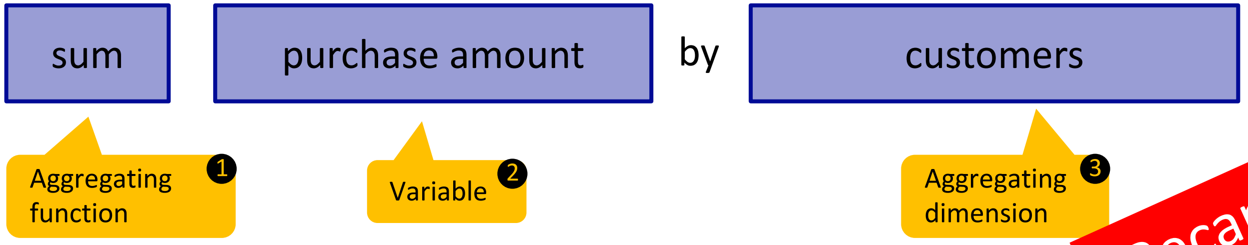
Aggregate operations in SQL

Aggregating means:
"do <<function>> to <<variable>> by <<dimension>>"

Aggregating has 2 components:

- Function and variable by which to aggregate.
- Dimension by which to aggregate.

For example:



sum purchase amount by customers

1 Aggregating function

2 Variable

3 Aggregating dimension

Recap Lecture 17: Aggregate

Various aggregate operations are possible in SQL

1. Apply an aggregating function on **a variabl**e by an aggregating dimension
2. Apply an aggregating function to **multiple variables** by an aggregating dimension
3. Aggregate a variable by a **transformed aggregating dimension**
4. Creating **new columns** using an aggregating dimension

1. Apply an aggregating function to a variable by an aggregating dimension

Customer	TransDate	Quantity	PurchAmount	Cost	TransID
149332	15.11.2005	1	199.95	107.00	127998739
172951	29.08.2008	1	199.95	108.00	128888288
120621	19.10.2007	1	99.95	49.00	125375247
149236	14.11.2005	1	39.95	18.95	127996226
149236	12.06.2007	1	79.95	35.00	128670302
…	…	…	…	…	…

Sum all purchase amounts by `Customer` and call it `AggPurch`

Customer	AggPurch
149332	199.95
172951	199.95
120621	99.95
149236	119.90
…	…

R data.table

```
transactions[,list(AggPurch=sum(PurchAmount)),by=Customer]
```

SQL in R

```
dbGetQuery(con,"SELECT Customer,
                SUM(PurchAmount) AS AggPurch
                FROM transactions
                GROUP BY Customer;")
```

2 Specify the aggregating dimension with `GROUP BY`

1 Set the name of the new column

1. Apply an aggregating function to a variable by an aggregating dimension

Customer	TransDate	Quantity	PurchAmount	Cost	TransID
149332	15.11.2005	1	199.95	107.00	127998739
172951	29.08.2008	1	199.95	108.00	128888288
120621	19.10.2007	1	99.95	49.00	125375247
149236	14.11.2005	1	39.95	18.95	127996226
149236	12.06.2007	1	79.95	35.00	128670302
…	…	…	…	…	…

Sum all purchase amounts by `Customer` and call it `AggPurch`

Customer	AggPurch
149332	199.95
172951	199.95
120621	99.95
149236	119.90
…	…

R data.table

```
transactions[,list(AggPurch=sum(PurchAmount)),by=Customer]
```

SQL in R

```
dbGetQuery(con,"SELECT Customer,
                SUM(PurchAmount) AS AggPurch
                FROM transactions
                GROUP BY Customer;")
```

2 Specify the aggregating dimension with `GROUP BY`

1 Set the name of the new column

1. Apply an aggregating function to a variable by an aggregating dimension

Customer	TransDate	Quantity	PurchAmount	Cost	TransID
149332	15.11.2005	1	199.95	107.00	127998739
172951	29.08.2008	1	199.95	108.00	128888288
120621	19.10.2007	1	99.95	49.00	125375247
149236	14.11.2005	1	39.95	18.95	127996226
149236	12.06.2007	1	79.95	35.00	128670302
…	…	…	…	…	…

Sum all purchase amounts by `Customer` and call it `AggPurch`

Customer	AggPurch
149332	199.95
172951	199.95
120621	99.95
149236	119.90
…	…

R data.table

```
transactions[,list(AggPurch=sum(PurchAmount)),by=Customer]
```

SQL in R

```
dbGetQuery(con,"SELECT Customer,
                 SUM(PurchAmount) AS AggPurch
                 FROM transactions
                 GROUP BY Customer;")
```

Set the name of the new column 1

Specify the aggregating dimension with `GROUP BY` 2

2. Apply an aggregating function to multiple variables by an aggregating dimension

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
…	…	…	…	…

Sum all purchase amounts and sum all quantities by `Customer`

Customer	AggPurch	AggQuant
149332	274.85	3
172951	889.80	4
120621	99.95	1
149236	119.90	2
…	…	

R data.table

```
transactions[,list(AggPurch=sum(PurchAmount),
                   AggQuant=sum(Quantity)), by=Customer]
```

SQL in R

```
dbGetQuery(con, "SELECT Customer, SUM(PurchAmount)
                 AS AggPurch, SUM(Quantity) AS AggQuant
                 FROM transactions
                 GROUP BY Customer;")
```

Separate all new variables by commas

3. Apply an aggregating function by a transformed aggregating dimension

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
…	…	…	…	…

Sum `PurchAmount` by month of each year

Month	AggPurch
2004-12-01	27623.90
2005-01-01	83363.73
2005-02-01	87341.59
…	…

R data.table

```
transactions[,list(AggPurch=sum(PurchAmount)),
          by=(Date=floor_date(TransDate, unit="month"))]
```

SQL in R

```
dbGetQuery(con, "SELECT date(TransDate, 'start of month')
                 AS Month, SUM(PurchAmount) AS AggPurch
                 FROM transactions
                 GROUP BY Month;")
```

3. Apply an aggregating function by a transformed aggregating dimension

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
…	…	…	…	…

Sum `PurchAmount` by month of each year

Month	AggPurch
2004-12-01	27623.90
2005-01-01	83363.73
2005-02-01	87341.59
…	…

R data.table

```
transactions[,list(AggPurch=sum(PurchAmount)),
          by=(Date=floor_date(TransDate, unit="month"))]
```

SQL in R

```
dbGetQuery(con, "SELECT date(TransDate, 'start of month')
                 AS Month, SUM(PurchAmount) AS AggPurch
                 FROM transactions
                 GROUP BY Month;")
```

3. Apply an aggregating function by a transformed aggregating dimension

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...

Sum `PurchAmount` by month of each year

Month	AggPurch
2004-12-01	27623.90
2005-01-01	83363.73
2005-02-01	87341.59
...	...

R data.table

```
transactions[,list(AggPurch=sum(PurchAmount)),
                    by=(Date=floor_date(TransDate, unit="month"))]
```

SQL in R

```
dbGetQuery(con, "SELECT date(TransDate, 'start of month')
                 AS Month, SUM(PurchAmount) AS AggPurch
                 FROM transactions
                 GROUP BY Month;")
```