

Unitedworld Institute Of Technology

B.Tech. Computer Science & Engineering

Semester : 3rd

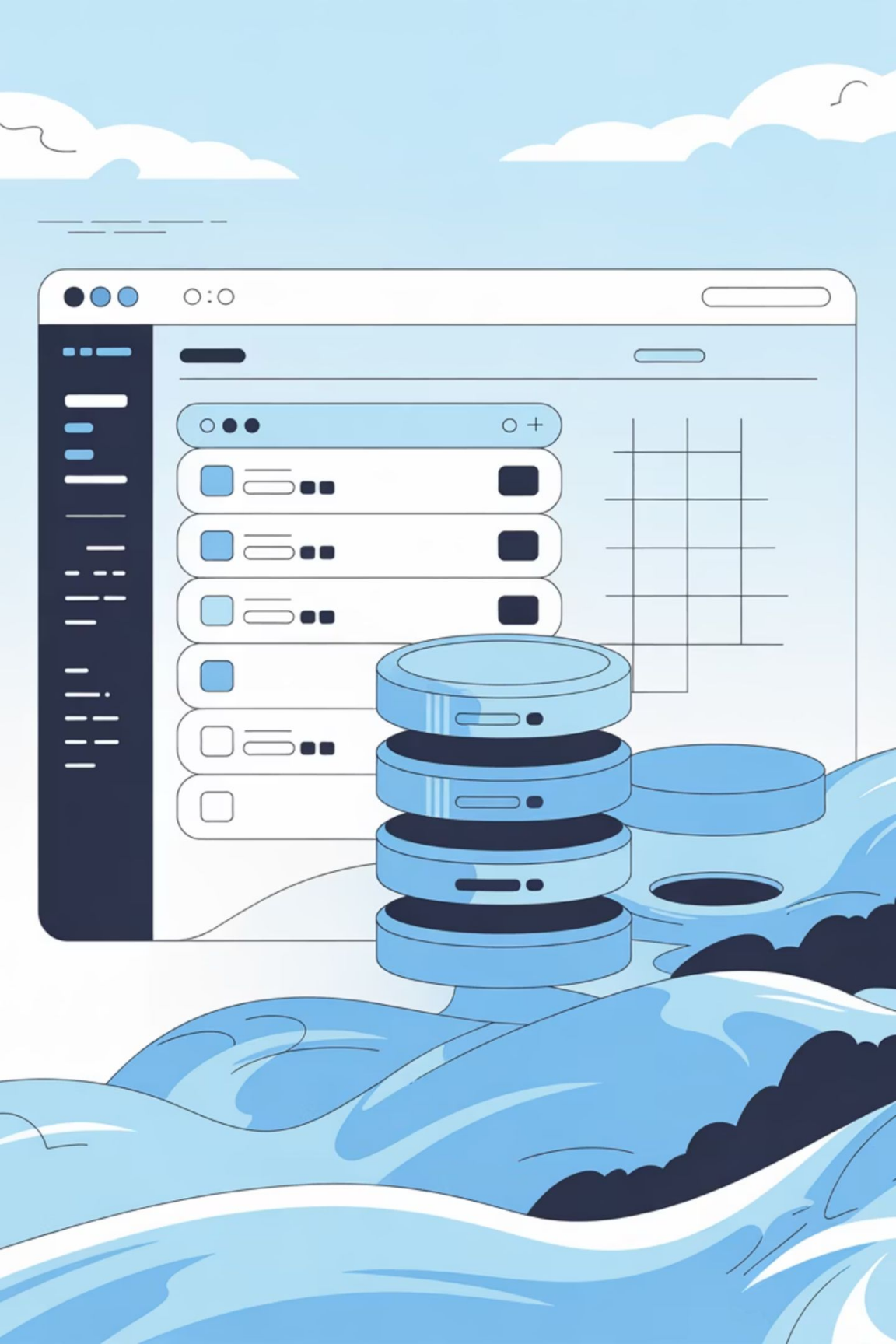
Introduction To Database Management System  
Course Code: 71203002003

Unit – 3 : STRUCTURED QUERY LANGUAGE - SQL AND PL/SQL

Prepared by:  
Mr. Utsav Kapadiya  
Assistant Professor (UIT)



## Views in SQL



# What is a View in SQL?

A View is a virtual table based on the result of a SQL query. It looks like a table, but it doesn't store data itself — it just displays data stored in other tables.

Think of it like a window (view) to look at certain data, without giving access to the entire table.

# Syntax:

```
CREATE VIEW view_name ASSELECT column1, column2, ...FROM table_nameWHERE condition;
```

# Simple Example

Let's say you have two tables:

## Students

student_id	name	department_id
1	Riya	101
2	Arjun	102
3	Sneha	101

## Departments

department_id	department_name
101	Computer Science
102	Electronics

# Example 1: Create a View

We want a view showing **students along with their department names**.

```
CREATE VIEW student_details AS
SELECT s.student_id, s.name, d.department_name
FROM Students s JOIN Departments d ON
s.department_id = d.department_id;
```

Now you can query it like a normal table:

```
SELECT * FROM student_details;
```

**Output:**

student_id	name	department_name
1	Riya	Computer Science
2	Arjun	Electronics
3	Sneha	Computer Science

## Example 2: View with Filter

If you only want Computer Science students:

```
CREATE VIEW cs_students AS SELECT name FROM Students WHERE department_id = 101;
```

Now query it:

```
SELECT * FROM cs_students;
```

**Output:**

name
Riya
Sneha

## Example 3: Update View

You can update the view definition:

```
CREATE OR REPLACE VIEW cs_students AS  
SELECT name, department_id  
FROM Students  
WHERE department_id = 101;
```



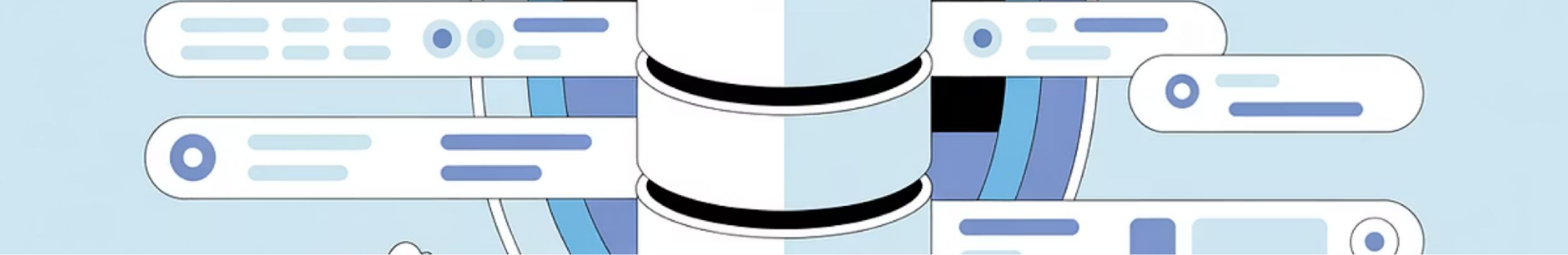
## Example 4: Drop View

To delete a view:

```
DROP VIEW cs_students;
```

# Key Concept:

Concept	Description
Virtual Table	Does not store data, only shows data from real tables.
Security	You can restrict access to specific columns or rows.
Simplifies Queries	Saves complex joins and filters for reuse.
Updatable View	Some views (not all) allow INSERT, UPDATE, or DELETE.



## Example of Security Use Case

Instead of giving users access to the full Employees table (which includes salary), you can create a safe view:

```
CREATE VIEW public_employees AS  
SELECT name, department  
FROM Employees;
```

This way, users can only see name and department — not sensitive data like salary.

# Views in SQL

Virtual tables that simplify data access and enhance security



**Thank You!**