

H7 Inleiding SQL -- Select.

SQL
Select statement

**HO
GENT**



Structured Query Language (SQL).

SQL – standaarden en dialecten

- Definitie:
 - relationele gegevenstaal voor relationele database systemen
 - niet-procedurele taal
- Standaard : ANSI/ISO-1992
- Database managementsystemen
 - Oracle : PL/SQL
 - SQL Server : TRANSACT-SQL
 - DB2 (IBM)
 - Informix
 - Sybase
 - MySQL

SQL – overzicht

- SQL bestaat uit 3 subtalen
 - **Data Definition Language (DDL)**
 - creatie van een database, en het definiëren van database objecten (tabellen, stored procedures, views,...)
 - CREATE, ALTER, DROP
 - **Data Manipulation Language (DML)**
 - opvragen en manipuleren van de gegevens in een database
 - SELECT, INSERT, UPDATE, DELETE
 - **Data Control Language (DCL)**
 - gegevensbeveiliging en autorisatie
 - GRANT, REVOKE, DENY
- Bijkomende taalelementen: operatoren, functies , control of flow (dialect gebonden)

DML – raadplegen van gegevens

- Eén tabel raadplegen
 - Basisvorm
 - SELECT clause
 - WHERE clause
 - Formatteren van rijen
 - Statistische functies
 - Groeperen
- Meerdere tabellen raadplegen



Select statement.

Basisvorm SELECT statement

- SELECT voor raadplegen van 1 tabel

```
SELECT [ALL | DISTINCT] {*|uitdrukking [,uitdrukking ...]}  
FROM tabelnaam  
[WHERE voorwaarde(n)]  
[GROUP BY kolomnaam [,kolomnaam ...]]  
[HAVING voorwaarde(n)]  
[ORDER BY {kolomnaam|volgnr}{ASC|DESC}[,...]]
```

- SELECT clause: specificeert de kolommen die je wenst te zien. DISTINCT zorgt ervoor dat de getoonde rijen alle uniek zijn
- FROM clause: geeft aan uit welke tabel de gegevens afkomstig zijn
- WHERE clause: opgave van de voorwaarden waaraan de getoonde rijen moeten voldoen
- ORDER BY clause: bepaalt de volgorde waarin de rijen getoond moeten worden
- GROUP BY en HAVING clause: groeperen van de gegevens

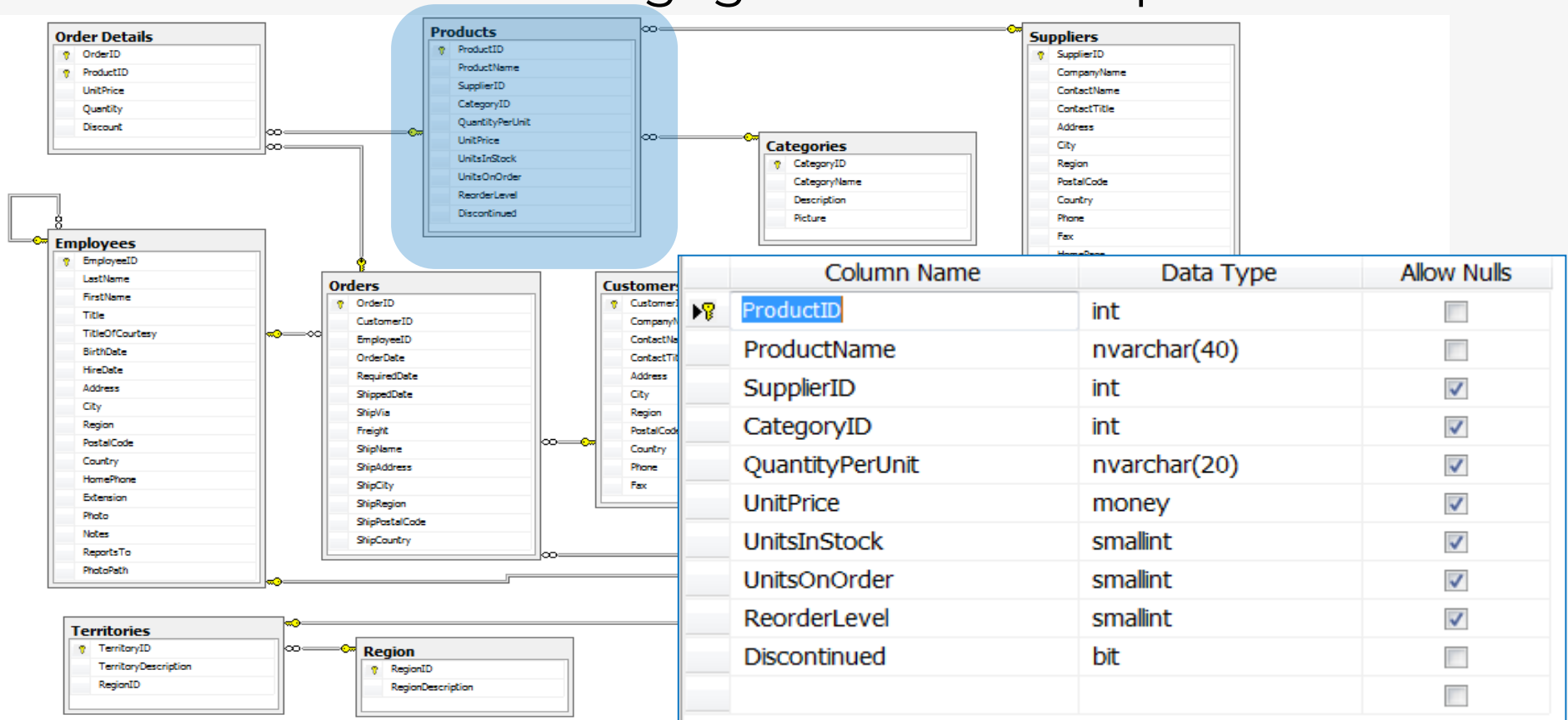
SELECT statement

SELECT clause: specificatie van de kolommen

- ophalen van alle kolommen uit 1 tabel:
gebruik *
 - SELECT *
- ophalen van specifieke kolommen: gebruik
kolomnamen of uitdrukking
 - SELECT kolom1, kolom2, ...

SELECT statement

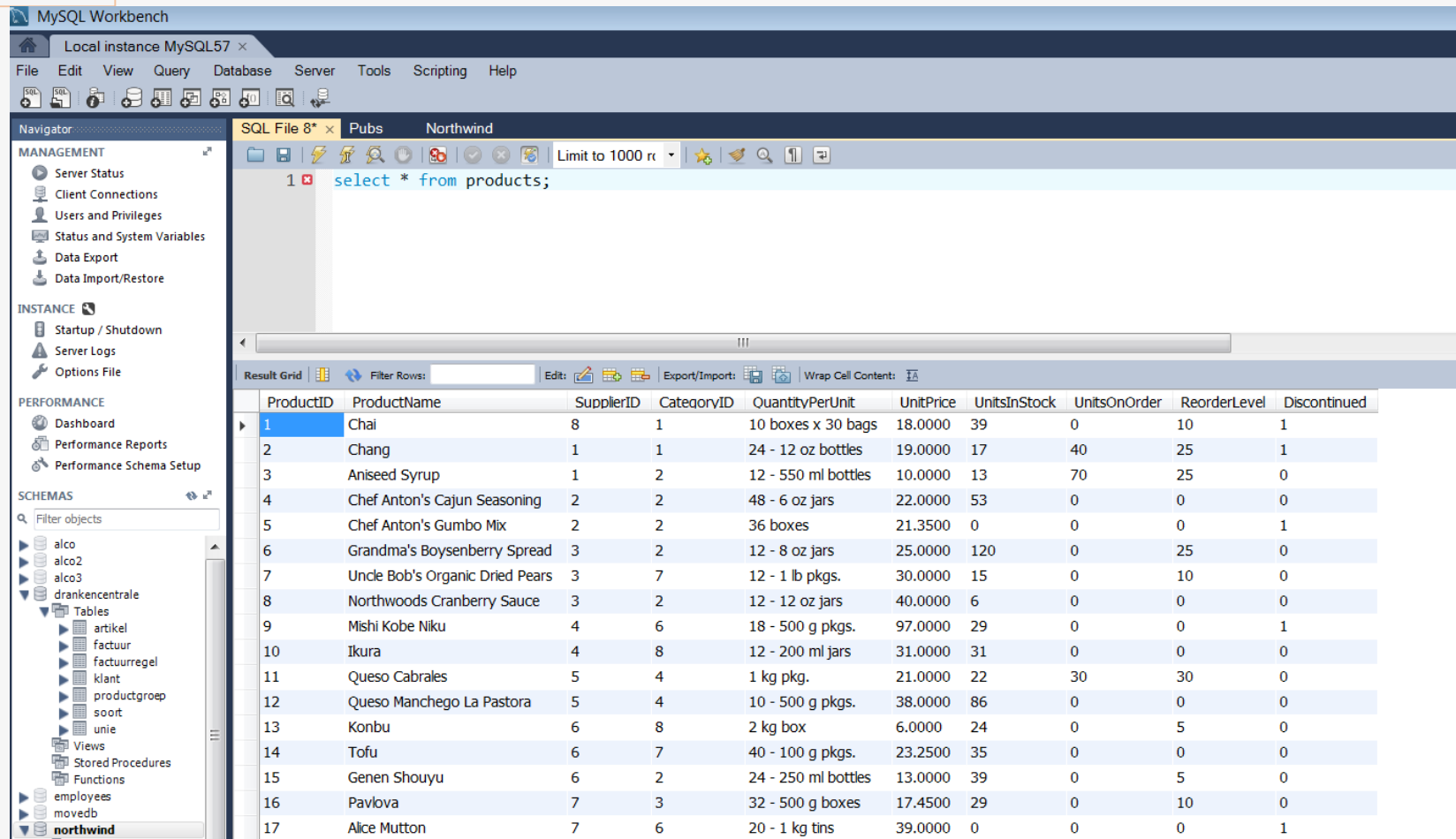
Voorbeeld1 - Toon alle gegevens van alle producten



SELECT statement

Voorbeeld1 - Toon alle gegevens van alle producten

```
SELECT *  
FROM products
```



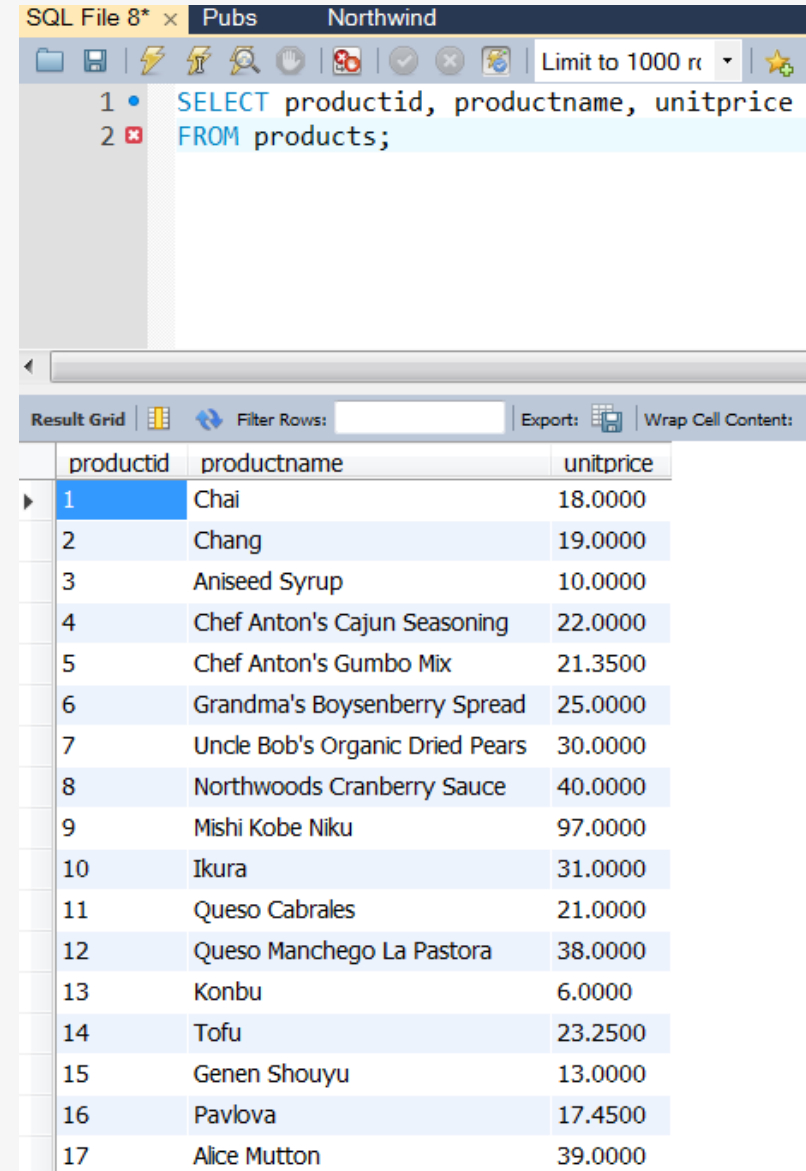
The screenshot shows the MySQL Workbench interface. The SQL editor contains the query `select * from products;`. The Results window displays a table with 10 columns: ProductID, ProductName, SupplierID, CategoryID, QuantityPerUnit, UnitPrice, UnitsInStock, UnitsOnOrder, ReorderLevel, and Discontinued. The table lists 17 products from the Northwind database.

ProductID	ProductName	SupplierID	CategoryID	QuantityPerUnit	UnitPrice	UnitsInStock	UnitsOnOrder	ReorderLevel	Discontinued
1	Chai	8	1	10 boxes x 30 bags	18.0000	39	0	10	1
2	Chang	1	1	24 - 12 oz bottles	19.0000	17	40	25	1
3	Aniseed Syrup	1	2	12 - 550 ml bottles	10.0000	13	70	25	0
4	Chef Anton's Cajun Seasoning	2	2	48 - 6 oz jars	22.0000	53	0	0	0
5	Chef Anton's Gumbo Mix	2	2	36 boxes	21.3500	0	0	0	1
6	Grandma's Boysenberry Spread	3	2	12 - 8 oz jars	25.0000	120	0	25	0
7	Uncle Bob's Organic Dried Pears	3	7	12 - 1 lb pkgs.	30.0000	15	0	10	0
8	Northwoods Cranberry Sauce	3	2	12 - 12 oz jars	40.0000	6	0	0	0
9	Mishi Kobe Niku	4	6	18 - 500 g pkgs.	97.0000	29	0	0	1
10	Ikura	4	8	12 - 200 ml jars	31.0000	31	0	0	0
11	Queso Cabrales	5	4	1 kg pkg.	21.0000	22	30	30	0
12	Queso Manchego La Pastora	5	4	10 - 500 g pkgs.	38.0000	86	0	0	0
13	Konbu	6	8	2 kg box	6.0000	24	0	5	0
14	Tofu	6	7	40 - 100 g pkgs.	23.2500	35	0	0	0
15	Genen Shouyu	6	2	24 - 250 ml bottles	13.0000	39	0	5	0
16	Pavlova	7	3	32 - 500 g boxes	17.4500	29	0	10	0
17	Alice Mutton	7	6	20 - 1 kg tins	39.0000	0	0	0	1

SELECT statement

Voorbeeld 2:
toon van alle producten
het productID,
de naam,
de eenheidsprijs

```
SELECT productid,  
       productname,  
       unitprice  
FROM products
```



The screenshot shows a SQL Server Enterprise Manager window with a query executed against the Northwind database. The query is: `SELECT productid, productname, unitprice FROM products;`. The results are displayed in a table with 17 rows. The first row is highlighted in blue.

	productid	productname	unitprice
1	1	Chai	18.0000
2	2	Chang	19.0000
3	3	Aniseed Syrup	10.0000
4	4	Chef Anton's Cajun Seasoning	22.0000
5	5	Chef Anton's Gumbo Mix	21.3500
6	6	Grandma's Boysenberry Spread	25.0000
7	7	Uncle Bob's Organic Dried Pears	30.0000
8	8	Northwoods Cranberry Sauce	40.0000
9	9	Mishi Kobe Niku	97.0000
10	10	Ikura	31.0000
11	11	Queso Cabrales	21.0000
12	12	Queso Manchego La Pastora	38.0000
13	13	Konbu	6.0000
14	14	Tofu	23.2500
15	15	Genen Shouyu	13.0000
16	16	Pavlova	17.4500
17	17	Alice Mutton	39.0000

SELECT ... WHERE

= specificatie van de voorwaarden waaraan de getoonde rijen moeten voldoen

Voorbeeld:

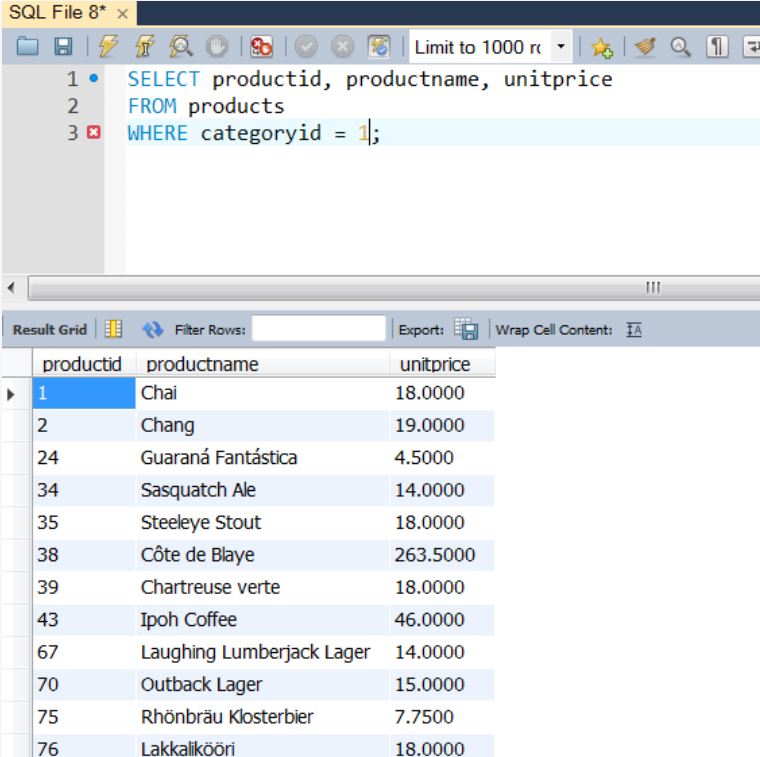
toon productID,

naam,

eenheidsprijs

van de producten die

behoren tot categorie 1



The screenshot shows a SQL IDE window titled 'SQL File 8* x'. The query editor contains the following SQL code:

```
1 • SELECT productid, productname, unitprice
2 FROM products
3 WHERE categoryid = 1;
```

Below the query editor is a 'Result Grid' showing the results of the query. The grid has three columns: 'productid', 'productname', and 'unitprice'. The first row is highlighted in blue.

productid	productname	unitprice
1	Chai	18.0000
2	Chang	19.0000
24	Guaraná Fantástica	4.5000
34	Sasquatch Ale	14.0000
35	Steeleye Stout	18.0000
38	Côte de Blaye	263.5000
39	Chartreuse verte	18.0000
43	Ipoh Coffee	46.0000
67	Laughing Lumberjack Lager	14.0000
70	Outback Lager	15.0000
75	Rhönbräu Klosterbier	7.7500
76	Lakkalikööri	18.0000

```
SELECT productid, productname, unitprice
FROM products
WHERE categoryid = 1
```

SELECT ... WHERE

- Gebruik van literals
 - Numerische waarden: ... WHERE categoryID = 1
 - Alfnumerische waarden: ... WHERE productName = 'Chai'
 - Datums: ... WHERE orderDate = '1996-07-04 00:00:00'
- Voorwaarden aan rijen opleggen
 - Vergelijkingsoperatoren
 - Wildcards
 - Logische operatoren
 - Een interval van specifieke waarden
 - Een lijst van waarden
 - Onbekende waarden
 - Je kan haakjes gebruiken om de prioriteitsregels te doorbreken of het geheel leesbaarder te maken

SELECT ... WHERE

- Vergelijkingsoperatoren

- =, >, >=, <, <=, <>

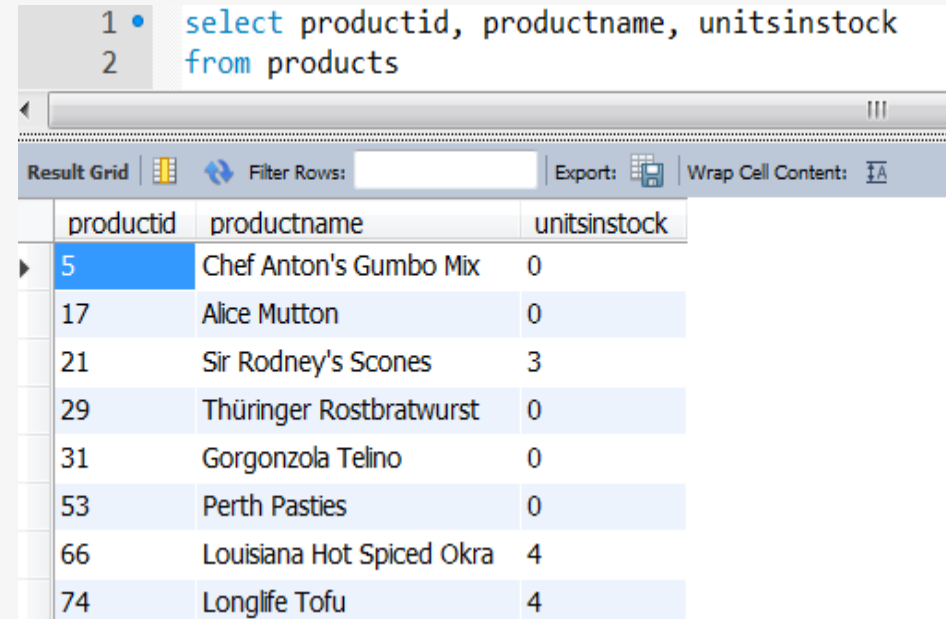
- Voorbeelden

- toon productID, naam, aantal in stock van de producten waarvan er minder dan 5 in stock zijn

```
select productid, productname, unitsinstock
from products
where unitsinstock < 5
```

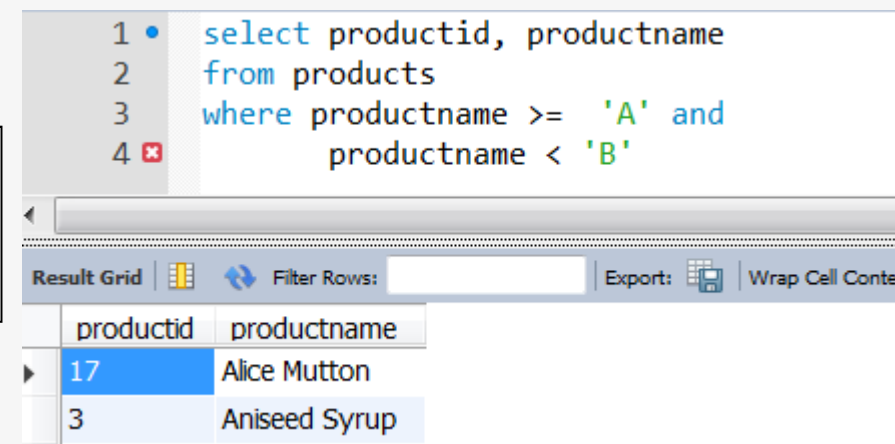
- toon productID, naam, aantal in stock van de producten waarvan naam begint met een A

```
select productid, productname
from products
where productname >= 'A' and productname < 'B'
```



```
1 • select productid, productname, unitsinstock
2 from products
```

	productid	productname	unitsinstock
▶	5	Chef Anton's Gumbo Mix	0
	17	Alice Mutton	0
	21	Sir Rodney's Scones	3
	29	Thüringer Rostbratwurst	0
	31	Gorgonzola Telino	0
	53	Perth Pasties	0
	66	Louisiana Hot Spiced Okra	4
	74	Longlife Tofu	4



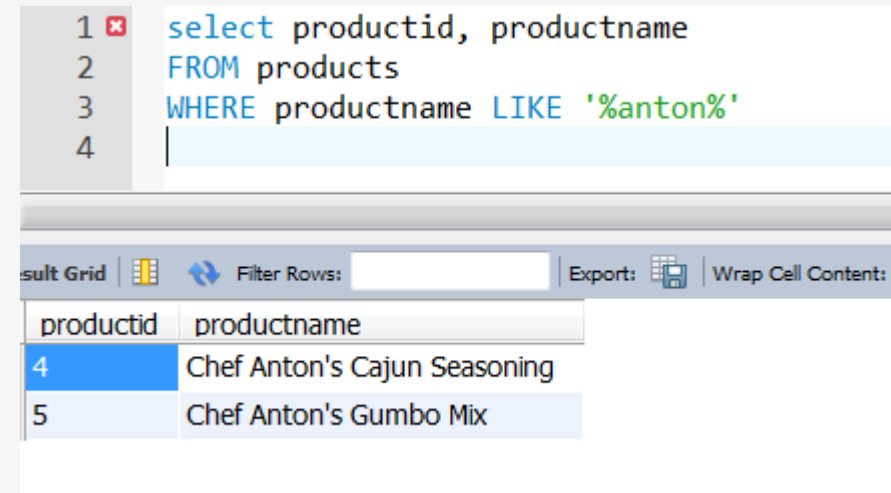
```
1 • select productid, productname
2 from products
3 where productname >= 'A' and
4   productname < 'B'
```

	productid	productname
▶	17	Alice Mutton
	3	Aniseed Syrup

SELECT ... WHERE

- Wildcards (zoeken naar patronen)
 - De operator **LIKE, NOT LIKE**
 - In combinatie met wildcards:
 - % willekeurige tekenrij met 0 of meerdere tekens
 - _ 1 teken
 - Voorbeeld: toon productID, naam van de producten waarbij de tekenreeks anton voorkomt in de naam

```
SELECT productid, productname  
FROM products  
WHERE productname LIKE '%anton%'
```



The screenshot shows a database query interface. The top part displays the SQL query: `select productid, productname FROM products WHERE productname LIKE '%anton%'`. Below the query, there is a table with two columns: `productid` and `productname`. The table contains two rows of data: `4` for `Chef Anton's Cajun Seasoning` and `5` for `Chef Anton's Gumbo Mix`. The interface also includes a 'Filter Rows' section and an 'Export' button.

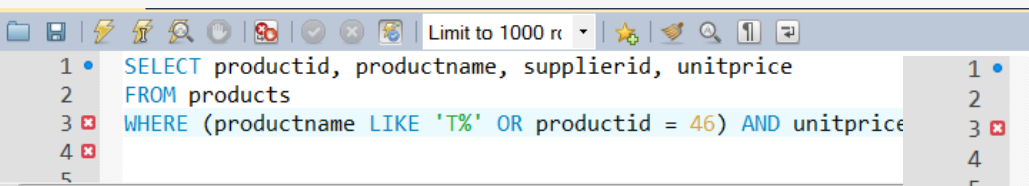
productid	productname
4	Chef Anton's Cajun Seasoning
5	Chef Anton's Gumbo Mix

SELECT ... WHERE

- Logische operatoren
 - **OR, AND, NOT** (volgens stijgende prioriteit)
 - Voorbeelden:




```
SELECT productid, productname, supplierid, unitprice
FROM products
WHERE (productname LIKE 'T%' OR productid = 46) AND unitprice > 16.00
```

```
SELECT productid, productname, unitprice
FROM products
WHERE (productname LIKE 'T%') OR (productid = 46 AND unitprice > 16.00)
```

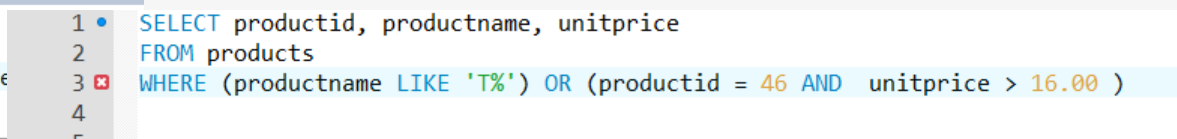


Limit to 1000 rows

```
1 • SELECT productid, productname, supplierid, unitprice
2 FROM products
3 WHERE (productname LIKE 'T%' OR productid = 46) AND unitprice > 16.00
4
5
```

Result Grid |  Filter Rows: | Export:  | Wrap Cell Content: 

	productid	productname	supplierid	unitprice
▶	14	Tofu	6	23.2500
	29	Thüringer Rostbratwurst	12	123.7900
	62	Tarte au sucre	29	49.3000



```
1 • SELECT productid, productname, unitprice
2 FROM products
3 WHERE (productname LIKE 'T%') OR (productid = 46 AND unitprice > 16.00 )
4
5
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	productid	productname	unitprice
14	Tofu	23.2500	
19	Teatime Chocolate Biscuits	9.2000	
23	Tunnbröd	9.0000	
29	Thüringer Rostbratwurst	123.7900	
54	Tourtière	7.4500	
62	Tarte au sucre	49.3000	

SELECT ... WHERE

- Waarden in een interval
 - **BETWEEN, NOT BETWEEN**
 - Voorbeeld: selecteer de producten (naam en eenheidsprijs) waarvan de eenheidsprijs tussen 10 en 15 euro (grenzen inbegrepen)

```
SELECT productid, unitprice  
FROM products  
WHERE unitprice BETWEEN 10 AND 15
```

productid	unitprice
3	10.0000
15	13.0000
21	10.0000
25	14.0000
31	12.5000
34	14.0000
42	14.0000
46	12.0000
48	12.7500
58	13.2500
67	14.0000
68	12.5000
70	15.0000
73	15.0000
74	10.0000
77	13.0000

SELECT ... WHERE

- Lijst van waarden
 - **IN, NOT IN**
 - Voorbeeld: geef productID, naam, supplierID van de producten die geleverd worden door de suppliers met ID 1, 3 of 5

```
SELECT productid, productname, supplierid  
FROM products  
WHERE supplierid in (1,3,5)
```

	productid	productname	supplierid
▶	2	Chang	1
	3	Aniseed Syrup	1
	6	Grandma's Boysenberry Spread	3
	7	Uncle Bob's Organic Dried Pears	3
	8	Northwoods Cranberry Sauce	3
	11	Queso Cabrales	5
	12	Queso Manchego La Pastora	5

SELECT ... WHERE

- Testen op onbekende waarden (niet ingevulde waarden)
 - **IS NULL, IS NOT NULL**
 - NULL waarden komen voor wanneer er bij input in een bepaalde kolom geen waarde werd ingebracht en er geen defaultwaarde voor die kolom voorzien was.
 - Een NULL waarde verschilt van 0 (numerische waarden) en blanco (character waarden)!
 - NULL velden worden onderling gelijk beschouwd (voor testen met DISTINCT)
 - Als in een rekenkundige uitdrukking een NULL-veld wordt verwerkt is het resultaat ook NULL
 - Voorbeeld: Selecteer de leveranciers waarvan region onbekend is

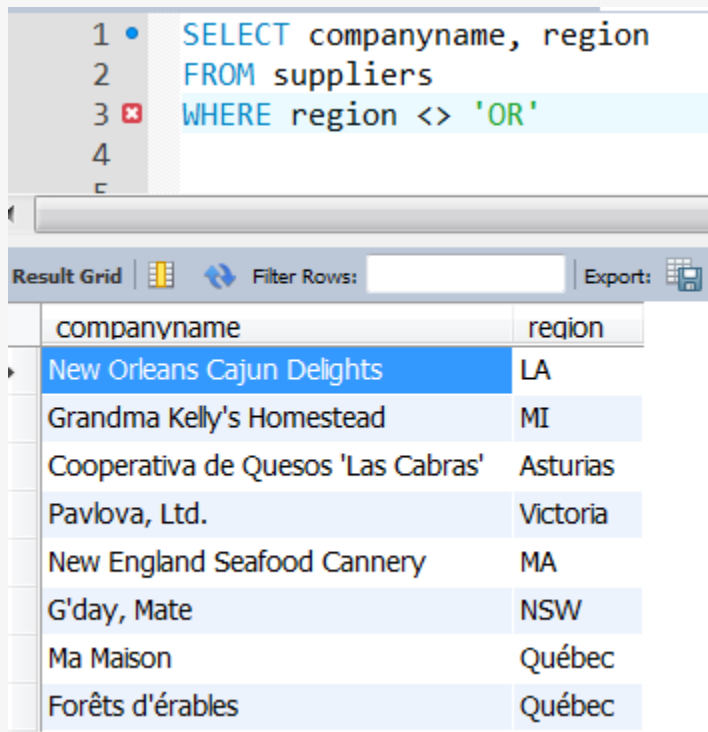
```
SELECT companyname, region
FROM suppliers
WHERE region IS NULL
```

companyname	region
Exotic Liquids	NULL
Tokyo Traders	NULL
Mayumi's	NULL
Specialty Biscuits, Ltd.	NULL
PB Knäckebröd AB	NULL
Refrescos Americanas LTDA	NULL
Heli Süßwaren GmbH & Co. KG	NULL
Plutzer Lebensmittelgroßmärkte AG	NULL
Nord-Ost-Fisch Handelsgesellschaft m...	NULL
Formaggi Fortini s.r.l.	NULL
Norske Meierier	NULL
	NULL

SELECT ... WHERE

- Let op met **NULL**!

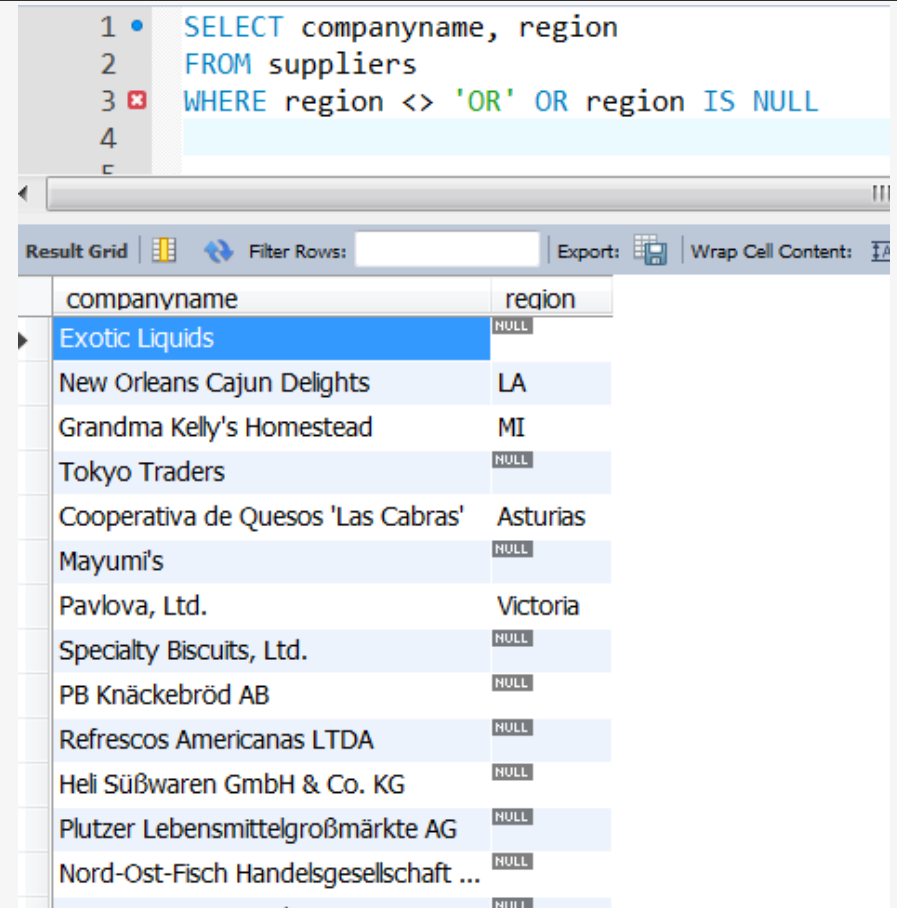
```
SELECT companyname, region
FROM suppliers
WHERE region <> 'OR'
```



```
1 • SELECT companyname, region
2 FROM suppliers
3 ✖ WHERE region <> 'OR'
4
```

companyname	region
New Orleans Cajun Delights	LA
Grandma Kelly's Homestead	MI
Cooperativa de Quesos 'Las Cabras'	Asturias
Pavlova, Ltd.	Victoria
New England Seafood Cannery	MA
G'day, Mate	NSW
Ma Maison	Québec
Forêts d'érables	Québec

```
SELECT companyname, region
FROM suppliers
WHERE region <> 'OR'
      OR region IS NULL
```



```
1 • SELECT companyname, region
2 FROM suppliers
3 ✖ WHERE region <> 'OR' OR region IS NULL
4
```

companyname	region
Exotic Liquids	NULL
New Orleans Cajun Delights	LA
Grandma Kelly's Homestead	MI
Tokyo Traders	NULL
Cooperativa de Quesos 'Las Cabras'	Asturias
Mayumi's	NULL
Pavlova, Ltd.	Victoria
Specialty Biscuits, Ltd.	NULL
PB Knäckebröd AB	NULL
Refrescos Americanas LTDA	NULL
Heli Süßwaren GmbH & Co. KG	NULL
Plutzer Lebensmittelgroßmärkte AG	NULL
Nord-Ost-Fisch Handelsgesellschaft ...	NULL

SELECT - enkele oefeningen

NR	VNAAM	INIT	FNAAM	AFD	IN DIENST	CODE	NIV	GESL	GEBDAT	SALARIS
10	Christine	I	Haas	A00	650101	66	18	V	330814	52750
20	Michel	L	Theunis	B01	731001	61	18	M	480202	41250
30	Sally	A	Kramer	C01	750405	60	20	V	410511	38250
50	Johan	B	Geysen	E01	490817	58	16	M	250915	40175
60	Irving	F	Steur	D11	730914	55	16	M	450707	32250
70	Eva	D	Pulanski	D21	800930	56	16	V	530526	36170
90	Evelien	W	Hendriks	E11	700815	55	16	V	410515	29750
100	Theo	Q	Spencer	E21	800619	54	14	M	561218	26150
110	Vincent	G	Leman	A00	631205	58	19	M	291105	46500
120	Sean		Connors	A00	580516	58	14	M	421018	29250
130	Danielle	M	Scheire	C01	710728	55	16	V	250915	23800
140	Hilde	A	Nagels	C01	761215	56	18	V	460119	28420
150	Bruno		Adams	D11	720212	55	16	M	470517	25280
160	Els	R	Placke	D11	771011	54	17	V	550412	22250
170	Mats	J	Sierens	D11	780915	54	16	M	510105	24680
180	Marleen	S	Schouters	D11	730707	53	17	V	490221	21340
190	Jan	E	Wauters	D11	740726	53	16	M	520625	20450
200	David		De Bruyn	D11	660303	55	16	M	410529	27740
210	Willem	T	Jansens	D11	790411	25	17	M	530223	18270
220	Jennifer	K	Luyckx	D11	680829	55	18	V	480319	29840

AFDNR	AFDNAAM	MANNR
A00	Computer	10
B01	Planning	20
C01	Informatie	30
D01	Ontwikkelingssc	50
E01	Support	60
D11	Administratie	70
D21	Software	80
E21	Tools	90

tabel Afdeling

tabel Werknemer

**HO
GENT**

SELECT - enkele oefeningen

- Geef voornaam en familienaam van werknemers met code 54, die in een willekeurige afdeling werken met uitsluiting van afdeling D11.
- Geef nummer, naam en afdelingsnummer van alle werknemers met salaris tussen 15000 en 24000 en niveau tussen 17 en 20.
- Geef nummer, naam en opleidingsniveau van alle werknemers met niveau 16, 18 of 20.
- Geef nummer, naam van vrouwelijke werknemers waarvan familienaam start met een 'S' of 'T'.
- Geef nummer, naam van alle werknemers met onbekende jobcode.
- Geef nummer, naam en afdelingsnummer van alle werknemers, waarvan de familienaam start met een P en die in een afdeling werken beginnend met D en als 3^o karakter een 1 hebben.

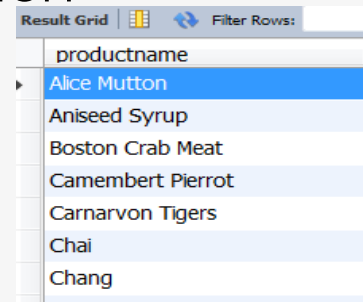
SELECT - formateren resultaten

- Formatteren van de resultaten
 - Sorteren data
 - Eliminatie van duplicaten
 - Wijzigen van kolomnaam
 - Berekenende resultaatkolommen
- Commentaar
 - /* commentaar */
 - -- commentaar (beperkt zich tot 1 lijn)

SELECT - ORDER BY

- Sorteren van de data
 - **ORDER BY** clause
 - Kan 1 of meerdere sorteervelden bevatten
 - Een sorteerveld kan gespecificeerd worden via de kolomnaam, of door het volgnummer op te geven dat overeenkomt met de volgorde van het gegeven achter de SELECT clause (startend vanaf 1)
 - Indien meerdere sorteervelden voorkomen, gebeurt het sorteren eerst op basis van het eerste veld, bij gelijkheid op basis van het tweede,...
 - Standaard gebeurt het sorteren in stijgende volgorde (volgens numerieke waarde, of volgens computercode bvb ASCII). Een dalende volgorde moet expliciet vermeld worden met DESC
 - Voorbeeld: toon een alfabetische lijst van de productnamen

```
SELECT productname  
FROM products  
ORDER BY productname    -- of ORDER BY 1
```



The screenshot shows a 'Result Grid' window with a 'Filter Rows' button. The grid contains a single column named 'productname'. The data is sorted alphabetically, with 'Alice Mutton' selected as the first row. Other visible rows include 'Aniseed Syrup', 'Boston Crab Meat', 'Camembert Pierrot', 'Carnarvon Tigers', 'Chai', and 'Chang'.

productname
Alice Mutton
Aniseed Syrup
Boston Crab Meat
Camembert Pierrot
Carnarvon Tigers
Chai
Chang

SELECT - formateren resultaten

- Voorbeeld: toon productid, naam, categoryid en eenheidsprijs van de producten gesorteerd op categoryid. Indien binnen 1 categorie producten dezelfde prijs hebben, dan dient het product met de hoogste prijs bovenaan te staan.

```
SELECT productid, productname, categoryid, unitprice  
FROM products  
ORDER BY categoryid, unitprice DESC
```

```
1 • SELECT productid, productname, categoryid, unitprice  
2 FROM products  
3 ORDER BY categoryid, unitprice DESC  
4  
5
```

Result Grid Filter Rows: Export: Wrap Cell Content:			
productid	productname	categoryid	unitprice
38	Côte de Blaye	1	263.5000
43	Ipoh Coffee	1	46.0000
2	Chang	1	19.0000
1	Chai	1	18.0000
39	Chartreuse verte	1	18.0000
76	Lakkalikööri	1	18.0000
35	Steeleye Stout	1	18.0000
70	Outback Lager	1	15.0000
34	Sasquatch Ale	1	14.0000
67	Laughing Lumberjack Lager	1	14.0000
75	Rhönbräu Klosterbier	1	7.7500
24	Guaraná Fantástica	1	4.5000
63	Vegie-spread	2	43.9000

SELECT - DISTINCT/ALL

- Uniciteit van de rijen
 - **DISTINCT** produceert een lijst waar alle rijen uniek zijn, gelijke rijen worden uit resultaat verwijderd
 - **ALL** (default) toont alle rijen, ook duplicaten
 - Voorbeeld: toon de leveranciers die producten leveren

```
SELECT supplierid  
FROM products  
ORDER BY supplierid
```



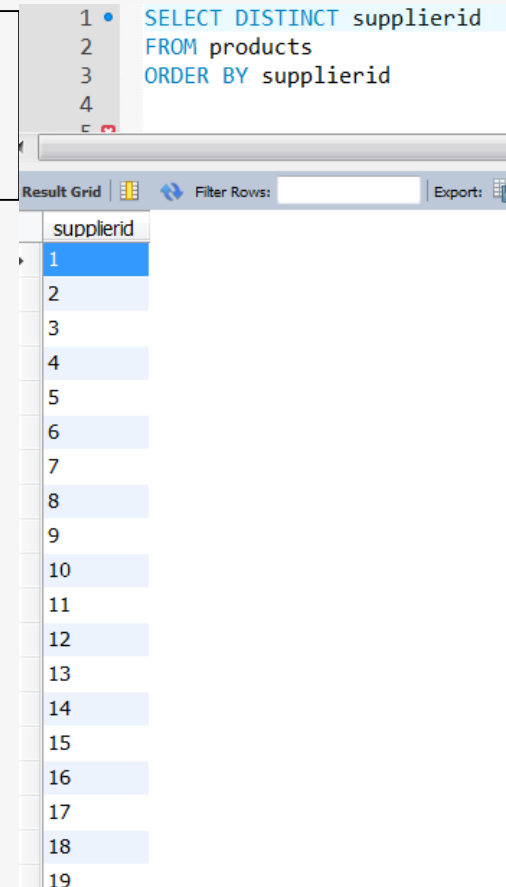
The screenshot shows a SQL query editor with the following code:

```
1 • SELECT supplierid  
2 FROM products  
3 ORDER BY supplierid  
4
```

Below the editor is a 'Result Grid' with a 'Filter Rows' field. The grid displays the 'supplierid' column with the following values (row 1 is highlighted):

supplierid
1
1
2
2
2
2
3
3
3
4
4
4
5
5
6
6
6
7
7
7

```
SELECT DISTINCT supplierid  
FROM products  
ORDER BY supplierid
```



The screenshot shows a SQL query editor with the following code:

```
1 • SELECT DISTINCT supplierid  
2 FROM products  
3 ORDER BY supplierid  
4
```

Below the editor is a 'Result Grid' with a 'Filter Rows' field and an 'Export' button. The grid displays the 'supplierid' column with the following values (row 1 is highlighted):

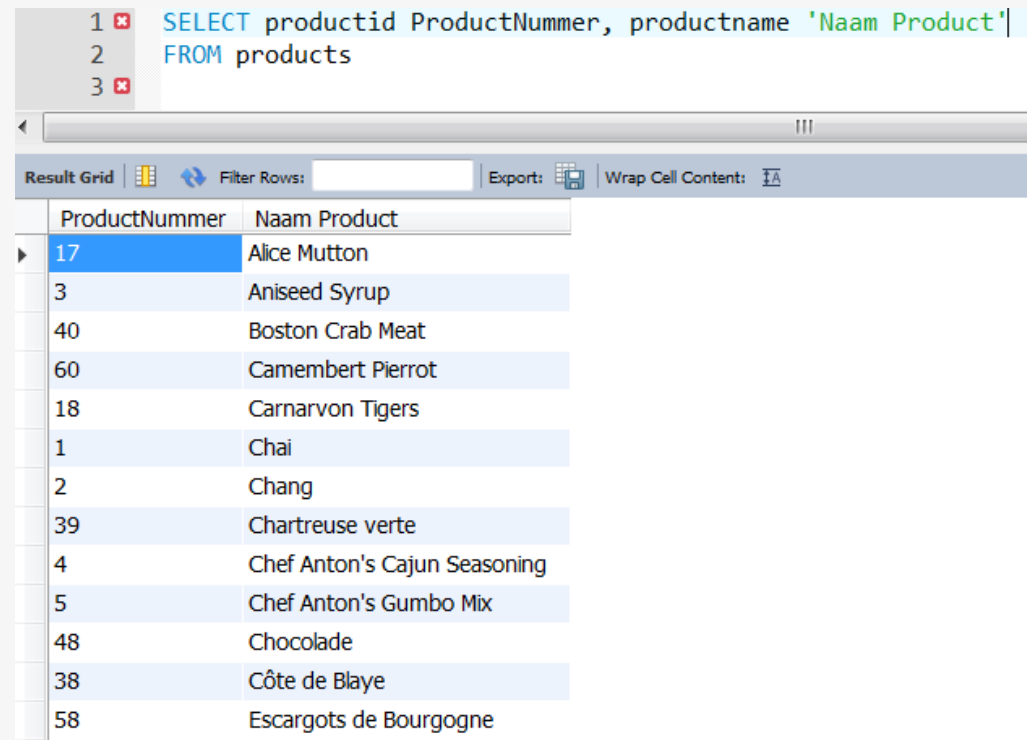
supplierid
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19

SELECT en gebruik van aliases

- Benoemen van kolommen
 - Standaard : kolomtitel = naam van kolom in tabel; berekende kolommen krijgen geen kolomnaam
 - Via **AS** keyword kan je een kolom een andere titel geven, of door opgave van kolom, dan een spatie en vervolgens de uitdrukking
 - Opm. Die nieuwe kolomnaam kan je ook gebruiken in ORDER BY (niet in WHERE, HAVING, GROUP BY)
 - Voorbeeld: selecteer ProductID en ProductNaam van de producten en geef als kolomtitel ProductNummer en Naam Product.

```
SELECT productid AS ProductNummer,  
       productname AS 'Naam Product'  
FROM products
```

```
SELECT productid ProductNummer,  
       productname 'Naam Product'  
FROM products
```



The screenshot shows a database query interface. At the top, a SQL query is entered in a text area:

```
1 SELECT productid ProductNummer, productname 'Naam Product'  
2 FROM products  
3
```

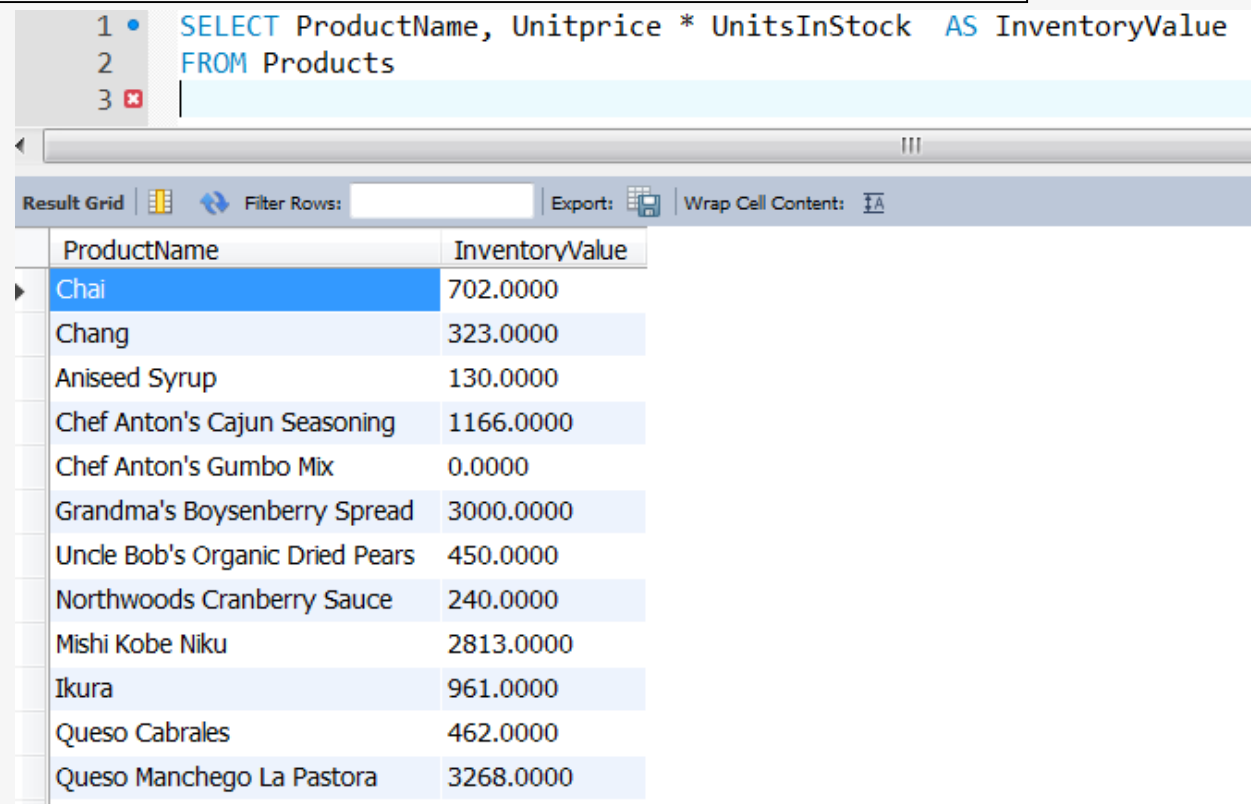
Below the query area is a toolbar with options like 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. The main area displays a table with the results of the query:

ProductNummer	Naam Product
17	Alice Mutton
3	Aniseed Syrup
40	Boston Crab Meat
60	Camembert Pierrot
18	Carnarvon Tigers
1	Chai
2	Chang
39	Chartreuse verte
4	Chef Anton's Cajun Seasoning
5	Chef Anton's Gumbo Mix
48	Chocolate
38	Côte de Blaye
58	Escargots de Bourgogne

SELECT met berekende resultaten

- Berekende resultaatkolommen
 - Via wiskundige operatoren : +, -, /, *
 - Voorbeeld : geef naam en inventariswaarde van de producten

```
SELECT ProductName, Unitprice * UnitsInStock AS InventoryValue  
FROM Products
```



The screenshot shows a SQL query editor with the following query:

```
1 • SELECT ProductName, Unitprice * UnitsInStock AS InventoryValue  
2 FROM Products  
3 ✖
```

Below the query editor is a result grid with the following columns: ProductName and InventoryValue. The grid displays 14 rows of data, with the first row 'Chai' highlighted in blue.

ProductName	InventoryValue
Chai	702.0000
Chang	323.0000
Aniseed Syrup	130.0000
Chef Anton's Cajun Seasoning	1166.0000
Chef Anton's Gumbo Mix	0.0000
Grandma's Boysenberry Spread	3000.0000
Uncle Bob's Organic Dried Pears	450.0000
Northwoods Cranberry Sauce	240.0000
Mishi Kobe Niku	2813.0000
Ikura	961.0000
Queso Cabrales	462.0000
Queso Manchego La Pastora	3268.0000

SELECT en gebruik van functies

- Functies
 - **String** functies: concat, left, right, length, substring, replace, ...
 - **DateTime** functies: dateAdd, dateDiff, day, month, year, ...
 - NOW(): functie die huidige datum en tijd retourneert in DATETIME format: YYYY-MM-DD HH:MM:SS.
 - SELECT CURDATE() → huidige datum YYYY-MM-DD
 - **Rekenkundige** functies: round, floor, ceiling, cos, sin, ...
 - **Aggregate** functies: AVG, SUM, ...
 - **IFNULL**: vervangt NULL waarde met opgegeven waarde
Voorbeeld: SELECT IFNULL(unitprice, 10.00) FROM products

SELECT en data conversie

- Impliciet
 - mogelijke voor sommige conversies
 - Voorbeeld: `UnitPrice * 0,5`
UnitPrice (integer) wordt automatisch naar decimal geconverteerd
- Expliciet
 - **CAST** (<value expression> AS <data type>)
 - Voorbeeld: `PRINT CAST(-25.25 AS INTEGER) -> -25`
`SELECT CAST(38,8 AS CHAR) → '38,8'`
 - **CONVERT** (<expression>, type)
 - Voorbeeld: `CONVERT(curdate(), char) -> '2016-11-24'`

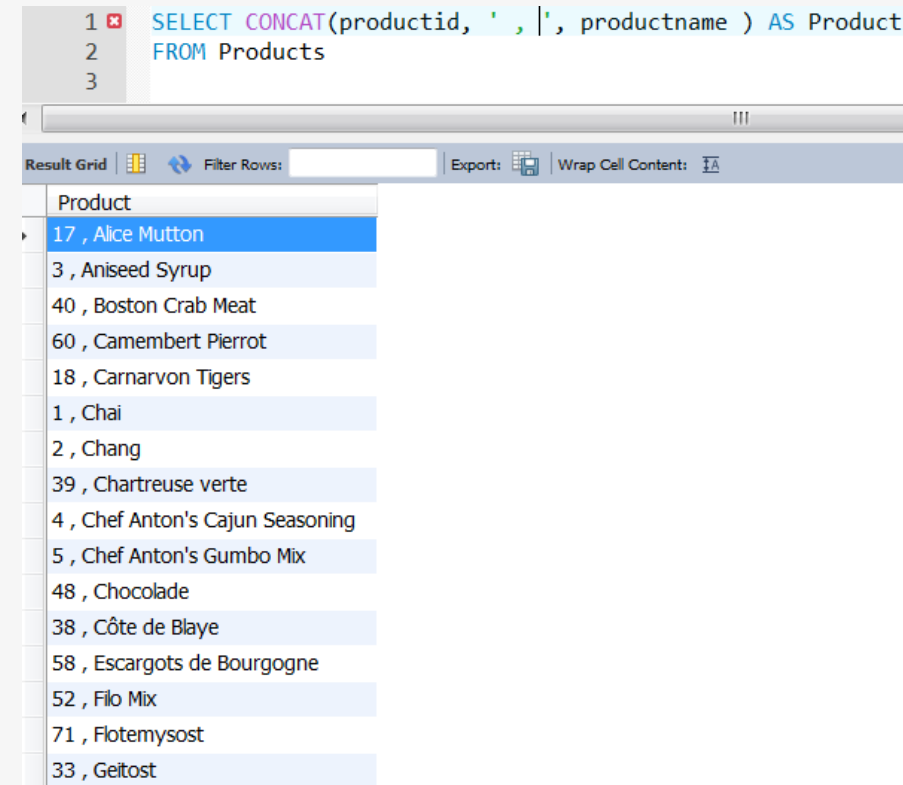
SELECT en strings

- Stringoperator: concatenatie

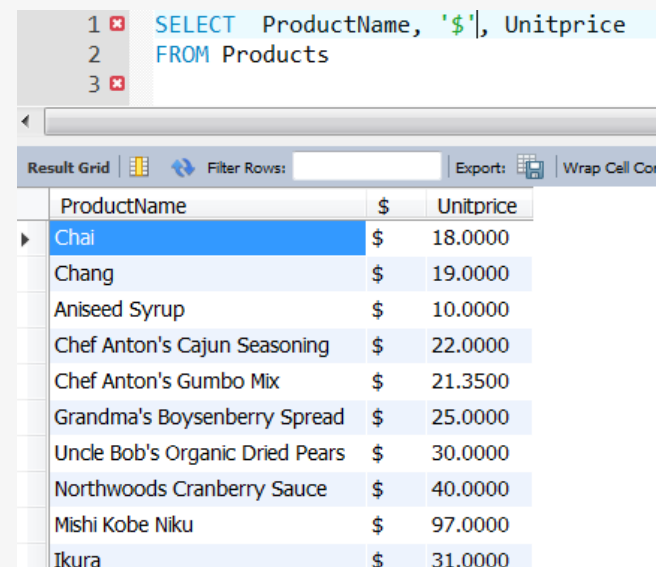
```
SELECT CONCAT(productid, ' ', productname )  
AS Product  
FROM Products
```

- Gebruik maken van tekst (literals)

```
SELECT ProductName, '$', Unitprice  
FROM Products
```



Product
17 , Alice Mutton
3 , Aniseed Syrup
40 , Boston Crab Meat
60 , Camembert Pierrot
18 , Carnarvon Tigers
1 , Chai
2 , Chang
39 , Chartreuse verte
4 , Chef Anton's Cajun Seasoning
5 , Chef Anton's Gumbo Mix
48 , Chocolate
38 , Côte de Blaye
58 , Escargots de Bourgogne
52 , Filo Mix
71 , Flotemysost
33 , Geitost



ProductName	\$	Unitprice
Chai	\$	18.0000
Chang	\$	19.0000
Aniseed Syrup	\$	10.0000
Chef Anton's Cajun Seasoning	\$	22.0000
Chef Anton's Gumbo Mix	\$	21.3500
Grandma's Boysenberry Spread	\$	25.0000
Uncle Bob's Organic Dried Pears	\$	30.0000
Northwoods Cranberry Sauce	\$	40.0000
Mishi Kobe Niku	\$	97.0000
Ikura	\$	31.0000

SELECT en de CASE functie

- Voorbeeld:

```
SELECT
```

```
  CASE
```

```
    WHEN price IS NULL THEN 'Not yet priced'
```

```
    WHEN price < 10 THEN 'Very Reasonable Title'
```

```
    WHEN price >= 10 and price < 20 THEN 'Coffee Table Title'
```

```
    ELSE 'Expensive book!'
```

```
  END AS 'Price Category',
```

```
  CONVERT(title, char(20)) AS 'Shortened Title'
```

```
FROM titles
```

```
ORDER BY price
```

Price Category	Shortened Title
Very Reasonable Title	You Can Combat Compu
Very Reasonable Title	The Gourmet Microwav
Very Reasonable Title	Life Without Fear
Very Reasonable Title	Emotional Security:
Coffee Table Title	Is Anger the Enemy?
Coffee Table Title	Cooking with Compute
Coffee Table Title	Fifty Years in Bucki
Coffee Table Title	Sushi, Anyone?
Coffee Table Title	The Busy Executive's
Coffee Table Title	Straight Talk About
Coffee Table Title	Silicon Valley Gastr
Coffee Table Title	Prolonged Data Depri
Expensive book!	Secrets of Silicon V
Expensive book!	Onions, Leeks, and G
Expensive book!	Computer Phobic AND