**DATABASEBEHEER |** HOOFDSTUK 4

**INLEIDING SQL**

W. VAN BEECK

# Leerstofoverzicht

|  |  |
| --- | --- |
|  | Heb je een vraag of opmerkingen over de leerstof die in deze leerstofbundel aan bod komt? Aarzel dan niet om contact met de leerkracht op te nemen via de rubriek ‘hulp nodig’ op Moodle. |

|  |  |  |  |
| --- | --- | --- | --- |
|  | | WAT MOET JE **KENNEN**? | **Herhaald?** |
| 1 | De cursist kan de begrippen **SQL** en **query** en **CRUD** in eigen woorden toelichten. | |  |
| 2 | De cursist kent het belang van backticks (`). | |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | | WAT MOET JE **KUNNEN**? | **Herhaald?** |
| 1 | De cursist kan de structuur van een database visualiseren aan de hand van een EER diagram. | |  |
| 2 | De cursist kan alle of een selectie velden van een tabel opvragen. | |  |
| 3 | De cursist kan een enkelvoudige en meervoudige commentaarregel aan de webpagina toevoegen. | |  |
| 4 | De cursist kan databasegegevens filteren op basis van   * Een overeenkomst * Een patroon * Een waarde die kleiner/groter is * Een waarde die (niet) in een gegeven bereik ligt * Meerdere voorwaarden (AND/OR) | |  |
| 5 | De cursist kan werken met veldaliassen. | |  |

# Voorbereiding

## Word-document bewaren

|  |  |  |
| --- | --- | --- |
|  |  | 1. Maak de map ‘documenten’ > ‘databasebeheer’ > ‘hoofdstuk\_4\_inleiding\_SQL’ aan. |
|  | 1. Bewaar dit Word-document in de bovengenoemde map. De oplossingen van de oefeningen die in deze bundel aan bod komen moeten in dit document verwerkt worden. Na het afwerken van dit hoofdstuk moet de aangevulde versie van dit document onder de daarvoor voorziene opdracht op Moodle worden ingeleverd. |

## Databanken importeren

De oefeningen in dit hoofdstuk doen beroep op twee nieuwe databanken: ‘zoo’ & ‘digitech’. De SQL-scripts waarmee je beide databanken kan importeren kan je op Moodle terugvinden, alsook het EER-diagram van beide databanken.

**De oefeningen in dit hoofdstuk zijn onderverdeeld in twee groepen:**

* De groene oefeningen (die beroep doen op de database ‘zoo’) beschikken over een verbetersleutel. Probeer deze oefeningen eerst zelf uit en verbeter ze vervolgens aan de hand van de verbetersleutel. Lever enkel de verbeterde versie van je oefeningen in! Het maken van deze oefeningen geeft je toegang tot het volgende hoofdstuk.
* De roze oefeningen (die beroep doen op de database ‘digitech’) beschikken niet over een verbetersleutel. Deze oefeningen worden door de leerkracht nagekeken en maken deel uit van de permanente evaluatie (staan op punten).

Het is m.a.w. belangrijk dat je **alle oefeningen** volledig maakt.

|  |  |
| --- | --- |
|  | De voorbeelden in deze afstandsbundel maken gebruik van de database ‘cvo\_volt’. Deze databank heb je in het vorige hoofdstuk reeds geïmporteerd in MySQL Workbench. |

**Database ‘cvo\_volt’:**

**Afbeelding met tekst, schermopname, Lettertype, nummer

Automatisch gegenereerde beschrijving**

# Queries

Een query is een opdracht die via de codetaal SQL (Structured Query Language) aan het RDBMS (relationel database management system), lees: ‘MySQL’, wordt doorgegeven om een bepaalde actie uit te voeren. De belangrijkste acties kunnen volgens het CRUD-principe worden opgedeeld in 4 verschillende groepen.

* **C**REATE: het toevoegen van nieuwe gegevens (insert)
* **R**EAD: het opvragen van nieuwe gegevens (select)
* **U**PDATE: het wijzigen van gegevens (update);
* **D**ELETE: verwijderen van gegevens (delete).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **INSTRUCTIEVIDEO** | **QUERIES UITWERKEN EN TESTEN IN MYSQL WORKBENCH** |  | 15 min |
| Klik [hier](https://www.youtube.com/watch?v=kCRRZhT-jtU) om de video te bekijken. (houd de CTRL-toets ingedrukt om de link te openen) | | |  |  |

# Een selectie van gegevens opvragen

Wanneer je gegevens uit een database wenst op te vragen, moet je gebruik maken van het sleutelwoord SELECT.

|  |  |
| --- | --- |
|  | **SYNTAX** |
| **Alle velden uit een tabel opvragen** | SELECT\* FROM **`**tabelnaam**`**;  *Met de asterisk (\*) geef je aan dat je alle velden wenst op te vragen uit de opgegeven tabel.* |
| **Een selectie van velden uit een tabel opvragen** | SELECT`veldnaam1`, `veldnaam2`, `veldnaam3` FROM`tabelnaam`;  *De volgorde waarin je de veldnamen opsomt, bepaalt ook de volgorde van deze velden in de resultatenset.* |

|  |  |
| --- | --- |
|  | Merk op dat de sleutelwoorden SELECT en FROM in hoofdletters uitgewerkt werden. Dit is niet noodzakelijk. In deze cursus doen we dit om duidelijk te maken dat dit sleutelwoorden zijn en geen veld-, tabel- of databanknamen. |

Het is belangrijk dat je steeds aangeeft op welke databank je de query wenst uit te voeren. Het is namelijk mogelijk dat eenzelfde tabelnaam in meerdere databanken op je server voorkomt. Dit kan op twee manieren:

1. Typ USE `databasenaam`; boven de query(‘s) die je wenst uit te voeren. Het voordeel van deze methode is dat alle queries die op deze instructie volgen dezelfde databank zullen gebruiken.

|  |  |  |  |
| --- | --- | --- | --- |
| **SYNTAX** | | | **DATABASEBENOEMING** |
| 1 | USE `databasenaam`; | | |
| 2 | SELECT \* FROM `tabelnaam`; | | |
| **VOORBEELD** | | **EXTERNE DATABASEBENOEMING** | |
| 1 | USE `cvo\_volt`; | | |
| 2 | SELECT \* FROM `courses`; | | |

1. Verwerk de naam van de database die je wenst te gebruiken in je opvraagquery. Deze methode krijgt vaak de voorkeur, aangezien deze minder regels code vergt.

|  |  |  |
| --- | --- | --- |
| **SYNTAX** | | **INTERNE DATABASEBENOEMING** |
| 1 | SELECT \* FROM `databasenaam`.`tabelnaam`; | |
| **VOORBEELD** | | **INTERNE DATABASEBENOEMING** |
| 1 | SELECT \* FROM `cvo\_volt`.` courses`; | |

|  |  |
| --- | --- |
|  | **BACKTICKS**  Merk op dat veld-, tabel- en databanknamen altijd tussen backticks (`) worden geschreven. Hiermee voorkom je dat de databasesoftware deze namen als SQL-sleutelwoorden zal beschouwen. Dankzij het gebruik van backticks kunnen sleutelwoorden als 'use', 'select', 'from' (en vele andere) ook gebruikt worden als veld-, tabel- en databanknamen in de ontwerpfase van een database. Het gebruik van backticks is optioneel, maar in deze cursus steeds verplicht. Een backtick kan worden uitgewerkt door eerst de toetsencombinatie **ALT GR** + **`** te drukken, gevolgd door een spatie. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OEFENINGEN** | EEN SELECTIE VAN GEGEVENS OPVRAGEN | | | |
| Niveau |  |  | 10 min |
| Bekijk de onderstaande oefeningen. Werk en test de gevraagde SQL-code uit in MySQL Workbench en kopieer deze vervolgens in het daarvoor voorziene veld in dit Word-document.  Vergeet niet in elke query:   * aan te geven met welke database je wenst te werken. * alle niet-sleutelwoorden tussen backticks (``) uit te werken. * Indien de oefening over een verbetersleutel beschikt mag je enkel de verbeterde versie in dit document verwerken. | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **1** |  | **ZOO** | Toon alle velden uit de tabel ‘*animals’*. |
| **OPLOSSING** | USE `zoo`;  SELECT \* FROM `animals`; | | |
| **BEOOGD RESULTAAT** | Afbeelding met tekst, schermopname, Lettertype, nummer  Automatisch gegenereerde beschrijving | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **2** |  | **ZOO** | Toon de naam van alle parkgebieden. |
| **OPLOSSING** | USE `zoo`;  SELECT name FROM `parkzones`; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **3** |  | **ZOO** | Toon de naam, het gewicht en de geboortedatum (in deze volgorde!) van alle dieren. |
| **OPLOSSING** | USE `zoo`;  SELECT name,weight,date\_of\_birth FROM `animals`; | | |
| **BEOOGD RESULTAAT** | Afbeelding met tekst, schermopname, Lettertype, nummer  Automatisch gegenereerde beschrijving | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **INSTRUCTIEVIDEO** | **VERBETERSLEUTEL OEFENING 1 - 3** |  | 12 min |
| Klik [hier](https://youtu.be/uYzNtqkVgGE) om de video te bekijken. (houd de CTRL-toets ingedrukt om de link te openen) | | |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **4** |  | **DIGITECH** |  |
| **OPGAVE** | Toon alle velden uit de tabel ‘*employees’*. | | |
| **OPLOSSING** | USE `digitech`;  SELECT \* FROM `employees`; | | |
| **BEOOGD RESULTAAT** | Afbeelding met tekst, schermopname, Lettertype, nummer  Automatisch gegenereerde beschrijving | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **5** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam en de beschrijving (in deze volgorde!) van alle rollen. | | |
| **OPLOSSING** | USE `digitech`;  SELECT name,description FROM `roles`; | | |
| **BEOOGD RESULTAAT** | Afbeelding met tekst, schermopname, Lettertype, nummer  Automatisch gegenereerde beschrijving | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **6** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam, voornaam, geboortedatum en het bedrijfse-mailadres (in deze volgorde!) van alle werknemers. | | |
| **OPLOSSING** | USE `digitech`;  SELECT last\_name,first\_name,date\_of\_birth,company\_email FROM `employees`; | | |
| **BEOOGD RESULTAAT** |  | | |

# Commentaarregels

Via commentaaregels is het mogelijk om notities in je querybestanden te verwerken, zonder dat deze door het DBMS als SQL-instructies beschouwd zullen worden. Daarnaast kunnen commentaaregels ook gebruikt worden om query-onderdelen, tijdens het testproces, tijdelijk uit te sluiten, zonder deze delen te moeten verwijderen.

## Enkele commentaarregel

Wens je je notitie op slechts 1 regel uit te werken, plaats dan twee opeenvolgende koppeltekens (--) OF een hashtag (#) voor je notitie. Voorzie achter deze tekens wel steeds een spatie.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYNTAX** | | | **ENKELE COMMENTAARREGEL** | | | |
| 1 | - - commentaartekst | | | OF | 1 | #commentaartekst |
| **VOORBEELD** | | **ENKELE COMMENTAARREGEL** | | | | |
| 1 | - - De onderstaande query vraagt alle gegevens uit de tabel students op | | | | | |
| 2 | SELECT \* FROM `cvo\_volt`.`students`; | | | | | |
| 3 | #De onderstaande query vraagt alle gegevens uit de tabel students op | | | | | |
| 4 | SELECT \* FROM `cvo\_volt`.`students`; | | | | | |

Zoals eerder gezegd wordt kan een commentaarregel ook gebruikt worden om aan te geven dat een query bij het uitvoeren van de code genegeerd moet worden. De query op regel 1 zal niet uitgevoerd worden, daar deze in een commentaarregel werd verwerkt. De tweede query wordt wel uitgevoerd.

|  |  |  |
| --- | --- | --- |
| **VOORBEELD** | | **ENKELE COMMENTAARREGEL** |
| 1 | **#**SELECT \* FROM `cvo\_volt`.`students`; | |
| 2 | SELECT \* FROM `cvo\_volt`.` courses`; | |

## Meervoudige commentaarregel

Indien je notitie meerdere coderegels in beslag zal nemen, kan je gebruik maken van /\* \*/.

|  |  |  |  |
| --- | --- | --- | --- |
| **SYNTAX** | | | **MEERVOUDIGE COMMENTAARREGEL** |
| 1 | /\* commentaartekst verspreid | | |
| 2 | over meerdere regels \*/ | | |
| **VOORBEELD** | | **MEERVOUDIGE COMMENTAARREGEL** | |
| 1 | /\* De onderstaande query vraagt | | |
| 2 | alle gegevens uit de tabel students op. \*/ | | |
| 3 | SELECT \* FROM `cvo\_volt`.`students`; | | |

# Filteren van gegevens

Wanneer je in je database opzoek wenst te gaan naar gegevens die aan bepaalde voorwaarde moeten voldoen, kan je gebruik maken van één van de vele filtertechnieken. Je kan een filtermethode aan je SELECT-query toevoegen, door de query uit te breiden met het sleutelwoord WHERE. Achter dit sleutelwoord kan je één of meerdere voorwaarden opgeven.

## Filteren op basis van een overeenkomst

Wil je gegevens filteren op basis van een overeenkomst, maak dan gebruik van de vergelijkingsoperator ‘=’.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SYNTAX** | | | **OPVRAGING MET OVEREENKOMST** | |
| 1 | SELECT \* FROM `databasenaam`.`tabelnaam` WHERE `veldnaam` = 'waarde'**;** | | | |
| Je vraagt alle rijen op waarbij een bepaald veld een vaste waarde bevat. | | | | |
| **VOORBEELD** | | **OPVRAGING MET OVEREENKOMST** | | |
| 1 | SELECT \* FROM `cvo\_volt`.`students` WHERE `last\_name` = 'Liefsoens'; | | | |
| Je vraagt alle rijen op uit de tabel cursisten, waarbij het veld ‘last\_name’ de waarde ‘Liefsoens’ bevat. | | | | |
|  | | | | |
| **RESULTATENSET** | | | | **OPVRAGING MET OVEREENKOMST** |
| Afbeelding met tekst, schermopname, Lettertype, lijn  Automatisch gegenereerde beschrijving | | | | |

|  |  |
| --- | --- |
|  | Tekstuele waarden, zogenaamde strings, moeten in een filter steeds tussen aanhalingstekens (') geplaatst worden. Numerieke hoeven niet omringt te worden door aanhalingstekens. |

Indien je wil controleren of een veld niet overeenkomt met een bepaalde waarde, dan moet je een uitroepteken (!) voor het gelijkteken plaatsen.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SYNTAX** | | | **OPVRAGING MET OVEREENKOMST** | |
| 1 | SELECT \* FROM `databasenaam`.`tabelnaam` WHERE `veldnaam` != 'waarde'**;** | | | |
| Je vraagt alle rijen op waarbij een bepaald veld geen vaste waarde bevat. | | | | |
| **VOORBEELD** | | **OPVRAGING MET OVEREENKOMST** | | |
| 1 | SELECT \* FROM `cvo\_volt`.`students` WHERE `last\_name` != 'Liefsoens'; | | | |
| Je vraagt alle rijen op uit de tabel cursisten, waarbij het veld ‘last\_name’ niet de waarde ‘Liefsoens’ bevat. | | | | |
| **RESULTATENSET** | | | | **OPVRAGING MET OVEREENKOMST** |
| Afbeelding met tekst, schermopname, Lettertype, nummer  Automatisch gegenereerde beschrijving | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OEFENINGEN** | FILTEREN VAN GEGEVENS | | | |
| Niveau |  |  | 10 min |
| Bekijk de onderstaande oefeningen. Werk en test de gevraagde SQL-code uit in MySQL Workbench en kopieer deze vervolgens in het daarvoor voorziene veld in dit Word-document.  Vergeet niet in elke query:   * aan te geven met welke database je wenst te werken. * alle niet-sleutelwoorden tussen backticks (``) uit te werken. * Indien de oefening over een verbetersleutel beschikt mag je enkel de verbeterde versie in dit document verwerken. | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **7** |  | **ZOO** |  |
| **OPGAVE** | Toon alle gegevens van de vrouwelijke dieren. | | |
| **OPLOSSING** | SELECT \* FROM `zoo`.`animals` WHERE `gender` = 'v'; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **8** |  | **ZOO** |  |
| **OPGAVE** | Toon de naam en geboortedatum van alle dieren, behalve die van het dier met de naam “Pasta”. | | |
| **OPLOSSING** | /\* De opdracht zegt 'naam en geboortedatum'  maar de screenshot geeft alle kolommen  dus doe ik beide in die volgorde. \*/  SELECT name,date\_of\_birth FROM `zoo`.`animals` WHERE `name` != 'Pasta';  SELECT \* FROM `zoo`.`animals` WHERE `name` != 'Pasta'; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **INSTRUCTIEVIDEO** | **VERBETERSLEUTEL OEFENING 7 - 8** |  | 4 min |
| Klik [hier](https://youtu.be/pmOIaarlIBA) om de video te bekijken. (houd de CTRL-toets ingedrukt om de link te openen) | | |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **9** |  | **DIGITECH** |  |
| **OPGAVE** | Toon alle gegevens van de bedrijfsfunctie 'Junior backend developer'.  Voeg ook een enkele commentaarregel toe met de vermelding van de functienaam die je zelf zou willen bekleden indien je binnen dit bedrijf zou werken. | | |
| **OPLOSSING** | -- Ik zou voor Full Stack Dev gaan  SELECT \*  FROM `digitech`.`roles`  WHERE name = 'Junior backend developer'; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **10** |  | **DIGITECH** |  |
| **OPGAVE** | Toon het id, de naam en voornaam (in die volgorde!) van de werknemers die op 15 maart 1991 geboren zijn.  Voeg ook een meervoudige commentaarregel toe waarin je je eigen geboortedatum en geboorteplaats beschrijft (dit mogen fictieve gegevens zijn). | | |
| **OPLOSSING** | /\*  Geboortedatum 1983-07-31  Geboorteplaats Leuven  \*/  SELECT id,last\_name,first\_name  FROM `digitech`.`employees`  WHERE `date\_of\_birth` = "1991-03-15"; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **11** |  | **DIGITECH** |  |
| **OPGAVE** | Toon alle gegevens van alle werknemers, behalve de werknemer met het bedrijfsmailadres ‘lisa.looyen@digitech.com’. | | |
| **OPLOSSING** | SELECT \*  FROM `digitech`.`employees`  WHERE `company\_email`!= 'lisa.looyen@digitech.com'; | | |
| **BEOOGD RESULTAAT** |  | | |

## Filteren op basis van een patroon

Bij het opvragen van gegevens kan er ook gefilterd worden op basis van een patroon. Hierbij moet de WHERE voorwaarde uitgebreid worden met het LIKE sleutelwoord en één of meerdere van de onderstaande jokertekens.

|  |  |
| --- | --- |
| **JOKERTEKEN** | **BETEKENIS** |
| % | Op de plaats van dit jokertekens kunnen 0, 1 of meerdere tekens staan. |
| \_ | Op de plaats van dit teken (liggend streepje) moet er 1 teken staan. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SYNTAX** | | **OPVRAGING VAN GEGEVENS DIE AAN PATROON VOLDOEN** | | |
| 1 | SELECT \* FROM `databasenaam`.`tabelnaam` WHERE `veldnaam` LIKE '%patroon%'**;** | | | |
| Je vraagt alle records en velden op waarbij het gegeven veld aan de voorwaarden van het patroon voldoet. | | | | |
| **VOORBEELD** 1 | | | **OPVRAGING VAN GEGEVENS DIE AAN PATROON VOLDOEN** | |
| 1 | SELECT \* FROM `cvo\_volt`.`students` WHERE `last\_name` LIKE 'b%'; | | | |
| Je vraagt alle gegevens op van de cursisten waarvan de naam met de letter ‘b’ begint. | | | | |
| **RESULTATENSET** 1 | | | | **OPVRAGING VAN GEGEVENS DIE AAN PATROON VOLDOEN** |
| Afbeelding met tekst, schermopname, Lettertype, lijn  Automatisch gegenereerde beschrijving | | | | |
| **VOORBEELD** 2 | | | **OPVRAGING VAN GEGEVENS DIE AAN PATROON VOLDOEN** | |
| 1 | SELECT \* FROM `cvo\_volt`.`students` WHERE `email` LIKE '%@%.be'; | | | |
| Je vraagt alle gegevens op van de cursisten die een e-mailadres hebben met een Belgisch topleveldomein. | | | | |
| **RESULTATENSET 2** | | | | **OPVRAGING VAN GEGEVENS MET PATROON** |
| Afbeelding met tekst, Lettertype, lijn, nummer  Automatisch gegenereerde beschrijving | | | | |

Indien je wil controleren of een veld niet aan een bepaald patroon voldoet, dan moet je gebruik maken van het sleutelwoord NOT LIKE.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SYNTAX** | | | **OPVRAGING VAN GEGEVENS DIE NIET AAN PATROON VOLDOEN** | |
| 1 | SELECT \* FROM `databasenaam`.`tabelnaam` WHERE `veldnaam` NOT LIKE '%patroon%'**;** | | | |
| Vraagt alle velden op die aan de voorwaarden van het patroon voldoen. | | | | |
| **VOORBEELD** | | **OPVRAGING VAN GEGEVENS DIE NIET AAN PATROON VOLDOEN** | | |
| 1 | SELECT \* FROM `cvo\_volt`.`students` WHERE `first\_name` NOT LIKE '\_a%'; | | | |
| Je vraagt alle velden op uit de tabel cursisten, waarbij de tweede letter van de voornaam geen ‘a’ is. | | | | |
| **RESULTATENSET** | | | | **OPVRAGING VAN GEGEVENS DIE NIET AAN PATROON VOLDOEN** |
| Afbeelding met tekst, schermopname, Lettertype, nummer  Automatisch gegenereerde beschrijving | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OEFENINGEN** | FILTEREN VAN GEGEVENS | | | |
| Niveau |  |  | 10 min |
| Bekijk de onderstaande oefeningen. Werk en test de gevraagde SQL-code uit in MySQL Workbench en kopieer deze vervolgens in het daarvoor voorziene veld in dit Word-document.  Vergeet niet in elke query:   * aan te geven met welke database je wenst te werken. * alle niet-sleutelwoorden tussen backticks (``) uit te werken. * Indien de oefening over een verbetersleutel beschikt mag je enkel de verbeterde versie in dit document verwerken. | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **12** |  | **ZOO** |  |
| **OPGAVE** | Toon alle parkgebiednamen die op een ‘a’ eindigen. | | |
| **OPLOSSING** | SELECT \*  FROM `zoo`.`parkzones`  WHERE `name` LIKE '%a'; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **13** |  | **ZOO** |  |
| **OPGAVE** | Toon alle gegevens van de dieren die op de 6de dag van de maand zijn geboren. | | |
| **OPLOSSING** | SELECT \*  FROM `zoo`.`animals`  WHERE `date\_of\_birth` LIKE '\_\_\_\_-\_\_-06'; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **INSTRUCTIEVIDEO** | **VERBETERSLEUTEL OEFENING 12 - 13** |  | 4 min |
| Klik [hier](https://youtu.be/DOygoK63zDo) om de video te bekijken. (houd de CTRL-toets ingedrukt om de link te openen) | | |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **14** |  | **DIGITECH** |  |
| **OPGAVE** | Toon alle gegevens van de werknemers die over een persoonlijk gmail e-mailadres beschikken. | | |
| **OPLOSSING** | SELECT \*  FROM `digitech`.`employees`  WHERE `personal\_email` LIKE '%gmail%'; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **15** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam, voornaam en aanwerfdatum van alle werknemers die in het jaar 2016 aangeworven werden. | | |
| **OPLOSSING** | SELECT last\_name,first\_name,recruit\_date  FROM `digitech`.`employees`  WHERE `recruit\_date` LIKE '%2016%'; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **16** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam, voornaam en geboortedatum van alle werknemers die in de maand augustus geboren zijn. | | |
| **OPLOSSING** | SELECT last\_name,first\_name,date\_of\_birth  FROM `digitech`.`employees`  WHERE `date\_of\_birth` LIKE '%-08-%'; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **17** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam en beschrijving van alle bedrijfsfuncties waarvan de naam **niet** op 'developer' eindigt. | | |
| **OPLOSSING** | SELECT name,description  FROM `digitech`.`roles`  WHERE `name` NOT LIKE '%developer'; | | |
| **BEOOGD RESULTAAT** |  | | |

## Groter dan of gelijk aan / kleiner dan of gelijk aan

Met de '>' (groter dan) en '<' (kleiner dan) operator kan je nagaan of de waarde van een veld groter of kleiner is dan een gegeven waarde.

|  |  |  |
| --- | --- | --- |
| **SYNTAX** | | **GROTER DAN OF GELIJK AAN / KLEINER DAN OF GELIJK AAN** |
| 1 | SELECT \* FROM `databasenaam`.`tabelnaam` WHERE `veldnaam` > waarde; | |
| De resultatenset bevat enkel de records waarbij de waarde van het gegeven veld **groter** is dan de gegeven waarde. | | |
| 1 | SELECT \* FROM `databasenaam`.`tabelnaam` WHERE `veldnaam` >= waarde; | |
| De resultatenset bevat enkel de records waarbij de waarde van het gegeven veld **groter** of **gelijk** is aan de gegeven waarde. | | |
| 1 | SELECT \* FROM `databasenaam`.`tabelnaam` WHERE `veldnaam` < waarde; | |
| De resultatenset bevat enkel de records waarbij de waarde van het gegeven veld **kleiner** is dan de gegeven waarde. | | |
| 1 | SELECT \* FROM `databasenaam`.`tabelnaam` WHERE `veldnaam` <= waarde; | |
| De resultatenset bevat enkel de records waarbij de waarde van het gegeven veld **kleiner** of **gelijk** is aan de gegeven waarde. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **VOORBEELD** | | **GROTER DAN** | |
| 1 | SELECT \* FROM `cvo\_volt`.`courses` WHERE `max\_students` > 10; | | |
| *Je vraagt alle cursusvelden op van de cursussen die meer dan 10 studenten toelaten (dus minstens 11 studenten).* | | | |
| **RESULTATENSET** | | | **GROTER DAN** |
| Afbeelding met tekst, schermopname, Lettertype, nummer  Automatisch gegenereerde beschrijving | | | |

Meestal worden deze vergelijkingsoperatoren gebruikt om numerieke waarden met elkaar te vergelijken. Ze kunnen echter ook gebruikt worden om tekstuele gegevens als datums en strings met elkaar te vergelijken. Bij datums wordt er gecontroleerd of de veldwaarde voor of na een bepaalde datum ligt.

|  |  |  |  |
| --- | --- | --- | --- |
| **VOORBEELD** | | **GROTER DAB MET DATUMS** | |
| 1 | SELECT \* FROM `cvo\_volt`.`students` WHERE `birth\_date` > '1990-01-01'; | | |
| *Je vraagt alle velden op van de studenten die na 1 januari 1990 geboren zijn (dus vanaf 2 januari 1190).* | | | |
| **RESULTATENSET** | | | **GROTER DAB MET DATUMS** |
| Afbeelding met tekst, schermopname, Lettertype, nummer  Automatisch gegenereerde beschrijving | | | |

Bij andere tekstuele waarden wordt de alfabetische volgorde vergeleken.

|  |  |  |  |
| --- | --- | --- | --- |
| **VOORBEELD** | | **GROTER / KLEINER OF GELIJK AAN MET STRINGS** | |
| 1 | SELECT \* FROM `cvo\_volt`.`students` WHERE `first\_name` > 'Marcel'; | | |
| *Je vraagt alle velden op van de studenten waarvan de voornaam alfabetisch achter ‘Marcel’ gerangschikt zou worden.* | | | |
| **RESULTATENSET** | | | **GROTER / KLEINER OF GELIJK AAN MET DATUMS** |
|  | | | |

Afbeelding met tekst, schermopname, Lettertype, nummer

Automatisch gegenereerde beschrijving

## (not) between

Met BETWEEN kan je de waarden opvragen die tussen een opgegeven bereik liggen (inclusief de begin- en eindwaarde).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SYNTAX** | | | **BETWEEN** | |
| 1 | SELECT \* FROM `databasenaam`.`tabelnaam` WHERE `veldnaam` BETWEEN waarde1 AND waarde2; | | | |
| **VOORBEELD 1** | | **BETWEEN** | | |
| 1 | SELECT \* FROM `cvo\_volt`.`courses` WHERE `max\_students` BETWEEN 1 AND 10; | | | |
| *Je vraagt alle velden op van de cursussen die maximaal tussen de 1 en 10 studenten toelaten. Beide waarden zijn inclusief (tellen mee). 1 en 10 kunnen dus, 0 en 11 niet.* | | | | |
| **RESULTATENSET** 1 | | | | **BETWEEN** |
| Afbeelding met tekst, schermopname, Lettertype, lijn  Automatisch gegenereerde beschrijving | | | | |
| **VOORBEELD 2** | | **BETWEEN** | | |
| 1 | SELECT \* FROM `cvo\_volt`.`students` WHERE `birth\_date` BETWEEN '1980-01-01' AND '1989-12-31'; | | | |
| *Je vraagt alle velden op uit de tabel students, van de studenten die tussen 1980 en 1990 geboren zijn. De begin en einddatum zijn steeds inclusief (tellen mee).* | | | | |
| **RESULTATENSET** 2 | | | | **BETWEEN** |
| Afbeelding met tekst, Lettertype, lijn, schermopname  Automatisch gegenereerde beschrijving | | | | |

Het is ook mogelijk om te controleren of een bepaalde waarde niet tussen een bepaald bereik ligt. Maak dan gebruik van NOT BETWEEN.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SYNTAX** | | | **NOT BETWEEN** | |
| 1 | SELECT \* FROM `databasenaam`.`tabelnaam` WHERE `veldnaam` NOT BETWEEN waarde1 AND waarde2; | | | |
| **VOORBEELD 1** | | **NOT BETWEEN** | | |
| 1 | SELECT \* FROM `cvo\_volt`.`students` WHERE `birth\_date` NOT BETWEEN '1980-05-04' AND '1995-12-31'; | | | |
| *Je vraagt alle gegevens op van de studenten die* ***niet*** *tussen 4 mei 1980 en 31 december 1995 geboren zijn.* | | | | |
| **RESULTATENSET** 1 | | | | **BETWEEN** |
| Afbeelding met tekst, schermopname, Lettertype, lijn  Automatisch gegenereerde beschrijving | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OEFENINGEN** | FILTEREN VAN GEGEVENS | | | |
| Niveau |  |  | 25 min |
| Bekijk de onderstaande oefeningen. Werk en test de gevraagde SQL-code uit in MySQL Workbench en kopieer deze vervolgens in het daarvoor voorziene veld in dit Word-document.  Vergeet niet in elke query:   * aan te geven met welke database je wenst te werken. * alle niet-sleutelwoorden tussen backticks (``) uit te werken. * Indien de oefening over een verbetersleutel beschikt mag je enkel de verbeterde versie in dit document verwerken. | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **18** |  | **ZOO** |  |
| **OPGAVE** | Toon de naam en het gewicht van alle dieren die meer dan 400kg wegen. | | |
| **OPLOSSING** | SELECT name,weight  FROM `zoo`.`animals`  WHERE `weight` > 400; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **19** |  | **ZOO** |  |
| **OPGAVE** | Toon de naam en geboortedatum van alle dieren die tussen 2013 en 2016 (excl.) geboren zijn. | | |
| **OPLOSSING** | SELECT name,date\_of\_birth  FROM `zoo`.`animals`  WHERE `date\_of\_birth` BETWEEN '2013-01-01'AND '2015-12-31'; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **20** |  | **ZOO** |  |
| **OPGAVE** | Toon de naam van alle parkgebieden die alfabetisch voor ‘Azië’ gerangschikt moeten worden. | | |
| **OPLOSSING** | SELECT name  FROM `zoo`.`parkzones`  WHERE `name` < 'Azië'; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **INSTRUCTIEVIDEO** | **VERBETERSLEUTEL OEFENING 18 - 20** |  | 7 min |
| Klik [hier](https://youtu.be/-IsuIPslbkc) om de video te bekijken. (houd de CTRL-toets ingedrukt om de link te openen) | | |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **21** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam, voornaam en aanwervingsdatum van alle werknemers die voor 2018 aangeworven werden. | | |
| **OPLOSSING** | SELECT last\_name,first\_name,recruit\_date  FROM `digitech`.`employees`  WHERE `recruit\_date` > 2018; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **22** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam, voornaam, datum en tijd van het laatste functioneringsgesprek van alle werknemers die tussen 16 oktober 2021 om 09.00 u. en 5 september 2023 om 16.00 u. hun laatste functioneringsgesprek hebben gehad. | | |
| **OPLOSSING** | SELECT last\_name,first\_name,last\_performance\_interview  FROM `digitech`.`employees`  WHERE `last\_performance\_interview` BETWEEN '2021-10-16 09:00:00' AND '2023-09-05 16:00:00'; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **23** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam en het maandloon van de werknemers die meer dan 1850 euro/maand verdienen. | | |
| **OPLOSSING** | SELECT last\_name,monthly\_wages  FROM `digitech`.`employees`  WHERE `monthly\_wages` > 1850; | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **24** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam, voornaam en geboortedatum van alle werknemers die **niet** tussen 1989 en 1992 (beide jaren inclusief) geboren zijn. | | |
| **OPLOSSING** | SELECT last\_name,first\_name,date\_of\_birth  FROM `digitech`.`employees`  WHERE `date\_of\_birth` NOT BETWEEN '1989-01-01' AND '1992-12-31'; | | |
| **BEOOGD RESULTAAT** |  | | |

## de AND en OR operator

Indien je meerdere voorwaarden in een query wenst te verwerken, kan je gebruikmaken van de AND en OR operator.

Indien je resultatenset aan meerdere voorwaarden moet voldoen, moet je gebruik maken van de AND operator.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SYNTAX** | | **AND OPERATOR** | | |
| 1 | SELECT **`**veldnaam1`, `veldnaam2` FROM`databasenaam`.`tabelnaam` WHERE voorwaarde\_1 AND voorwaarde\_2; | | | |
| **VOORBEELD** | | | **AND OPERATOR** | |
| 1 | SELECT `first\_name`, `last\_name`, `birth\_date`, `registered\_since` FROM `cvo\_volt`.`students` WHERE `birth\_date` < '2000-01-01' AND `registered\_since` >= '2017-01-01'; | | | |
| *Deze query vraagt de gegevens op van de cursisten die voor het jaar 2000 geboren zijn EN zich vanaf het jaar 2017 geregistreerd hebben.* | | | | |
| **RESULTATENSET** | | | | **AND OPERATOR** |
| Afbeelding met tekst, schermopname, Lettertype, nummer  Automatisch gegenereerde beschrijving | | | | |

Indien je resultatenset aan één van de voorwaarden moet voldoen, moet je gebruik maken van de **OR operator**.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SYNTAX** | | **OR OPERATOR** | | |
| 1 | SELECT **`**veldnaam1`, `veldnaam2` FROM`databasenaam`.`tabelnaam` WHERE voorwaarde\_1 OR voorwaarde\_2; | | | |
| **VOORBEELD** | | | **OR OPERATOR** | |
| 1 | SELECT `first\_name`, `last\_name`, `birth\_date`, `registered\_since` FROM `cvo\_volt`.`students` WHERE `last\_name` = 'De Vriendt' OR `last\_name` = 'Van Den Heuvel'; | | | |
| *Deze query vraagt de gegevens op van de cursisten met de achternaam 'De Vriendt' OF 'Van Den Heuvel'. De AND operator kan hier nooit voor gebruikt worden; een cursist kan namelijk slechts 1 achternaam hebben, geen twee.* | | | | |
| **RESULTATENSET** | | | | **AND OPERATOR** |
| Afbeelding met tekst, schermopname, Lettertype, lijn  Automatisch gegenereerde beschrijving | | | | |

Je kan de AND en OR operator ook combineren. Hierbij is het aangewezen om haakjes te gebruiken. Alles wat tussen haakjes staat wordt als 1 voorwaarde beschouwd.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SYNTAX** | | **COMBINATIE AND / OR** | | |
| 1 | SELECT **`**veldnaam1`, `veldnaam2` FROM`databasenaam`.`tabelnaam` WHERE voorwaarde\_1 AND (voorwaarde\_2 OR voorwaarde\_2); | | | |
| **VOORBEELD** | | | **COMBINATIE AND / OR** | |
| 1 | SELECT `first\_name`, `last\_name`, `registered\_since` FROM `cvo\_volt`.`students` WHERE `registered\_since` > '2015-06-01' AND (`last\_name` = 'De Vriendt' OR `last\_name` = 'Van Den Heuvel' ); | | | |
| *Deze query vraagt de gegevens op van de cursisten die zich na 1 juni 2015 hebben geregistreerd en die de achternaam 'De Vriendt' OF 'Van Den Heuvel' hebben.* | | | | |
| **RESULTATENSET** | | | | **AND OPERATOR** |
| Afbeelding met tekst, Lettertype, schermopname, lijn  Automatisch gegenereerde beschrijving | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OEFENINGEN** | FILTEREN VAN GEGEVENS | | | |
| Niveau |  |  | 20 min |
| Bekijk de onderstaande oefeningen. Werk en test de gevraagde SQL-code uit in MySQL Workbench en kopieer deze vervolgens in het daarvoor voorziene veld in dit Word-document.  Vergeet niet in elke query:   * aan te geven met welke database je wenst te werken. * alle niet-sleutelwoorden tussen backticks (``) uit te werken. * Indien de oefening over een verbetersleutel beschikt mag je enkel de verbeterde versie in dit document verwerken. | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **25** |  | **ZOO** |  |
| **OPGAVE** | Toon de naam, het geslacht en het gewicht van alle mannelijke dieren die 100kg of minder wegen. | | |
| **OPLOSSING** | Plak hier de gevraagde SQL-code. | | |
| **BEOOGD RESULTAAT** | (geen resultaten) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **26** |  | **ZOO** |  |
| **OPGAVE** | Toon de naam van alle diersoorten die met de letter ‘S’ beginnen of met de ‘r’ eindigen (tip: werk met LIKE). | | |
| **OPLOSSING** | Plak hier de gevraagde SQL-code. | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **INSTRUCTIEVIDEO** | **VERBETERSLEUTEL OEFENING 25 - 26** |  | 4 min |
| Klik [hier](https://youtu.be/Yn3vaG_-suI) om de video te bekijken. (houd de CTRL-toets ingedrukt om de link te openen) | | |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **26** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de rolnamen met het id 1 en 3. | | |
| **OPLOSSING** | Plak hier de gevraagde SQL-code. | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **27** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam, voornaam en maandloon van alle werknemers die in 2017 werden aangenomen (tip: werk met like) en meer dan 1900 euro per maand verdienen. | | |
| **OPLOSSING** | Plak hier de gevraagde SQL-code. | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **28** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam en voornaam van alle werknemers die in 1975 of 1988 geboren zijn (tip: werk met like). | | |
| **OPLOSSING** | Plak hier de gevraagde SQL-code. | | |
| **BEOOGD RESULTAAT** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **29** |  | **DIGITECH** |  |
| **OPGAVE** | Toon de naam, voornaam en maandloon van alle werknemers die maximum 2125 euro per maand verdienen en daarnaast over een persoonlijk telenet of hotmail emailadres beschikken. Tip: werk met haakjes! | | |
| **OPLOSSING** | Plak hier de gevraagde SQL-code. | | |
| **BEOOGD RESULTAAT** | Afbeelding met tekst, schermopname, Lettertype, lijn  Automatisch gegenereerde beschrijving | | |

# Werken met veldaliassen

Bij het opvragen van gegevens kan je zelf de veldnamen in de resultatenset bepalen door gebruik te maken van een alias. Dit kan handig zijn als de originele veldnamen in het Engels werden uitgewerkt en je deze bijvoorbeeld in de resultatenset in het Nederlands wil laten verschijnen.

Een alias kan worden toegevoegd door gebruik te maken van het sleutelwoord **AS**, gevolgd door de veldnaam die je in de resultatenset wil laten verschijnen.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SYNTAX** | | **VELDALIASSEN** | | |
| 1 | SELECT **`**veldnaam1` AS `alias\_veld\_1`, `veldnaam2` AS `alias\_veld\_2` FROM`databasenaam`.`tabelnaam`; | | | |
| *Aliasnamen kunnen zowel tussen aanhalingstekens als backticks worden uitgewerkt.* | | | | |
| **VOORBEELD** | | | **VELDALIASSEN** | |
| 1 | SELECT `first\_name` AS `voornaam`, `last\_name` AS `achternaam` FROM `cvo\_volt`.`students`; | | | |
| *Het sleutelwoord ‘AS’ is optioneel en mag dus weggelaten worden.* | | | | |
| **RESULTATENSET** | | | | **VELDALIASSEN** |
| Afbeelding met tekst, schermopname, Lettertype, nummer  Automatisch gegenereerde beschrijving | | | | |

|  |  |
| --- | --- |
|  | Een alias bepaalt enkel de naam van het veld in de resultatenset. De veldnaam waarin de data wordt bewaard blijft onaangeroerd. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OEFENINGEN** | VELDALIASSEN | | | |
| Niveau |  |  | 5 min |
| Bekijk de onderstaande oefeningen. Werk en test de gevraagde SQL-code uit in MySQL Workbench en kopieer deze vervolgens in het daarvoor voorziene veld in dit Word-document.  Vergeet niet in elke query:   * aan te geven met welke database je wenst te werken. * alle niet-sleutelwoorden tussen backticks (``) uit te werken. * Indien de oefening over een verbetersleutel beschikt mag je enkel de verbeterde versie in dit document verwerken. | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **30** |  | **ZOO** |  |
| **OPGAVE** | Werk de oplossing van oefening 19 bij zodat alle resultaatvelden een Nederlandse benaming krijgen. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **INSTRUCTIEVIDEO** | **VERBETERSLEUTEL OEFENING 30** |  | 2 min |
| Klik [hier](https://youtu.be/PMZmeobQjlE) om de video te bekijken. (houd de CTRL-toets ingedrukt om de link te openen) | | |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **31** |  | **DIGITECH** |  |
| **OPGAVE** | Werk oefening 27 bij zodat alle resultaatvelden een Nederlandse benaming krijgen. | | |

# Inleveren

|  |  |  |
| --- | --- | --- |
|  |  | 1. Upload de aangevulde versie van dit Word-document onder de daarvoor voorziene opdracht op Moodle. |
|  | 1. De leerkracht ontvang automatisch een bericht van je inzending en tracht je opdracht binnen de 5 werkdagen na te kijken. |
|  | 1. Nadat je oefeningen nagekeken werden krijg je toegang tot hoofdstuk 5. De leerkracht breng jou hier via mail van de op de hoogte. |