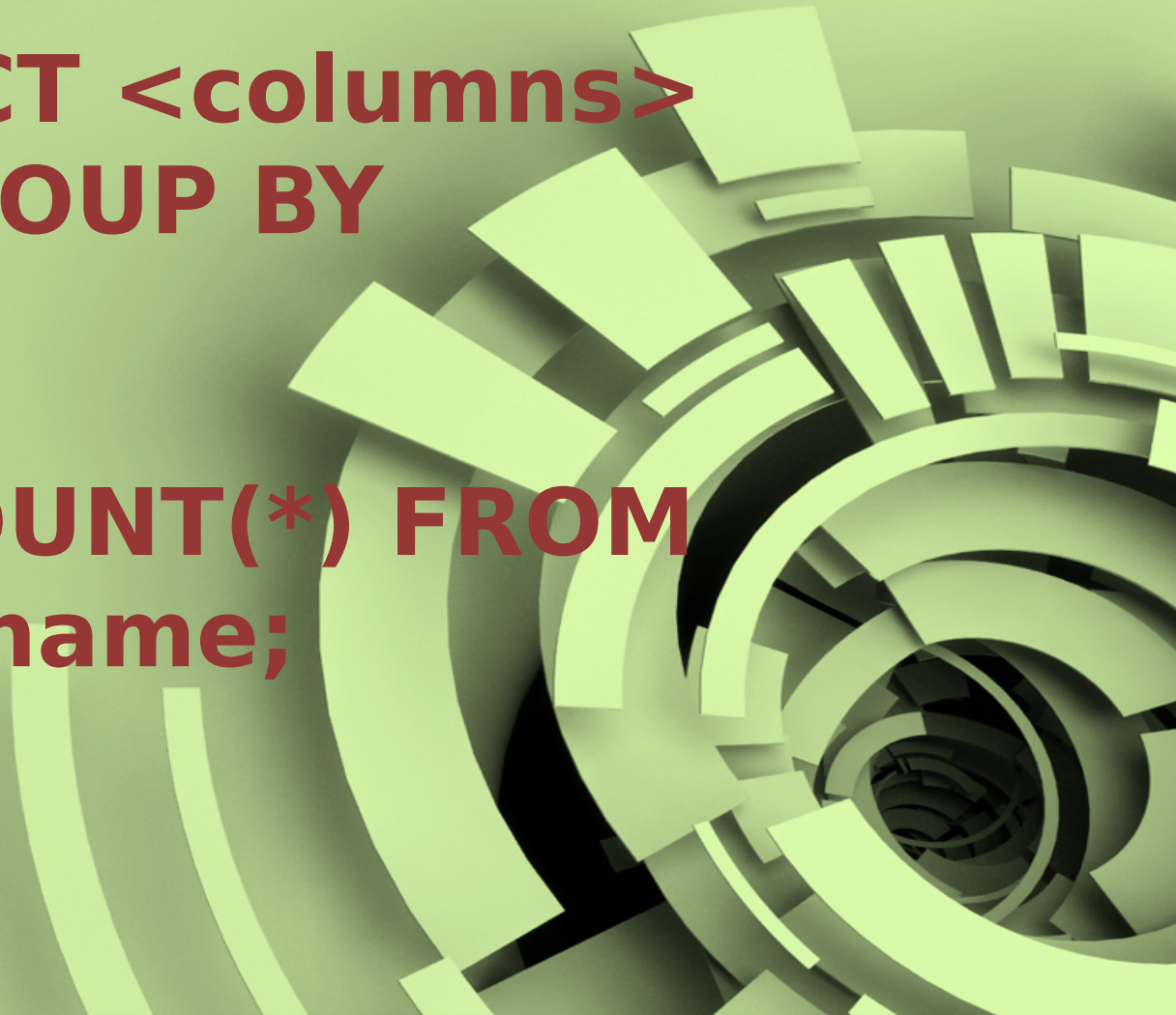
>SELECT * FROM teachers LIMIT 1,5;



SQL in Action

- Grouping:
  - **Grouping: >SELECT <columns> FROM <table> GROUP BY <column>**

>SELECT name, COUNT(*) FROM faculty GROUP BY name;



SQL in Action

- Pattern Matching Examples::
  - **>SELECT * FROM teachers WHERE name LIKE ‘j%’;**

Wildcard % selects joe, john, jones, etc.

  - **> SELECT * FROM teachers WHERE name LIKE ‘_ _ _’;**

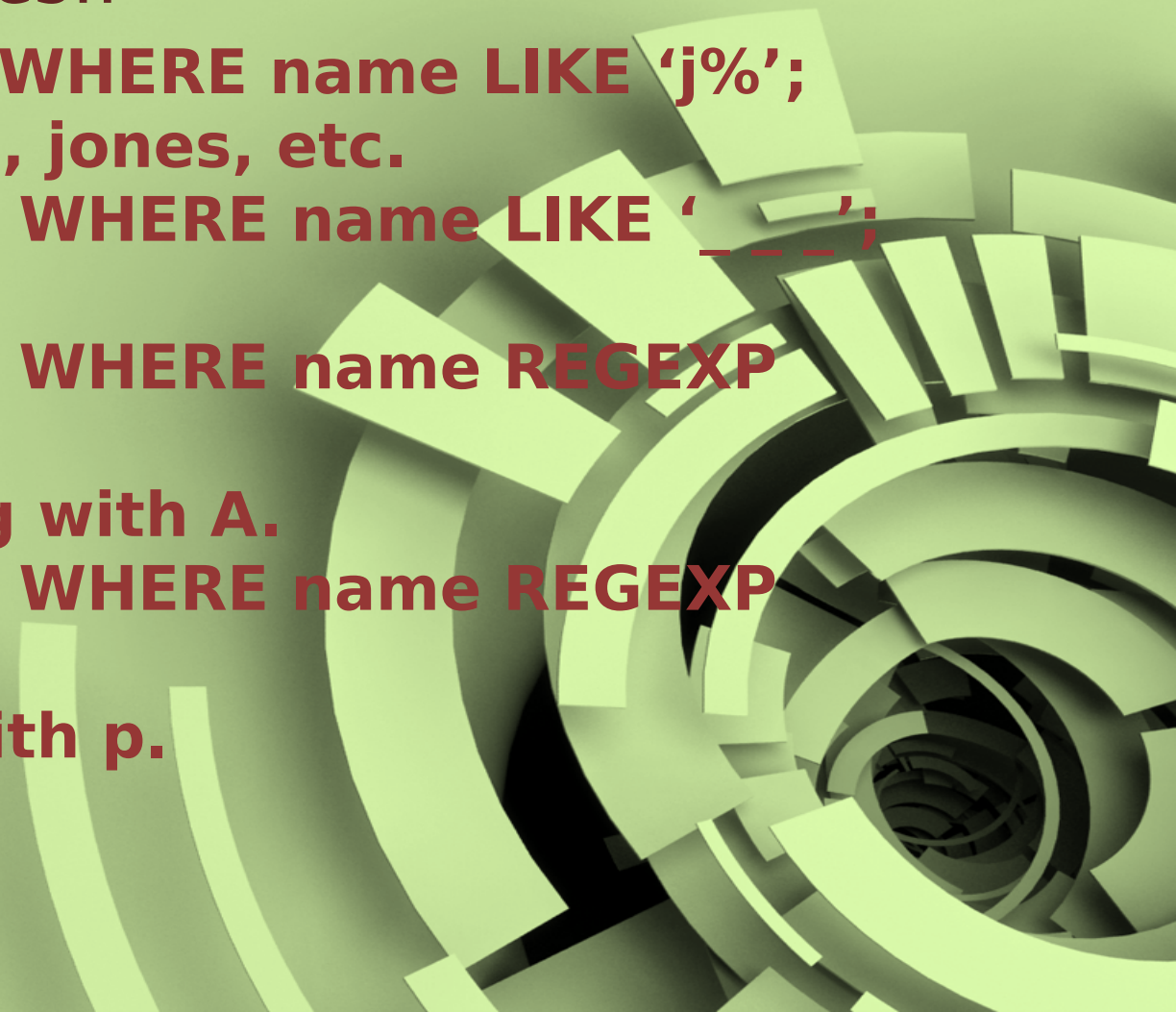
Selects 3 character values.

  - **> SELECT * FROM teachers WHERE name REGEXP ‘^A’;**

Selects all entries beginning with A.

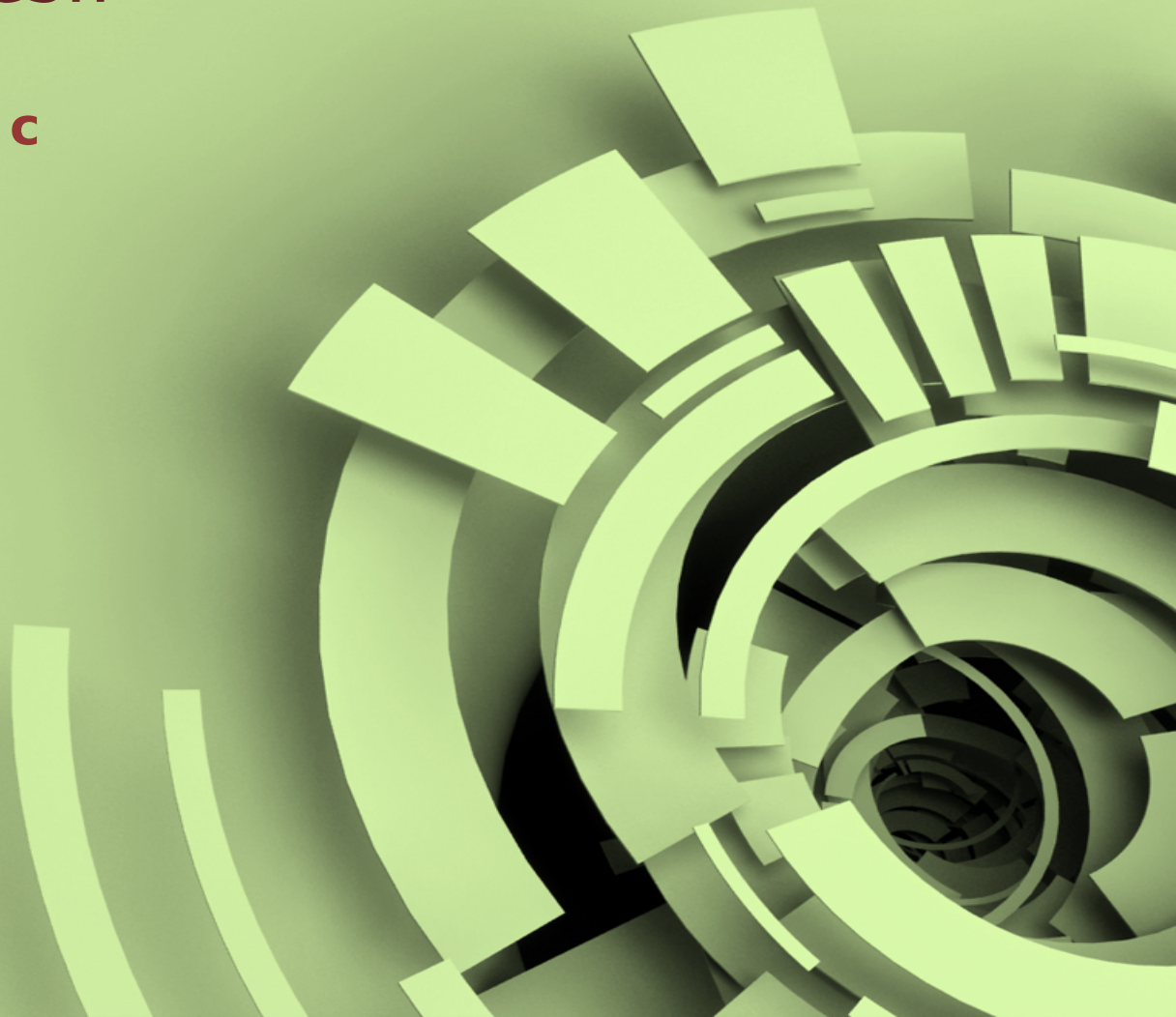
  - **> SELECT * FROM teachers WHERE name REGEXP ‘p$’;**

Selects all entries ending with p.



SQL in Action

- Pattern Matching Examples::
  - **[abc] match a, b, or c**
  - **[^abc] match all expect a, b, or c**
  - **[A-Z] match uppercase**
  - **[a-z] match lowercase**
  - **[0-9] match any digit**
  - *** match zero or more instances**
  - **+ match one or more instances**

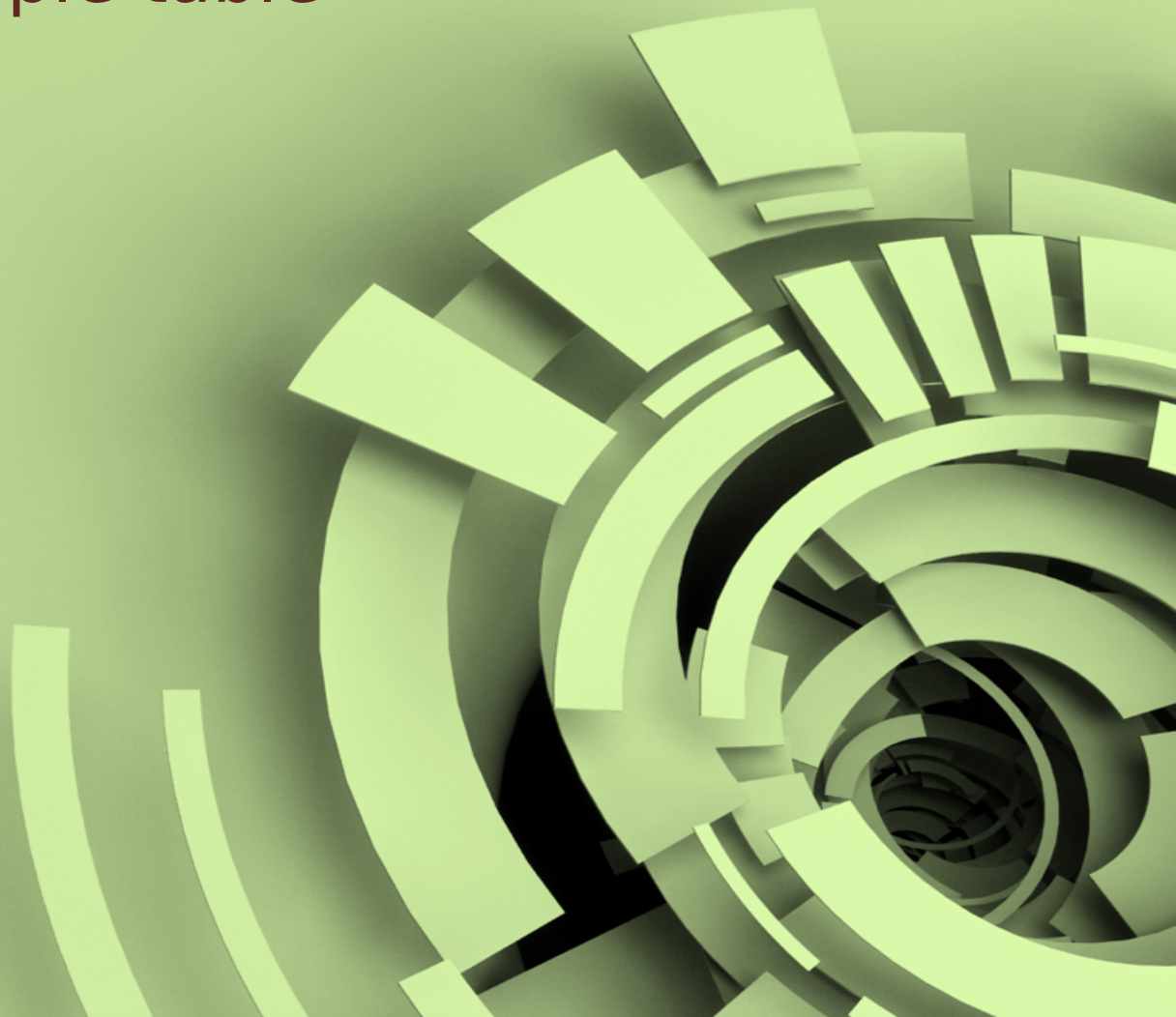


SQL in Action

- Pattern Matching Examples::
  - **. match any single char**
  - **^ match the beginning**
  - **$ match the end**
  - **| separates alternatives**
  - **{n,m} match at least n times but notmore than m times**
  - **{n} string must occur exactly n times**
  - **{n,} string must occur at least n times**

SQL in Action

- Retrieving Data from Multiple table
  - **LEFT JOIN**
  - **RIGHT JOIN**
  - **INNER JOIN**



Thank you!