Grouping and Aggregating Data

POWERING POTENTIAL

Version 1.5

Overview

- **Aggregate functions**
- **GROUP BY clause**
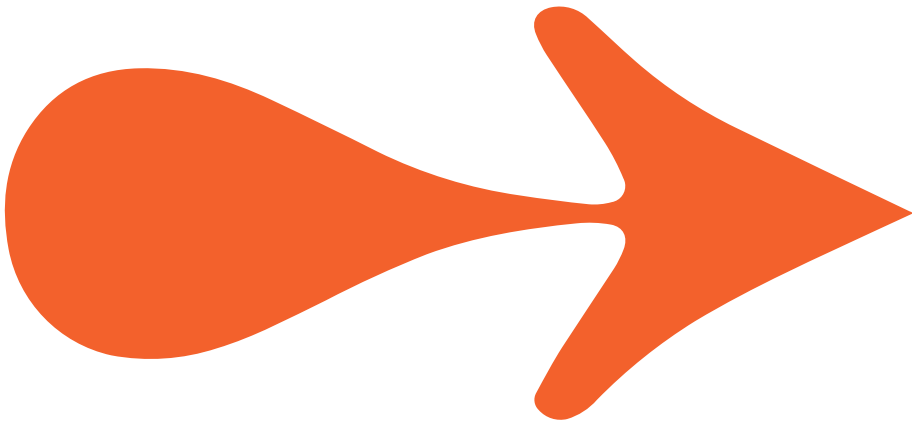- **HAVING clause**
- **Hands-on lab**

OBJECTIVES

At the end of this module you will be able to:

- write a query that calculates single aggregates
- write a query that calculates aggregates for grouped information
- write a query that aggregates, groups and filters on calculated values
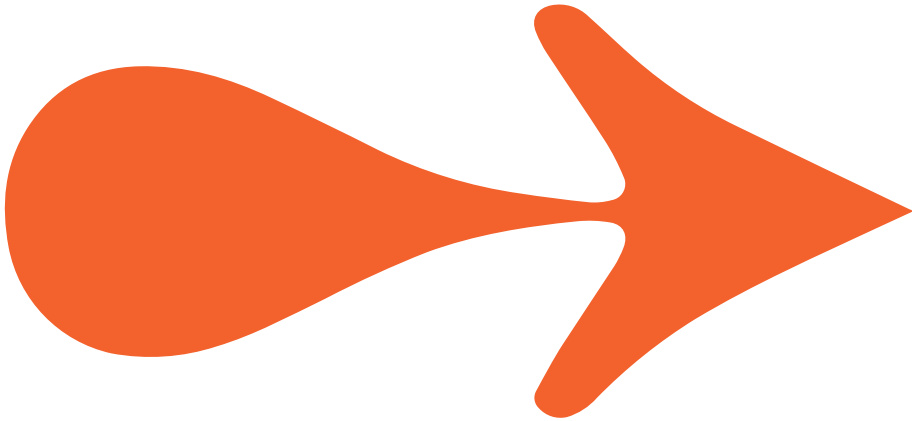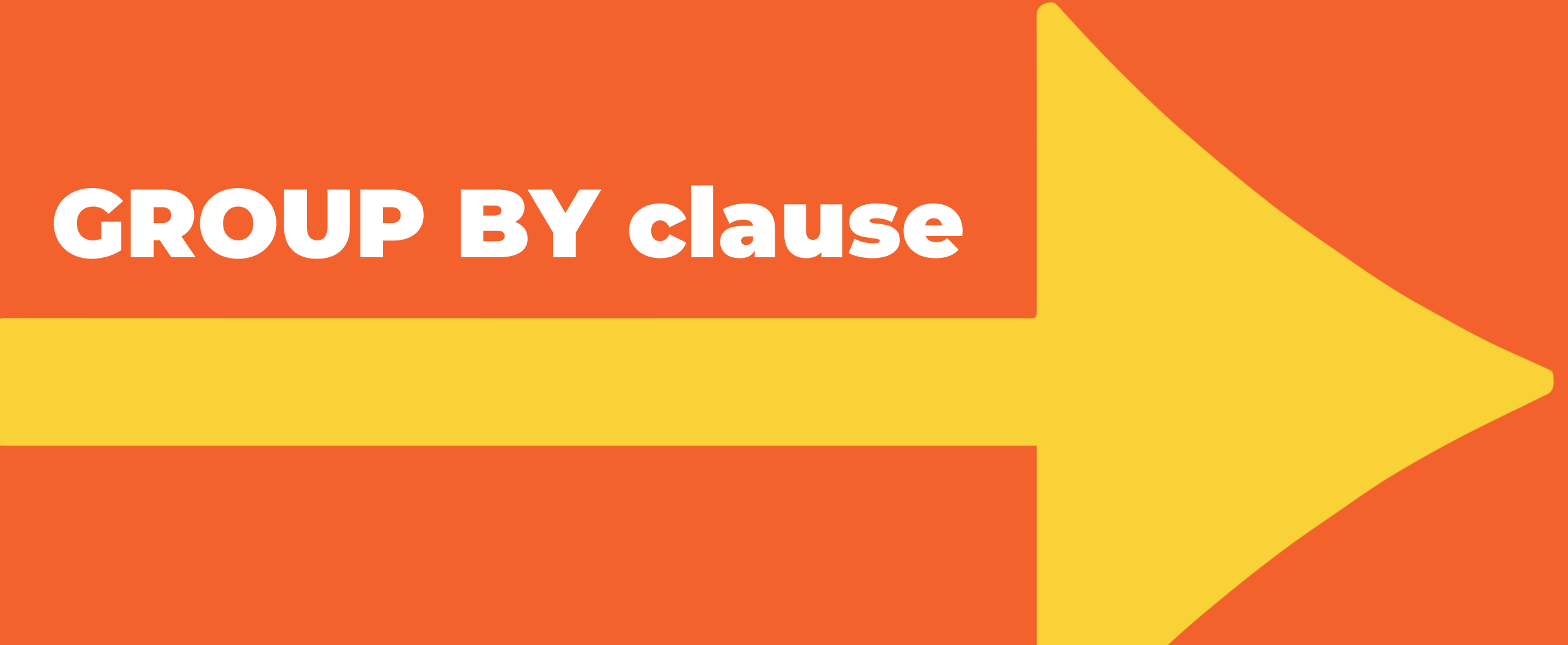
Aggregate functions

Aggregate functions

- **COUNT(*)**
- **COUNT(**<<*expression*>>**)**
- **SUM(**<<*expression*>>**)**
- **AVG(**<<*expression*>>**)**
- **MIN(**<<*expression*>>**) / MAX(**<<*expression*>>**)**
- **Statistical aggregates**
  → STDEV / STDEVP / VAR / VARP

GROUP BY clause

GROUP BY clause

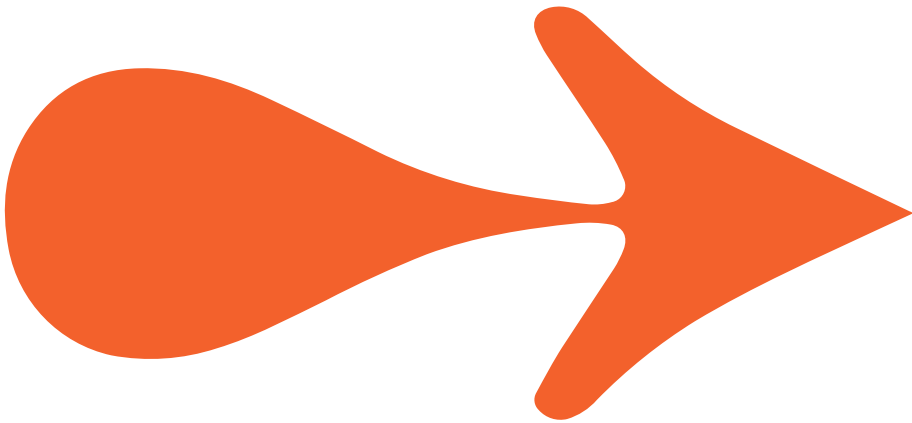- SELECT <<field(s)>>
- FROM <<table(s)>>
- WHERE <<condition(s)>>
- **GROUP BY** *<<field(s)>>*
- HAVING *<<condition(s)>>*
- ORDER BY <<field(s)>>

Grouping aggregates

```
SELECT
    CategoryID,
    AVG(UnitPrice)
FROM
    dbo.Products
GROUP BY
    CategoryID
```

	CategoryID	(No column name)
1	1	37.9791
2	2	23.0625
3	3	25.16
4	4	28.73
5	5	20.25
6	6	54.0066
7	7	32.37
8	8	20.6825

ent (55) Northwind 00:00:00 8 rows
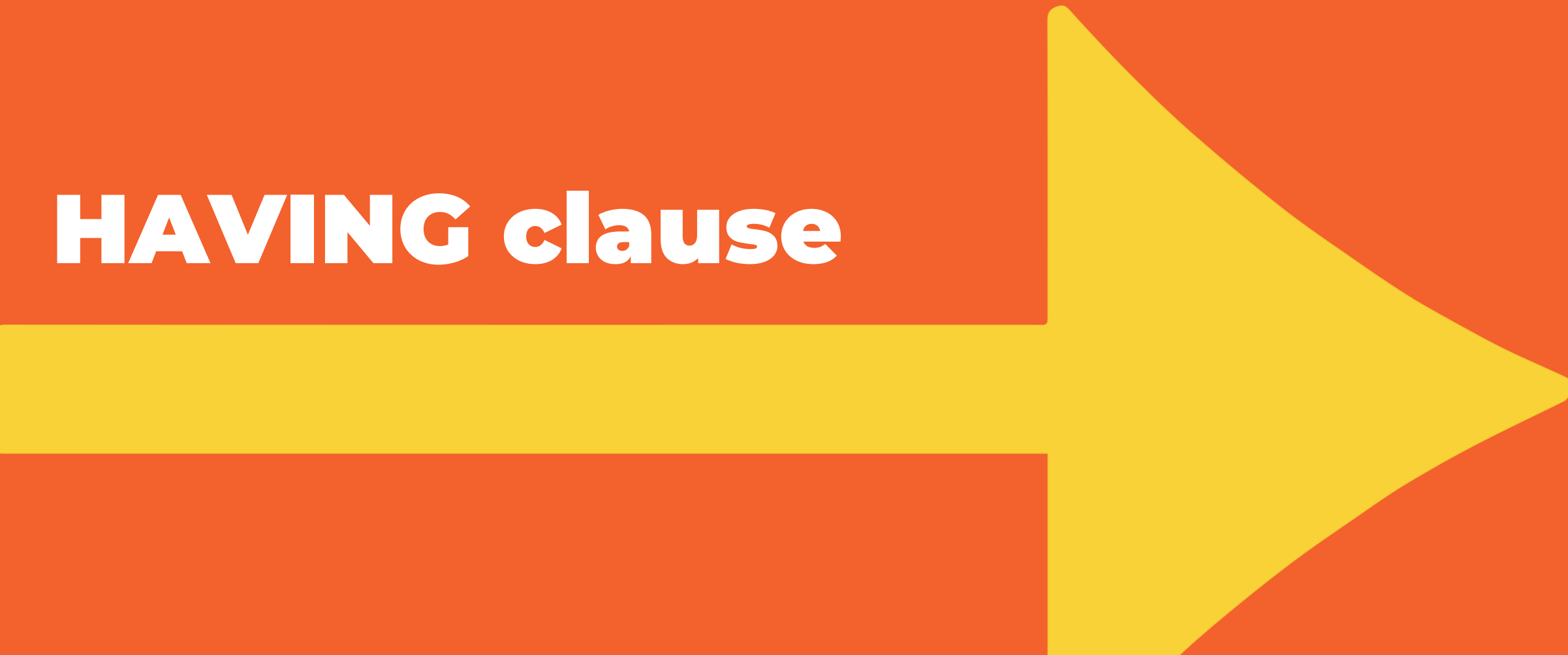
```
SELECT
    CategoryID,
    AVG(UnitPrice)
FROM
    dbo.Products
GROUP BY
    CategoryID
ORDER BY 2 DESC
```

	CategoryID	(No column name)
1	6	54.0066
2	1	37.9791
3	7	32.37
4	4	28.73
5	3	25.16
6	2	23.0625
7	8	20.6825
8	5	20.25

dent (55) Northwind 00:00:00 8 rows

HAVING clause

HAVING clause

- SELECT <<field(s)>>
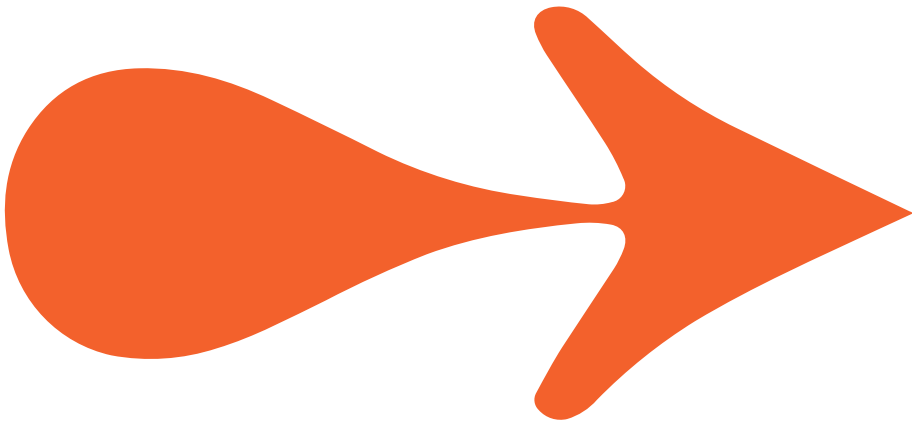- FROM <<table(s)>>
- WHERE *<<row-level filter(s)>>*
- **GROUP BY** *<<field(s)>>*
- **HAVING** *<<group-level filters(s)>>*
- ORDER BY <<field(s)>>

Filtering grouped aggregates

```
SELECT
    CategoryID,
    AVG(UnitPrice)
FROM
    dbo.Products
GROUP BY
    CategoryID
HAVING
    AVG(UnitPrice) > 30
```

Hands-on lab

- **Single aggregates**
- **GROUP BY**
- **HAVING**

Review

- **Aggregate functions**
- **GROUP BY**
- **HAVING**