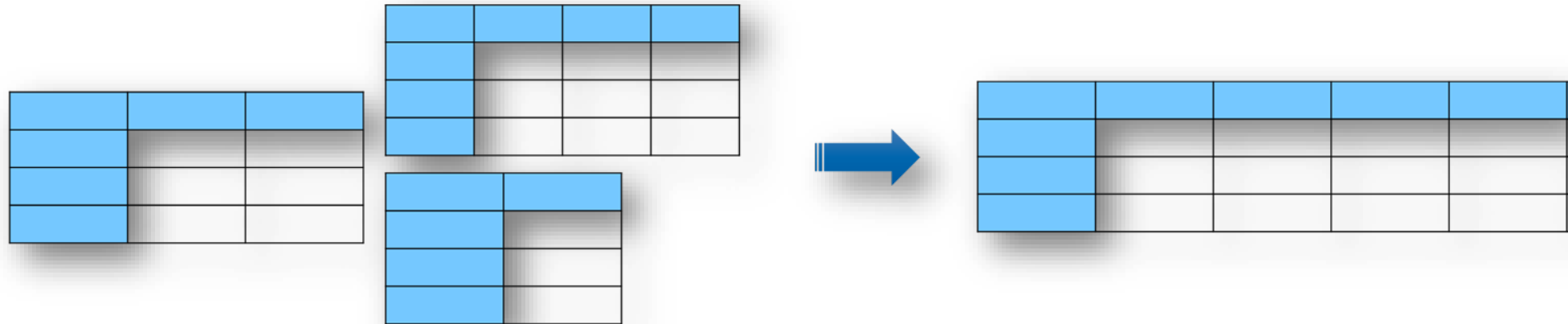
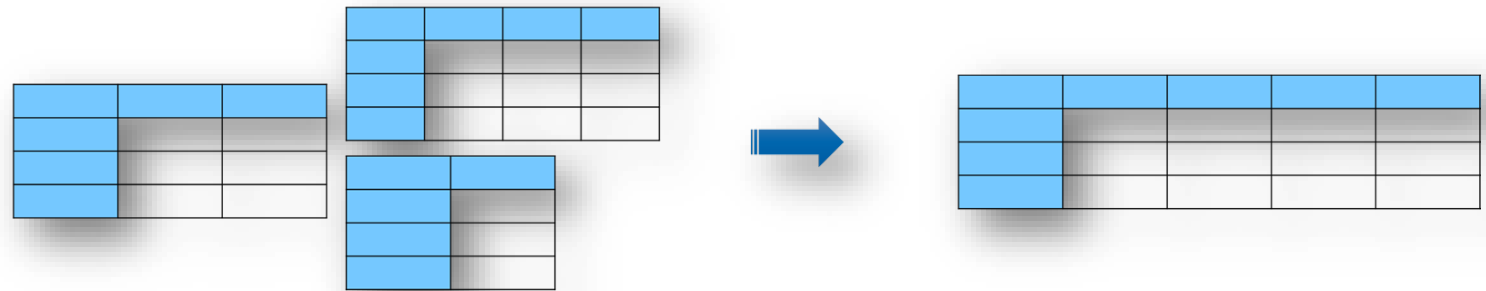


Views - Procedures



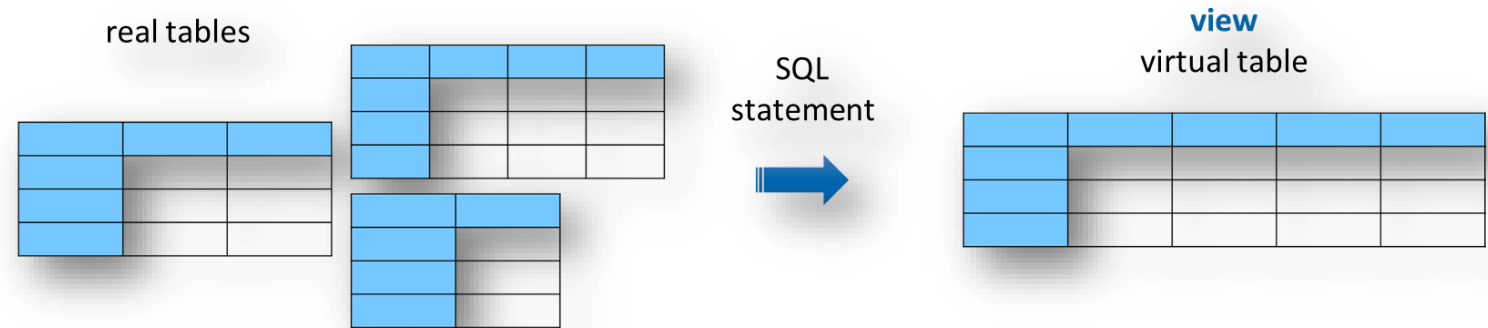
Content

- Views
- Procedures



Content

- Views
- Procedures

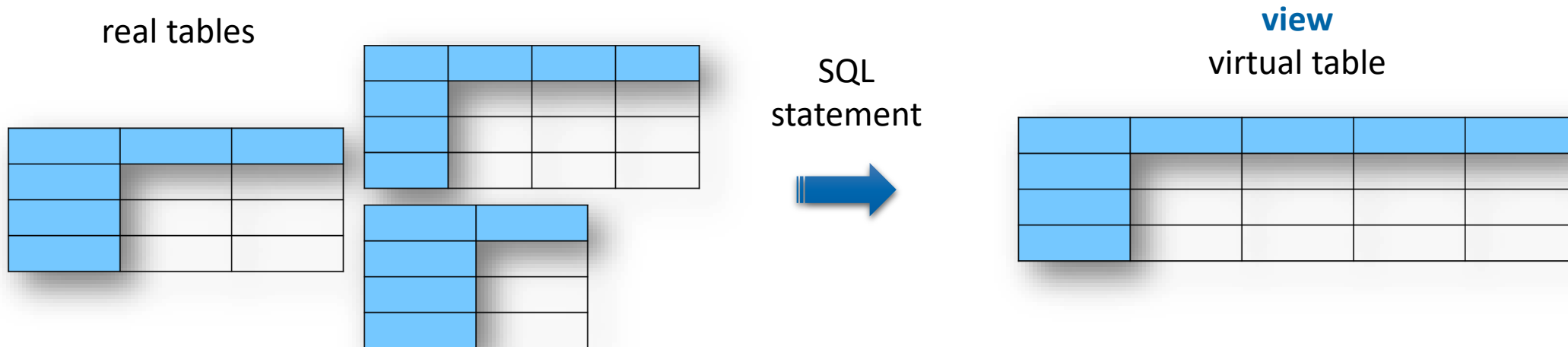


Views

A view is a **virtual table** based on the result of an SQL statement.

A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables.

You can add SQL functions, WHERE, and JOIN statements to a view and present the data as if the data were coming from one single table.



Views

CREATE – MODIFY – DELETE

Syntax

```
CREATE VIEW viewname AS  
SELECT attribute1, attribute2, ... FROM tablename  
WHERE condition;
```

Example

```
-- Create view for all clients with a gold abonnement  
CREATE VIEW gold_clients AS  
SELECT LastName, FirstName FROM Clients  
WHERE Clients.AboTypeID=3;
```

Views

CREATE – **MODIFY** – DELETE

Syntax

```
CREATE OR REPLACE VIEW viewname AS  
SELECT attribute1, attribute2, ... FROM tablename  
WHERE condition;
```

Example

```
-- Update view for all clients with a gold abonnement  
CREATE OR REPLACE VIEW gold_clients AS  
SELECT LastName, FirstName, Age FROM Clients  
WHERE Clients.AboTypeID=3;
```

Views

CREATE – MODIFY – DELETE

Syntax

DROP VIEW *viewname*;

Example

```
-- Delete view for all clients with a gold abonnement  
DROP VIEW gold_clients;
```

Views

Retrieve

Syntax

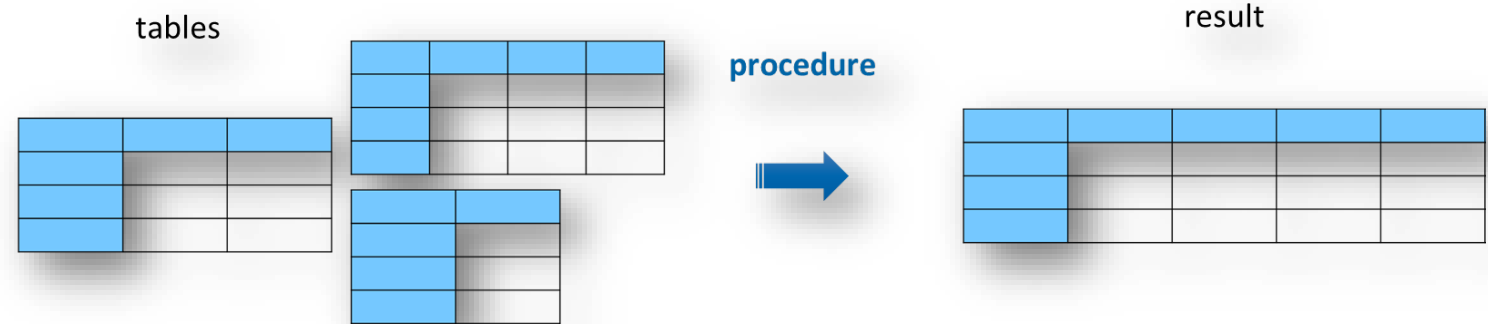
```
SELECT attribute01, attribute02, ... FROM viewname  
WHERE condition;
```

Example

```
-- Get all gold clients older than 35 years  
SELECT * FROM gold_clients  
WHERE Age>35;
```


Content

- Views
- Procedures

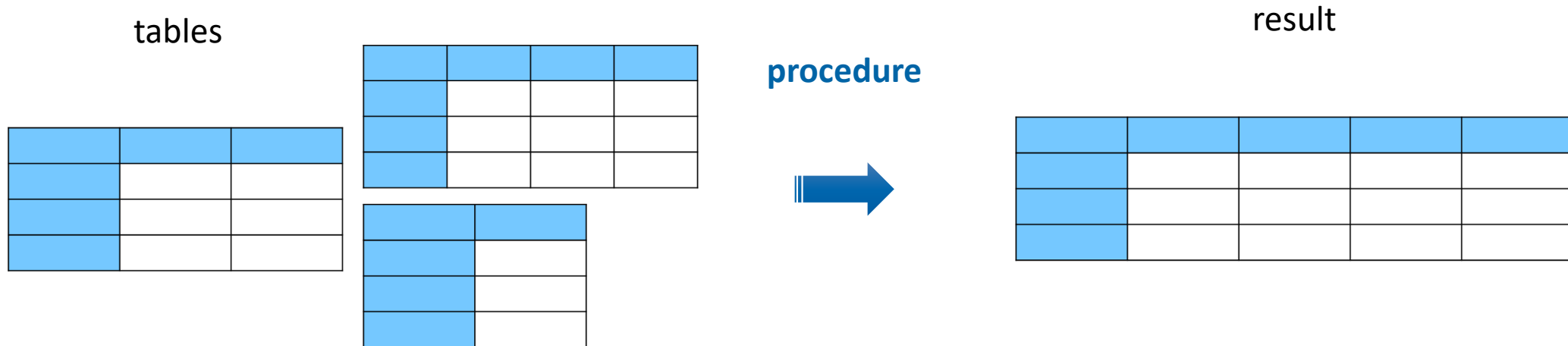


Procedures

A procedure is a **subroutine** that can be called with parameters.

A procedure usually contains predefined SQL queries with conditions that make use of the passed parameters.

Procedures are a way to extend the API and, thus, help reducing redundant coding.



Procedure

CREATE

Something you
don't use otherwise

Syntax

```
DELIMITER //  
CREATE PROCEDURE procedurename  
(IN parameter1 type1  
BEGIN  
    SELECT attribute1, attribute2, ... FROM tablename  
    WHERE condition;  
END //  
DELIMITER ;
```

Example

```
/* Create procedure to get all clients  
with gold abonnement above a given age */  
DELIMITER %%  
CREATE PROCEDURE gold_older  
(IN a int)  
BEGIN  
    SELECT FirstName, LastName FROM clients  
    WHERE AboTypeID=3 AND Age>a;  
END %%  
DELIMITER ;
```

Procedure

EXECUTE

Syntax

CALL *procedurename*(*parameter1*);

Example

```
-- Get all gold clients older than 35 years  
CALL gold_older(35);
```