Databases

Deel 5 - SELECT instructie WHERE

AP HOGESCHOOL ANTWERPEN

AP.BE

Topics

- De WHERE component
  - Condities
  - Vergelijkingsoperatoren
  - Condities koppelen met AND en OR
- SQL Operatoren voor condities
  - IN
  - BETWEEN
  - LIKE
  - IS NULL

WHERE component

Basics

AP HOGESCHOOL ANTWERPEN

AP.BE

De WHERE component

Doel

- Op basis van 1 of meerdere condities rijen filteren uit het tussenresultaat

De WHERE component

Instructie

```
SELECT [DISTINCT] select_expr [, select_expr] ... [into_option]
[FROM table_references]
[WHERE where_condition]
[GROUP BY {col_name | position}]
[HAVING where_condition]
[ORDER BY {col_name | position}
[ASC | DESC], ...]
```

De WHERE component

⌨ Syntax

```
WHERE where_condition
```

- **`WHERE`**:
  - Keyword om aan te geven dat hierna de filterconditie(s) komen
- ***`where_condition`***:
  - Minstens 1 logische conditie
  - Logische conditie kan resulteren in
    - WAAR ⇨ Rij wordt weerhouden uit het tussenresultaat
    - ONWAAR ⇨ Rij wordt verwijderd uit het tussenresultaat
    - ONBEPAALD ⇨ Rij wordt verwijderd uit het tussenresultaat

De WHERE component

Theorie

- Opbouw logische conditie (predikaat)
  - Scalaire expressie
  - Vergelijkingsoperator → (= | > | < | <= | >= | <>)
  - Scalaire expressie

- Scalaire expressie
  - Constante waarde (bv.: 1, 'Kast', '2020-10-31', 12.5)
  - Naam van een kolom (resultaat in waarde van kolom voor huidige rij)
  - Resultaat van een functie of andere query zolang dit 1 enkele waarde oplevert
  - NULL (resulteert in ONBEPAALD als resultaat)

De WHERE component

Voorbeelden

```
SELECT
    *
FROM
    `tblPlatforms`
WHERE
    `name` = 'PS4';
```

```
SELECT
    *
FROM
    `tblPlatforms`
WHERE
    `name` <> 'PS5';
```

```
SELECT
    *
FROM
    `tblPlatforms`
WHERE
    `id` < 15;
```

```
SELECT
    *
FROM
    `tblGameDefinitions`
WHERE
    YEAR(`releasedatum`) >= 2020;
```

De WHERE component

Voorbeelden

```
SELECT `Naam`,`Fabrikant`
FROM `tblConsoles`
WHERE `Fabrikant` = 'Sony';
```

1

id	naam	fabrikant	releasedatum	eindeproduktie	prijs
1	PS3	Sony	2006-11-06 00:00:00	NULL	399.00
2	PS4	Sony	2013-11-13 00:00:00	NULL	399.00
3	XBOX 360	Microsoft	2005-11-16 00:00:00	NULL	300.00
4	XBOX One	Microsoft	2013-11-22 00:00:00	NULL	499.00
5	Nintendo Wii	Nintendo	2006-11-19 00:00:00	2013-10-20 00:00:00	249.00
6	PS	Sony	1994-12-03 00:00:00	2006-03-31 00:00:00	299.00
7	PS2	Sony	2000-03-04 00:00:00	2012-12-28 00:00:00	299.00
8	DreamCast	Sega	1998-11-27 00:00:00	2001-03-31 00:00:00	199.00
9	Super Nintendo	Nintendo	1990-11-21 00:00:00	2003-09-25 00:00:00	199.00

De WHERE component

Voorbeelden

```
SELECT `Naam`,`Fabrikant`
FROM `tblConsoles`
WHERE `Fabrikant` = 'Sony';
```

2

	id	naam	fabrikant	releasedatum	eindeproduktie	prijs	
▶	1	PS3	Sony	2006-11-06 00:00:00	NULL	399.00	WAAR
	2	PS4	Sony	2013-11-13 00:00:00	NULL	399.00	WAAR
							ONWAAR
							ONWAAR
							ONWAAR
	6	PS	Sony	1994-12-03 00:00:00	2006-03-31 00:00:00	299.00	WAAR
	7	PS2	Sony	2000-03-04 00:00:00	2012-12-28 00:00:00	299.00	WAAR
							ONWAAR
							ONWAAR

De WHERE component

Voorbeelden

```
SELECT `Naam`,`Fabrikant` 3
FROM `tblConsoles`
WHERE `Fabrikant` = 'Sony';
```

id	naam	fabrikant	releasedatum	eindeproduktie	prijs
1	PS3	Sony	2006-11-06 00:00:00	NULL	399.00
2	PS4	Sony	2013-11-13 00:00:00	NULL	399.00
6	PS	Sony	1994-12-03 00:00:00	2006-03-31 00:00:00	299.00
7	PS2	Sony	2000-03-04 00:00:00	2012-12-28 00:00:00	299.00

Naam	Fabrikant
PS3	Sony
PS4	Sony
PS	Sony
PS2	Sony

De WHERE component

Theorie

- Condities combineren
  - Met de logische AND ⇨ WAAR wanneer beide condities WAAR opleveren
  - Met de logische OR ⇨ WAAR wanneer een van de condities WAAR oplevert
  - Met de logische NOT ⇨ Resultaat wordt geïnverteerd (Behalve bij onbepaald)

C1 (Conditie 1)	C2 (Conditie 2)	C1 AND C2	C1 OR C2	NOT C1
WAAR	WAAR	WAAR	WAAR	ONWAAR
WAAR	ONWAAR	ONWAAR	WAAR	ONWAAR
Onbepaald	WAAR	Onbepaald	WAAR	Onbepaald
Onbepaald	ONWAAR	ONWAAR	Onbepaald	Onbepaald

De WHERE component

Theorie

- AND condities worden eerst geëvalueerd
- Gebruik haakjes om de logische volgorde van verwerking in te stellen! (Analoog aan wiskunde)

De WHERE component

⚠ Aandachtspunten

- Bij meerdere condities voor eenzelfde kolom, moet deze per conditie herhaald worden!

```sql
SELECT
    *
FROM
    `tblPlatforms`
WHERE
    `name` = 'PS4'
    OR 'PS5';
```

```sql
SELECT
    *
FROM
    `tblPlatforms`
WHERE
    `name` = 'PS4'
    OR `name` = 'PS5';
```

De WHERE component

⚠ Aandachtspunten

- Opletten bij conditie met vergelijksoperator <> wanneer waarde NULL is

	spelersnr	bondsnr
▶	2	2411
	6	8467
	7	NULL
	8	2983
	27	2513
	28	NULL
	39	NULL
	44	1124
	57	6409
	83	1608
	95	NULL
	100	6524
	104	7060
	112	1319

```
SELECT
    `SPELERSNR`,`BONDSNR`
FROM
    `SPELERS`
WHERE
    `BONDSNR` <> '7060';
```

	spelersnr	bondsnr
▶	2	2411
	6	8467
	8	2983
	27	2513
	44	1124
	57	6409
	83	1608
	100	6524
	112	1319

Oefeningen

Deel 5 - Oefenbundel 1

- Surf naar Digitap
- Download de opgave
- Los de vragen op

WHERE component

SQL Operatoren voor condities

AP HOGESCHOOL ANTWERPEN

AP.BE

De IN operator

Doel

- Rijen filteren op basis van een lijst van scalaire waarden

De IN operator

Syntax

```
WHERE <expressie> [NOT] IN <expressielijst>
```

- ***expressie:***
  - De kolom (of waarde) waarvoor je wenst te valideren of die in de lijst voorkomt
- ***expressielijst:***
  - Een lijst van scalaire waarden
  - Geplaatst tussen haakjes
  - Gescheiden met een ,

De IN operator

Theorie

- Vervangt x-aantal OR combinaties
- Werking
  - MySQL zal eerst de waarden sorteren
  - Daarna zoeken tot aan eerste match (via binair zoekalgoritme)
  - Wanneer match gevonden is wordt WAAR teruggegeven
- Wanneer expressie of element in expressielijst NULL is wordt resultaat ook NULL
- Combineerbaar met de NOT operator

De IN operator

Voorbeelden

```
SELECT
    *
FROM
    `tblGameDefinitions`
WHERE
    `id` = 1 OR
    `id` = 5 OR
    `id` = 8 OR
    `id` = 9;
```



```
SELECT
    *
FROM
    `tblGameDefinitions`
WHERE
    `id` IN (1,5,8,9);
```

De IN operator

Voorbeelden

```
SELECT
    *
FROM
    `tblGameDefinitions`
WHERE
    `id` NOT IN (1,5,8,9);
```

Geeft als resultaat:

- Alle records uit tabel tblGameDefinitions
- Waarvan de waarde van het veld `id` **NIET** voorkomt in de gegeven expressielijst

Oefeningen

Deel 5 - Oefenbundel 2

- Surf naar Digitap
- Download de opgave
- Los de vragen op

De BETWEEN operator

Doel

- Rijen filteren op basis van een boven- en ondergrenswaarde

De BETWEEN operator

⌨ Syntax

```
WHERE <expressie> [NOT] BETWEEN <expressie min> AND <expressie max>
```

- ***expressie:***
  - De kolom (of waarde) waarvoor je wenst te valideren of die tussen de grenswaarden ligt
- ***expressie min:***
  - De ondergrens (kleinste waarde)
- ***expressie max:***
  - De bovengrens (grootste waarde)

De BETWEEN operator

Theorie

- Conditie geeft WAAR terug wanneer waarde van expressie:
  - Groter dan of gelijk aan ondergrenswaarde is
    EN
  - Kleiner dan of gelijk aan bovengrenswaarde is
- Werkt met (gehele) getallen, datum- en tijdwaarden, alfanumerieke waarden
- Wanneer een expressie NULL is wordt resultaat ook NULL
- Combineerbaar met de NOT operator

De BETWEEN operator

Voorbeelden

```
SELECT
    *
FROM
    `tblGameDefinitions`
WHERE
    `id` BETWEEN 1 AND 10;
```

```
SELECT
    *
FROM
    `tblPlatforms`
WHERE
    YEAR(`release`) BETWEEN 2010 AND 2020;
```

Oefeningen

Deel 5 - Oefenbundel 3

- Surf naar Digitap
- Download de opgave
- Los de vragen op

De LIKE operator

Doel

- Rijen filteren op basis van vergelijking met een alfanumeriek patroon

De LIKE operator

⌨ Syntax

```
WHERE <expressie> [NOT] LIKE <scalaire alfanumerieke expressie>
```

- ***expressie:***
  - De kolom (of waarde) waarvoor je wenst te valideren of die voldoet aan het patroon
- ***scalaire alfanumerieke expressie:***
  - Een alfanumeriek patroon waarop we zoeken
  - Kan jokertekens (wildcards) bevatten
    - % ⇨ geen, één of meer willekeurige tekens
    - _ ⇨ exact één willekeurig teken

De LIKE operator

Theorie

- Eenvoudig patroon bouwen gebruik makend van jokertekens
- Elke rij waarvoor de zoekwaarde overeenkomt met het patroon wordt weerhouden
- ESCAPE karakter gebruiken wanneer letterlijk op een jokerteken wordt gezocht
  - Standaard ESCAPE karakter is \
  - (Aanpasbaar met ESCAPE keyword na patroon)
- Combineerbaar met de NOT operator
- Wanneer een expressie NULL is wordt resultaat ook NULL

De LIKE operator

Voorbeelden

```
SELECT
    *
FROM
    `tblPlatforms`
WHERE
    `Uitgever` LIKE 'S%';
```

Haalt alle platformen op waarvan de waarde van kolom uitgever:

- Begint met een S
- Gevolgd door >= 0 willekeurige karakters

De LIKE operator

Voorbeelden

```
SELECT
    *
FROM
    `tblPlatforms`
WHERE
    `Uitgever` LIKE '%t';
```

Haalt alle platformen op waarvan de waarde van kolom uitgever:

- Eindigt op een t
- Voorafgegaan door >= 0 willekeurige karakters

De LIKE operator

Voorbeelden

```
SELECT
    *
FROM
    `tblPlatforms`
WHERE
    `Naam` LIKE 'PS _';
```

Haalt alle platformen op waarvan de waarde van kolom Naam:

- Begint met ‘PS ‘
- En eindigt met 1 willekeurig karakter
- Exact 4 karakters lang is

De LIKE operator

⚠ Aandachtspunten

- Denk aan **collation** bij gebruik van LIKE! ⇨ ‘a’ = ‘A’ ?

Oefeningen

Deel 5 - Oefenbundel 4

- Surf naar Digitap
- Download de opgave
- Los de vragen op

De IS NULL operator

Doel

- Rijen filteren waarvoor de waarde van een kolom NULL (onbekend) is

De IS NULL operator

Syntax

```
WHERE <expressie> IS [NOT] NULL
```

- ***expressie:***
  - De kolom (of waarde) waarvoor je wenst te valideren of die NULL is

De IS NULL operator

Theorie

- Conditie geeft WAAR terug wanneer de waarde van een kolom NULL is
- Combineerbaar met de NOT operator

De IS NULL operator

Voorbeelden

```
SELECT
    *
FROM
    `tblPlatforms`
WHERE
    `Eindeproductie` IS NULL;
```

Haalt alle platformen op waarvan de waarde van kolom `Eindeproductie` NULL (onbekend) is

De IS NULL operator

Voorbeelden

```
SELECT
    *
FROM
    `tblPlatforms`
WHERE
    `Eindeproductie` IS NOT NULL;
```

Haalt alle platformen op waarvan de waarde van kolom `Eindeproductie` niet NULL (gekend) is

Oefeningen

Deel 5 - Oefenbundel 5

- Surf naar Digitap
- Download de opgave
- Los de vragen op