

Databases

Introduction

Databases, 1st period. RDBMS en SQL.

Gegeven aan 2de jaars doorstroomgroep (OITIBO9M)

- [XAMPP Installeren](#)
- [Wat is een database?](#)
- [Nieuwe database aanmaken](#)
- [MyFirst Database](#)
- [Maak database en tabellen](#)
- [Opgaven, mijn eerste Queries](#)
- [Flight Data - Import, nog meer queries](#)
- [Data Types](#)
- [MySQL Main Data Types](#)
- [Mijn eerste queries - deel II](#)
- [Database aanpassen](#)
- [SQL Queries met opdrachten](#)
- [Huiswerk](#)
- [Joins \(les 2, periode 2\)](#)
- [Samenvatting - Leren!](#)
- [Lesplan DB Intro \(8ste les\)](#)

- [De Wereld-Database](#)
- [De Wereld-Database - Distinct](#)

XAMPP Installeren

In deze les leggen we uit hoe je XAMPP moet installeren. XAMPP is software voor de Mac en voor Windows waarin een RDMS (Database) en een webserver zit. Je kunt met deze software oefenen met het aanmaken en raadplegen van databases.

Download de software

Google of download de juiste versie van [apachefriends](http://apachefriends.org)

Voer de installatie uit

Indien je al een keer eerder XAMPP had geïnstalleerd dan moet je de folder c:\xampp weggooien of beter nog een andere naam geven. let op als je websites hebt ontwikkeld of databases hebt aangemaakt dan staan deze waarschijnlijk ergens in de folder c:\xampp.

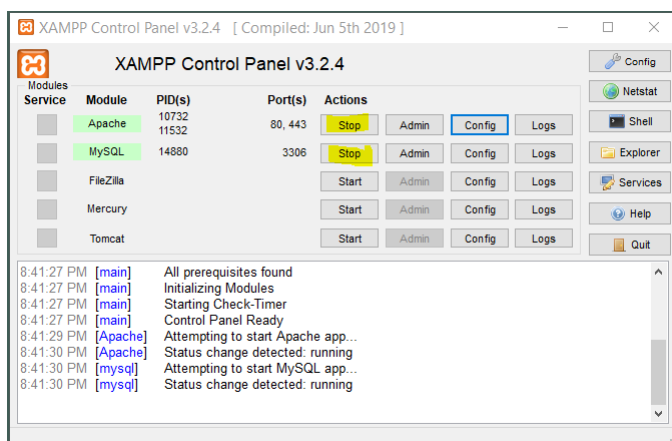
De Installatie spreekt voor zich je kunt overal de standaard opties aanhouden.

Opstarten XAMPP

Als het goed is, is er tijdens de installatie een short cut gemaakt die heet *XAMPP Control*.

Deze shortcut voert `C:\xampp\xampp-control.exe` uit.

Drk op de eerste twee knopjes start naast Apache en MySQL. Hiermee start je de webserver en de database op. Als alles goed is dan zie je het volgende.

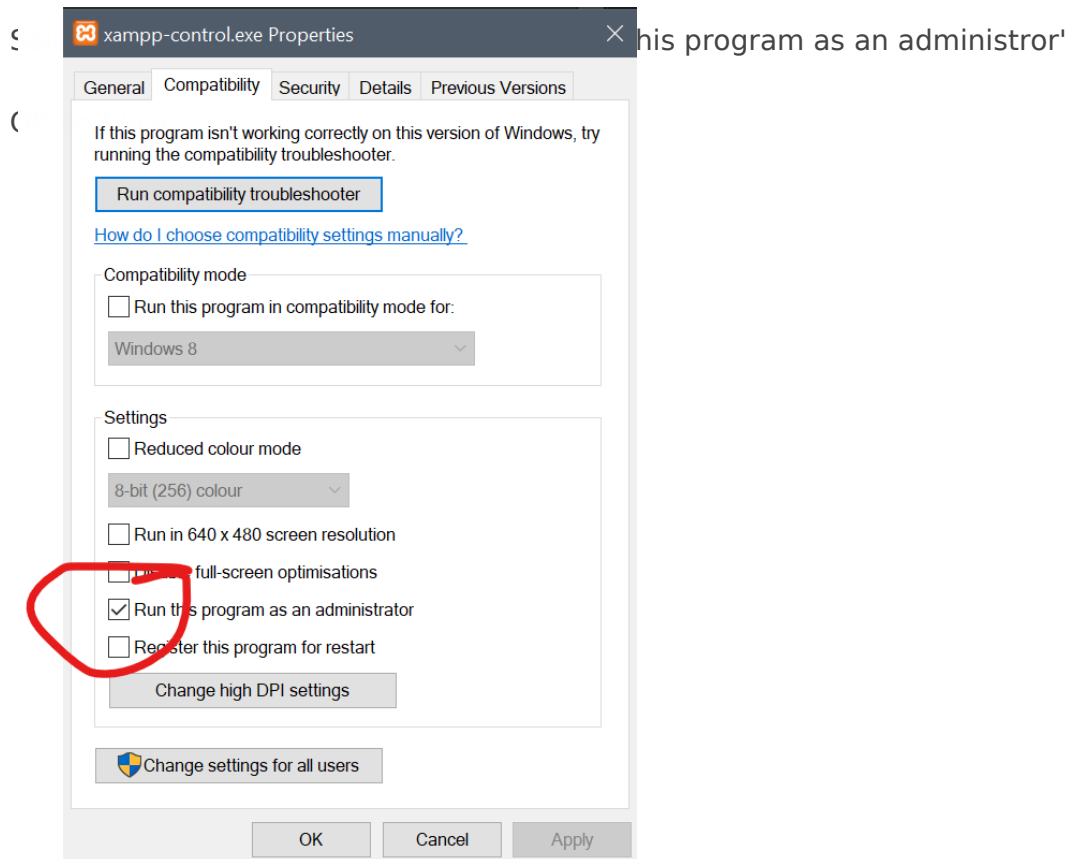


XAMPP Instellingen

We gaan een paar dingen anders instellen.

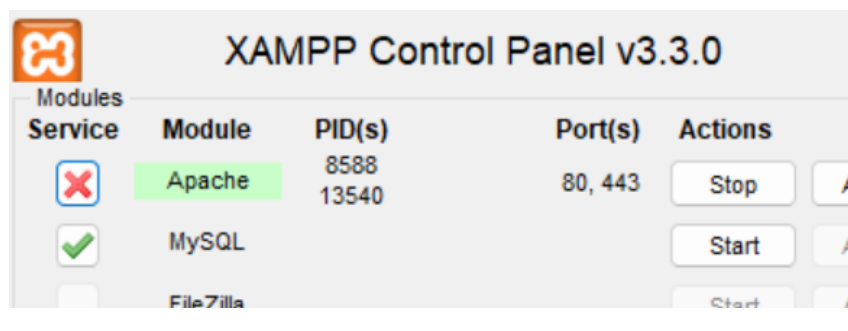
Allereerst gaan we de database als Windows-service laten draaien dat verhoogt de stabiliteit.

Ga via de file explorer naar c:\xampp\ en zoek naar xampp-control.exe, selecteer met rechtermuis propertjes (of eigenschappen).



Sluit XAMPP en start XAMPP opnieuw op.

Druk op het rode kruisje links naast MySQL. Het kruisje wordt een groen vinkje.



Tenslotte, ga naar c:\xampp\htdocs\ en gooi het bestand index.html weg.

Waarom dat is dat zien we later als we met XAMPP gaan werken.

Port conflict

Het kan zijn dat je foutmeldingen ziet, lees die goed. Zie je iets over port conflict op poort 443 (waarschijnlijk met VM-player) dan kun je twee dingen doen. Je kunt VM-player server stoppen of je kunt het portnummer van de webserver veranderen. Dit doe je als volgt:

Druk op de knop Config op de Apache regel en selecteer de `Apache (HTTP-SSL.conf)` optie. Er opent een window. Zoek het stukje dat er als volgt uitziet:

```
## When we also provide SSL we have to listen to the
# standard HTTP port (see above) and to the HTTPS port
#
Listen 443
```

en verander dat in

```
# When we also provide SSL we have to listen to the
# standard HTTP port (see above) and to the HTTPS port
#
Listen 1443
```

Er wordt dus een 1 voor de 443 geplaatst. Sluit de file en bewaar deze. Start de webserver opnieuw op.

PHP path not found in Visual Studio Code

Type je PHP in je VCS-terminal in en krijg je een foutmelding terwijl dat wel werkt in een CMD-box, dan is er iets niet compleet in jouw VCS set-up.

Waarschijnlijk heb je XAMPP geïnstalleerd nadat VCS was geïnstalleerd.

Je moet je configuratie in VCS aanpassen.

Go to **File->Preferences->settings->User settings** **tab->extensions->from** the drop down select php->on the right pane under **PHP > Validate:** Executable Path select edit in settings.json.

Then set the path as your case may be for a XAMPP user who installed XAMPP on c drive you will have:

```
"php.validate.executablePath": "c:\\xampp\\php\\php.exe"
```

If PHP is installed in your system independently or by other means simply set the path of your php.exe file like below:

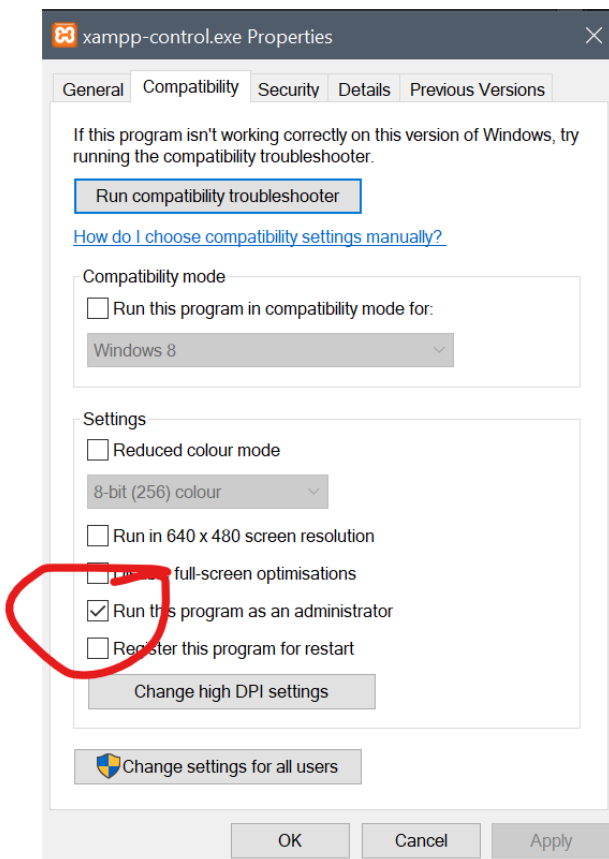
```
"php.validate.executablePath": "C://path.to.your.php.folder//php.exe"
```

Found this solution from [php not found visual studio](#).

Error als je XAMPP afsluit - XAMPP onder administrator

Oplossing: Run XAMPP as administrator.

Rechtermuis, properties van xampp-control.exe



Back-up database terugzetten

Opeens start MySQL niet meer op?

Je kunt in de folder

`C:\xampp\mysql\backup`

alle bestanden kopiëren en plaatsen in

`C:\xampp\mysql\data`

Je moet daarbij de bestaande bestanden *overschrijven*.

Lukt dit niet, dan zul je XAMPP moeten herinstalleren :(

Herinstalleren XAMPP

Opeens start MySQL niet meer op?

Je databases zijn corrupt geraakt. Dat kan bijvoorbeeld als je naar je database aan het schrijven bent en je laptop loopt vast.

In principe ben je je databases kwijt, maar al je PHP-, CSS- en HTML-bestanden gelukkig niet.

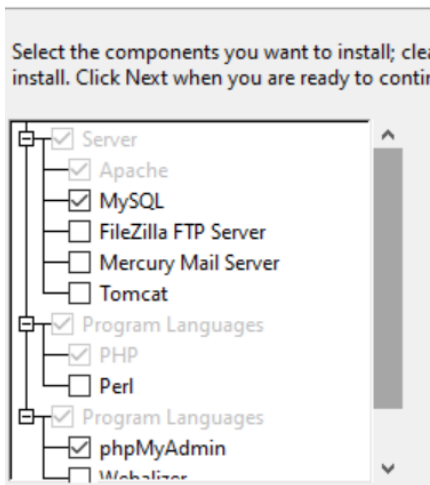
Je zult XAMPP opnieuw moeten installeren.

1. Restart je computer en start XAMPP **niet** op.
2. Ga naar c:\ en hernoem je xampp folder in *xampp.old*. Hiermee zorg je ervoor dat je de PHP bestanden in htdocs niet kwijtraakt als je XAMPP opnieuw installeert.
3. [Download xampp](#) (de laatste stabiele versie voor Windows).

4. Installeer xampp in de folder c:\xampp (dit is de default optie).

Installeer alleen *Apache*, *MySQL* en *phpMyAdmin*.

Select Components



5. Nu heb je een verse xampp installatie
6. Ga naar de c:\xampp folder en verwijder de folder *htdocs*
7. Ga naar c:\xampp.old en kopieer de folder *htdocs* naar de folder c:\xampp
8. Als je goed is, heb je nu al je oude PHP-bestanden terug.

Run XAMPP als een Windows Service

Heb je (te) vaak) last van een corrupte database en moet je dan alles weer opnieuw installeren?

Allereerst zorg ervoor dat je XAMPP onder administratieve rechten draait, [hierboven](#) staat hoe dat moet.

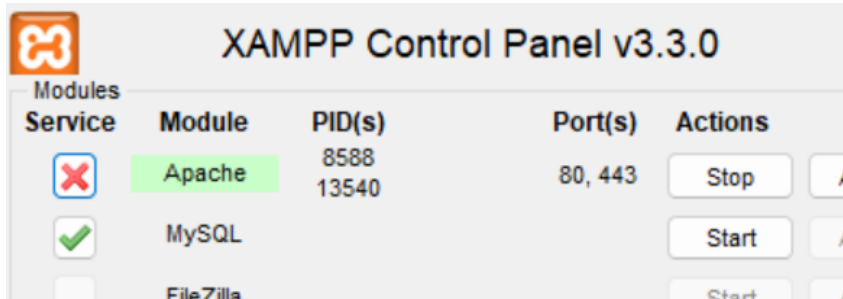
Daarna kan het helpen als je jouw MySQL als een **Windows Service** draait.

Het draaien van de database als een Windows-service verbetert de stabiliteit en betrouwbaarheid. De service wordt uitgevoerd in een afzonderlijk proces en is geïsoleerd van andere gebruikersprocessen. Als er bijvoorbeeld een probleem optreedt in een gebruikerssessie, heeft dit geen invloed op de service en blijft de database operationeel.

Dat wil ik ook, hoe moet dat!?

Heel eenvoudig.

1. Start XAMPP maar doe dat via je rechter muis knop en start het programma als administrator. Zoek naar XAMPP, rechtermuis en dan zie je de optie 'Run as Administrator' (bij mij is het de 1ste optie).
2. Start je database niet op, en als die wel draait dan stop je hem.
3. Click op het rode kruisje voor MySQL en zorg dat het een groen vinkje wordt.



4. Klaar, happy coding!

Voorbeeld vhosts in XAMPP

C:\xampp\apache\conf\extra\httpd-vhosts.conf

```
<VirtualHost *:80>
    ServerName localhost
    DocumentRoot "C:\Users\...\www"
</VirtualHost>

<VirtualHost *:80>
    ServerName test
    DocumentRoot "C:\Users\...\www\test"
</VirtualHost>

<VirtualHost *:80>
    ServerName cmon.local
    ServerAlias c21.cmon.local c22.cmon.local c20.cmon.local
    DocumentRoot "C:\Users\...\www\yii2\...\web"
    RewriteEngine on
    RewriteCond %{SERVER_NAME} =c22.cmon.ovh [OR]
    RewriteCond %{SERVER_NAME} =cmon.ovh [OR]
    RewriteCond %{SERVER_NAME} =c21.cmon.ovh
    RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI} [END,NE,R=permanent]
```

```
</VirtualHost>
```

Eventueel the document root in de (standaard) httpd-conf aanpassen.

```
DocumentRoot "C:\Users\...\www"
```

```
<Directory "C:\Users\...\www">
```

..

Wat is een database?

Opdracht

We beginnen deze les met huiswerk.

Zoek op internet op:

- Zoek op wat een database is
- Geef voorbeelden van een databases; noem een paar soorten (namen).
- Noem een drie voorbeelden van het gebruik van een database.
- Geef bij elk van de drie voorbeelden aan wat je in de database voor informatie zou stoppen.

--

Deze les is gegeven aan de hand van een Powerpoint presentatie.

Wat weet je na deze les?

- wat een database is
- dat er verschillende soorten databases zijn
- wat SQL is
- wat is PHPMyAdmin?
- wat zijn de basis-elementen van een database?
- aanmaken database

Wat is een database?

In de les wordt uitgelegd dat een database een gegevensverzameling is. Je harddisk en folder structure is in feite een database, maar een tekst-file kan je ook als database zien.

Er worden kort de volgende soorten databases uitgelegd:

- Flat file (txt of cvs)

- Hierarchisch (bijvoorbeeld file systeem)
- Netwerk
- Relatieveel RDBMS
- Object database

Een RDBMS is gebouwd op basis van relaties tussen de verschillende data eilandjes (tables).

SQL, Structured Query language is de taal waarmee je met de database praat in een RDBMS

Een RDBMS bestaat uit de volgende fysieke (hoofd)onderdelen:

1. **databases**

Dit is een verzameling tabellen die logisch bij elkaar horen.

2. **tables**

Dit is een groepje gegevens die bij iets horen, bijvoorbeeld bij de table persoon of klaslokaal.

3. **columns** (fields)

Dit zijn de gegevens die je van een tabel wilt vastleggen, bijvoorbeeld de naam van een persoon of het nummer van een klaslokaal.

4. **rows** (records)

Dit zijn de gegevens zelf. Bijvoorbeeld alle personen in de table persoon of alle klaslokalen in de table klaslokaal.

We gaan XAMPP installeren en maken kennis met PHPMyAdmin.

Nieuwe database aanmaken

Hier wordt kort uitgelegd hoe je met phpmyadmin (Engelstalig) een database moet aanmaken, een tabel moet maken en een table kan vullen.

Maak database

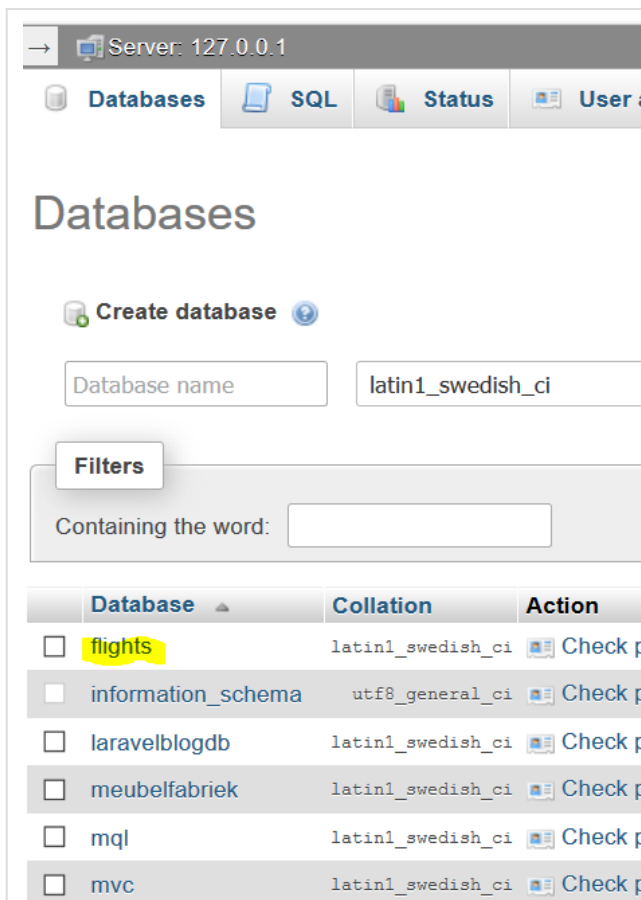


The screenshot shows the 'Databases' tab in phpMyAdmin. At the top, there are navigation tabs: 'Databases' (highlighted), 'SQL', 'Status', 'User accounts', and 'Export'. Below the tabs, the 'Databases' section is visible. It includes a 'Create database' link with a question mark icon. Below this, there is a form with a 'Database name' input field (highlighted in yellow) and a 'Collation' dropdown menu set to 'latin1_swedish_ci'. A 'Create' button is to the right of the form.

Zorg dat je onder het tabje database staat en vul de nieuwe databasenaam in en druk op create.

Noem je database *mijntest*.

Maak tabel



The screenshot shows the 'Databases' tab in phpMyAdmin. At the top, there are navigation tabs: 'Databases' (highlighted), 'SQL', 'Status', and 'User accounts'. Below the tabs, the 'Databases' section is visible. It includes a 'Create database' link with a question mark icon. Below this, there is a form with a 'Database name' input field and a 'Collation' dropdown menu set to 'latin1_swedish_ci'. Below the form, there is a 'Filters' section with a 'Containing the word:' input field. Below the filters, there is a table listing databases.

	Database	Collation	Action
<input type="checkbox"/>	flights	latin1_swedish_ci	Check p
<input type="checkbox"/>	information_schema	utf8_general_ci	Check p
<input type="checkbox"/>	laravelblogdb	latin1_swedish_ci	Check p
<input type="checkbox"/>	meubelfabriek	latin1_swedish_ci	Check p
<input type="checkbox"/>	mql	latin1_swedish_ci	Check p
<input type="checkbox"/>	mvc	latin1_swedish_ci	Check p

Ga eerst naar de database *mijntest* door op de juiste database te klikken.

Server: 127.0.0.1 » Database: flights

Structure SQL Search Query Export

Filters

Containing the word:

Table	Action
<input type="checkbox"/> airlines	★ Browse Structure Search Insert Empty
<input type="checkbox"/> airports	★ Browse Structure Search Insert Empty
<input type="checkbox"/> flights	★ Browse Structure Search Insert Empty

3 tables Sum

☐ Check all With selected:

Print Data dictionary

Create table

Name: Number of columns:

Vul de tabelnaam in en geef ook gelijk het aantal kolommen aan. Rechtsonder staat een knopje 'Go' of 'Start' (niet zichtbaar op dit plaatje).

Kies als tabelnaam *product*.

Name	Type	Length/Values
<input type="text"/>	INT	<input type="text"/>
<input type="text"/>	VARCHAR	<input type="text"/>

Pick from Central Columns

Vul de kolomnamen in en specificeer het type. Bij de een varchar moet je ook de lengte invullen. Als je klaar bent dan druk je op 'save' rechtsonder, niet zichtbaar op dit plaatje.

Maak de kolommen ID (int), Naam (varchar(20)), getal (int) en bedrag (int).

Vul tabel

Server: 127.0.0.1 » Database: flights » Table: qqq

Browse Structure SQL Search Insert Export

Column	Type	Function	Null	Value
ID	int(11)	<input type="text"/>	<input type="text"/>	<input type="text"/>
Naam	varchar(20)	<input type="text"/>	<input type="text"/>	<input type="text"/>
getal	int(11)	<input type="text"/>	<input type="text"/>	<input type="text"/>
bedrag	int(11)	<input type="text"/>	<input type="text"/>	<input type="text"/>


Go

Ga nu naar het tabje insert en vul de waarden in. De kolom function is niet nodig; je hoeft alleen de values in te vullen en dan op druk je op 'go'. Herhaal dit voor elke rij die je wilt toevoegen.

MyFirst Database

We gaan onze eerste database maken en we gaan queries in SQL uitvoeren.

Zorg ervoor dat je Apache Server en Database server draait:



The screenshot shows a window titled 'modules' with a table of services. The table has columns: Service, Module, PID(s), Port(s), and Actions. There are three rows: Apache (PID 14712, 4256, Port 80, 443), MySQL (PID 5764, Port 3306), and FileZilla. The 'Stop' button for MySQL is highlighted with a blue dashed border.

Service	Module	PID(s)	Port(s)	Actions
	Apache	14712 4256	80, 443	Stop Adm
	MySQL	5764	3306	Stop Adm
	FileZilla			Start Adm

We gebruiken hiervoor vluchtgegevens uit Amerika uit 2015. De database bestaat uit drie tabellen:

1. Airlines, dit zijn de verschillende maatschappijen die in de USA vliegen
2. Airports, een overzicht van alle vliegvelden van de USA (in 2015)
3. Flights, de eerste 5000 vluchten van 2015. het volledige bestand is ook beschikbaar maar dit bestand is vrij groot (~6 mio rows).

Naamgeving database objecten

Database objecten, databasenaam, tabellen en columnen worden meestal in lower case (kleine letters) geschreven. Als een object een naam krijgt dat uit twee woorden bestaat dan worden deze woorden door een _ (underscore) gescheiden. Bij voorkeur hanteer je Engelse namen. Op deze manier kunnen jou niet-Nederlands collega's jouw code makkelijker onderhouden.

Ga naar [PHPMyAdmin](#)

Onder het tabje database kun je een nieuwe database maken, doe dat en noem deze `flights`.

Zorg dat je de database hebt geopend, dat kan je zien doordat je boven in het beeld je databasenaam ziet:

Creer een tabel en noem die `airlines` (alleen kleine letters).

De tabel airlines bestaat uit twee columnen:

1. `code` met het data-type char(2)
2. `airline` met het data-type varchar(40)

De tabel `airports` bestaat uit seven columnen:

1. `iata_code` met het data-type `char(3)`
2. `airport` met data-type `varchar(80)`
3. `city` met data-type `varchar(40)`
4. `state` met data-type `char(2)`
5. `country` met data-type `char(4)`
6. `latitude` met data-type `float`
7. `longitude` met data-type `float`

Maar deze tabel ook aan.

Zorg dat je in de juiste tabel staat:

In het voorbeeld hierboven is de tabel `airports` actief. Zoek de bijbehorende CSV file en importeer deze. Hiervoor kun je via het tabje import de juiste file selecteren. Zorg ervoor dat je het juiste formaat hebt geselecteerd in dit geval CSV.

CSV is een text file waarin de regels (rows) zijn opgedeeld in kolommen (columns) door deze met komma's te scheiden. De strings staan bovendien tussen quotes, bijvoorbeeld:

```
IATA_CODE,AIRPORT,CITY,STATE,COUNTRY,LATITUDE,LONGITUDE
ABE,Lehigh Valley International Airport,Allentown,PA,USA,40.65236,-75.44040
ABI,Abilene Regional Airport,Abilene,TX,USA,32.41132,-99.68190
ABQ,Albuquerque International Sunport,Albuquerque,NM,USA,35.04022,-106.60919
ABR,Aberdeen Regional Airport,Aberdeen,SD,USA,45.44906,-98.42183
```

In dit voorbeeld staan de eerste 4 regels van het bestand `airports`. Let op dat de eerste regel ook een kopregel is. Deze moet niet in de database worden geïmporteerd. Indien dit wel is gebeurd dan kun je deze weer verwijderen.

Controleer of je de kopregel hebt geïmporteerd en als dat zo is verwijder deze. Probeer zelf uit te vinden hoe dit moet. Zolang je in de database `flights` blijft kan er weinig mis gaan want je kunt altijd de tabellen opnieuw aanmaken en de data opnieuw importeren.

Gefeliciteerd, je eerste database met twee tabellen is aangemaakt. Nu gaan we in de volgende les een paar queries maken.

Maak database en tabellen

Hier wordt kort uitgelegd hoe je met phpmyadmin (Engelstalig) een database moet aanmaken, een tabel moet maken en een table kan vullen.

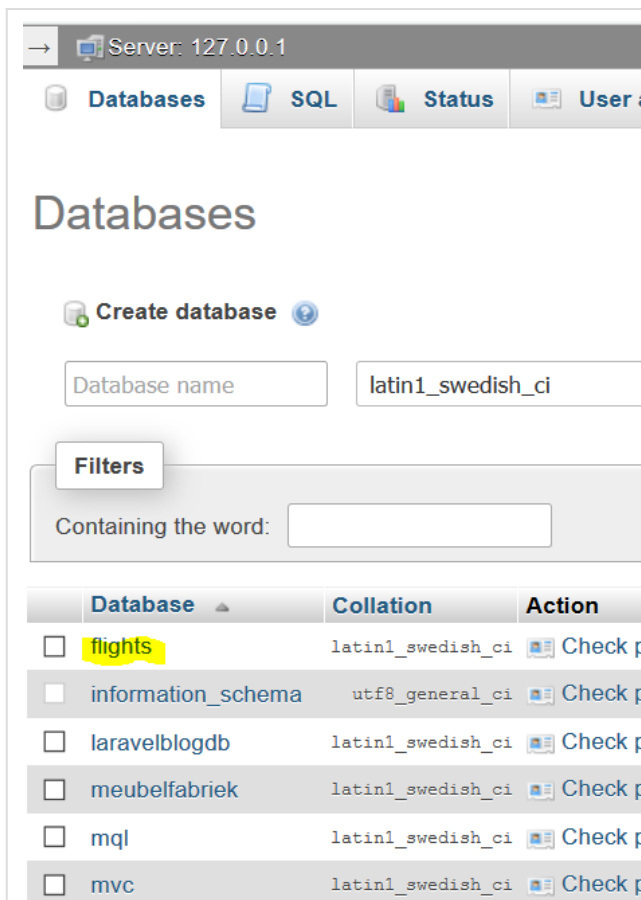
Maak database



The screenshot shows the 'Databases' tab in phpMyAdmin. The 'Create database' section is visible, with a text input field for the 'Database name' and a dropdown menu for the 'Collation' set to 'latin1_swedish_ci'. A 'Create' button is located to the right of the dropdown.

Zorg dat je onder het tabje database staat en vul de nieuwe databasenaam in en druk op create.

Maak tabel



The screenshot shows the 'Databases' tab in phpMyAdmin. Below the 'Create database' section, there is a 'Filters' section with a search bar. Below that, a table lists the existing databases. The 'flights' database is highlighted.

	Database	Collation	Action
<input type="checkbox"/>	flights	latin1_swedish_ci	Check p...
<input type="checkbox"/>	information_schema	utf8_general_ci	Check p...
<input type="checkbox"/>	laravelblogdb	latin1_swedish_ci	Check p...
<input type="checkbox"/>	meubelfabriek	latin1_swedish_ci	Check p...
<input type="checkbox"/>	mql	latin1_swedish_ci	Check p...
<input type="checkbox"/>	mvc	latin1_swedish_ci	Check p...

Ga eerst naar de juiste database, bijvoorbeeld naar flights:

Server: 127.0.0.1 » Database: flights

Structure SQL Search Query Export

Filters

Containing the word:

Table	Action
<input type="checkbox"/> airlines	★ Browse Structure Search Insert Empty
<input type="checkbox"/> airports	★ Browse Structure Search Insert Empty
<input type="checkbox"/> flights	★ Browse Structure Search Insert Empty

3 tables Sum

☐ Check all With selected:

Print Data dictionary

Create table

Name: Number of columns:

Vul de tabelnaam in en geef ook gelijk het aantal kolommen aan. Rechtsonder staat een knopje 'Go' of 'Start' (niet zichtbaar op dit plaatje).

Name	Type	Length/Values
<input type="text"/>	INT	<input type="text"/>
<input type="text"/>	VARCHAR	<input type="text"/>

Pick from Central Columns

Vul de kolomnamen in en specificeer het type. Bij de een varchar moet je ook de lengte invullen. Als je klaar bent dan druk je op 'save' rechtsonder, niet zichtbaar op dit plaatje.

Vul tabel

Server: 127.0.0.1 » Database: flights » Table: qqq

Browse Structure SQL Search Insert Export

Column	Type	Function	Null	Value
ID	int(11)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
Naam	varchar(20)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
getal	int(11)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
bedrag	int(11)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

Go

Ga nu naar het tabje insert en vul de waarden in. De kolom function is niet nodig; je hoeft alleen de vlues in te vullen en dan op druk je op 'go'. Herhaal dit voor elke rij die je wilt toevoegen.

Opgaven, mijn eerste Queries

Zelfstudie w3schools

Bestudeer https://www.w3schools.com/sql/sql_select.asp en probeer zelf of je een bijvoorbeeld een lijst kan krijgen met alle vliegvelden uit de airports tabel.

Zoek uit wat count(*) https://www.w3schools.com/sql/sql_count_avg_sum.asp

Opgaven

1. Hoeveel vliegvelden zijn er in de staat Pennsylvania (code PA)?
(noteer de query en het aantal vliegvelden in PA)
2. Hoeveel vliegvelden zijn er in de staat New York (code NY)?
(noteer de query en het aantal vliegvelden in NY)
3. ~~Hoeveel vliegvelden zijn er in de staat NY waarvan de iata_code groter dan 99 is?~~
~~(noteer de query en het aantal)~~
Hoeveel vliegvelden zijn er waarvan de iata_code met een 'A' begint. Tip: kijk even naar de sheets terug hoe het ook alweer precies zat met like.
4. Hoeveel vliegvelden zijn er die in de staat New York, Pennsylvania of California liggen?
(Bepaal dit in één query en noteer de query, en het aantal)

De latitude is de noord-zuid positie op de aarde. 0 is op de evenaar (zuiden) en 90 is de noordpool. Zoek op internet hoe de *order-by* functie werkt.

Zoek zelf op <https://www.w3schools.com/sql/> naar de juiste informatie.

5. Maak gebruik van de order-by functie en bepaal wat de meest zuidelijke plaats is waar op gevlogen is.
6. Maak gebruik van de order-by functie en bepaal wat de meest noordelijke plaats is waar op gevlogen is.

Bestudeer hoe de like functie werkt (select * from airlines wherelike)

7. Met welke query kun je alle airports vinden die met de letter 'a' beginnen?

Bestudeer hoe de *distinct* functie werkt (je kunt weer naar w3schools).

8. Maak een query die alle staten waar op gevlogen wordt in de US uit de tabel airports laat zien.

Elke staat moet precies één keer worden afgedrukt.

Succes!

--

Flight Data - Import, nog meer queries

We hebben een bestand met alle vluchten uit Amerika. Hiervan hebben we de eerste 5000 vluchten in een CSV file gezet. Deze file heet flights5000.zip (zie Teams).

Omdat het vrij veel tijd kost om deze table te definiëren is daarvoor een SQL query gemaakt.

Als laatste importeren we de flights5000.csv file. Hoervoor moet je natuurlijk eerst de tabel aanmaken.

Dit kun je op dezelfde manier aanmaken als de andere twee tabellen. Je kunt ook onderstaand SQL script (SQL programm) uitvoeren.

```
CREATE TABLE `flights` (  
  `year` int(11) NOT NULL,  
  `month` int(11) NOT NULL,  
  `day` int(11) NOT NULL,  
  `day_of_week` int(11) NOT NULL,  
  `airline` char(2) NOT NULL,  
  `flight_number` int(11) NOT NULL,  
  `tail_number` char(8) NOT NULL,  
  `origin_airport` char(3) NOT NULL,  
  `destination_airport` char(3) NOT NULL,  
  `scheduled_departure` int(11) NOT NULL,  
  `departure_time` int(11) NOT NULL,  
  `departure_delay` int(11) NOT NULL,  
  `taxi_out` int(11) NOT NULL,  
  `wheels_off` int(11) NOT NULL,  
  `scheduled_time` int(11) NOT NULL,  
  `elapsed_time` int(11) NOT NULL,  
  `airtime` int(11) NOT NULL,  
  `distance` int(11) NOT NULL,  
  `wheels_on` int(11) NOT NULL,  
  `taxi_in` int(11) NOT NULL,  
  `scheduled_arival` int(11) NOT NULL,
```

```
`arrival_time` int(11) NOT NULL,  
`arrival_dealy` int(11) NOT NULL,  
`diverted` int(11) NOT NULL,  
`cancelled` int(11) NOT NULL  
)
```

Controleer of de tabel goed is aangemaakt in PHPMYAdmin.

Je kunt deze code uitvoeren door in de juiste databse (flights) onder het tabje SQL deze code (en alleen deze code) uit te voeren. Copy/paste deze code en druk op de knop 'Go' rechtsonder aan het scherm.

Als de tabel zonder fouten is aangemaakt kan de csv worden geïmporteerd. Voer dit uit.

Om te controleren of de import goed is gelopen voer je query uit die telt hoeveel vluchten er in de tabel 'flights' staan.

Extra opgaven:

- Zoek op wat de code is voor het vliegveld 'Los Angeles' en tel hoeveel vluchten er in de tabel flights staan die vertrokken zijn vanuit Los Angeles.
- Hoeveel vluchten eindigen in Los Angeles?
- Hoeveel vluchten heeft *Delta Air Lines Inc.* uitgevoerd?

Ga naar [W3 Schools](https://www.w3schools.com/sql/default.asp) en zoek uit hoe je getallen kunt optellen met een SQL query met de functie sum().

- Hoeveel departure delay (vertraging bij vertrek) is er in totaal van alle vluchten van de maatschappij *Delta Air Lines Inc.* ?
- Op welke dag van de week vindt er de meeste vertraging plaats (van alle 5000 vluchten)?

Deze laatste query kun je opdelen in 7 queries voor elke dag van de week één. Het kan ook handiger maar dit zullen we later behandelen. Als je echt (ver) vooruit wilt werken lees dan hoe de 'Group by' functie werkt. Geen paniek als dit te lastig is, in een latere les komen we hierop terug.

Data Types

In SQL en ook in veel computertalen zoals PHP, Python, C en Java, heb je te maken met datatypes en vergelijkingen.

Datatypes

Een computer wil graag weten wat voor soort gegevens je wilt opslaan. Stel je verteld de computer "Hallo"+" "+"Jaap!" dan betekent dit dat de computer dit (een beetje afhankelijk van de taal) dat kan zien als "Hallo Jaap!". De woorden worden door het + teken achter elkaar gezet. Maar wat als je de computer verteld $4+3$? Is dit 7 of is dit 43?

Het antwoord is dat dit ligt aan het datatype. Is het een getal of een string? Een string is een verzameling letters of cijfers en een getal is een verzameling getallen.

Getallen

Met getallen kun je rekenen. Om getallen efficiënt op te slaan wil de computer iets meer weten dan alleen maar of iets een getal moet zijn. Is het een geheel getal, of een gebroken getal (met een komma)? Is het een klein getal of heb je extra geheugen nodig om een extra groot getal op te slaan? Dit leg je vast door het data type te definiëren.

Eigenlijk zijn er twee hoofdtypen, de INT, Integer. Dit is een geheel getal van -2147483648 tot 2147483647. Op de integer zijn allemaal varianten, kleine integers, zeer kleine integers, grote integers of zeer grote integers. Je zou kunnen zeggen dat je altijd de grootste variant neemt. Dat zou best werken, maar het nadeel is dat het werken met grotere getallen langzamer is en meer geheugenruimte kost (intern geheugen en hard diskruimte).

Naast de integer hebben we de FLOAT. Dit is een getal met een decimaal. Ook hiervan heb je weet grotere en kleinere varianten.

Strings

Tegenover de getallen staan strings, dit zijn 'woorden' waar je niet mee kan rekenen. Een woord kan ook een getal zijn, maar een string zet je altijd tussen quotes om aan te geven dat het een string is. "551" is dus een string en daarmee kan niet worden gerekend. "Dit is een zinnetje" is een andere string. "\$12.14" is ook een string.

Strings kun je ook in allerlei formaten opslaan. De meest voorkomende zijn CHAR en VARCHAR.

CHAR

CHAR heeft een vaste lengte. Bijvoorbeeld CHAR(2). Met een CHAR kan heel snel worden gewerkt omdat elke string precies even lang is. CHAR(n) gebruik je vaak als alle strings even lang zijn, bijvoorbeeld postcode.

VARCHAR

VARCHAR heeft een variabele lengte met een maximum van N., Dus VARCHAR(100) is een string die een lengte heeft van 0,1,2,3,4,5,...100 een variabele lengte dus met een maximum van 100. Het voordeel van een varchar is dat die minder opslag kost dan een CHAR. Als je VARCHAR(100) definieert en maar 10 posities gebruikt dan worden niet alle 100 posities ook gebruikt. Het nadeel van een VARCHAR is dat die iets langzamer is bij het verwerken.

DATE

Een date wordt gebruikt om een datum, en tijd vast te leggen.

Alle datatypen die in MySQL worden gebruikt, kun je vinden op: <http://domoticx.com/mysql-database-datatypes/>

MySQL Main Data Types

SQL Numeric Data Types

Datatype	From	To
bit	0	1
tinyint	0	255
smallint	-32,768	32,767
int	-2,147,483,648	2,147,483,647
bigint	-9,223,372,036,854,775,808	9,223,372,036,854,775,807
decimal	$-10^{38} + 1$	$10^{38} - 1$
numeric	$-10^{38} + 1$	$10^{38} - 1$
float	$-1.79E + 308$	$1.79E + 308$
real	$-3.40E + 38$	$3.40E + 38$

SQL Date and Time Data Types

Datatype	Description
DATE	Stores date in the format YYYY-MM-DD
TIME	Stores time in the format HH:MI:SS
DATETIME	Stores date and time information in the format YYYY-MM-DD HH:MI:SS
TIMESTAMP	Stores number of seconds passed since the Unix epoch ('1970-01-01 00:00:00' UTC)
YEAR	Stores year in 2 digit or 4 digit format. Range 1901 to 2155 in 4-digit format. Range 70 to 69, representing 1970 to 2069.

SQL Character and String Data Types

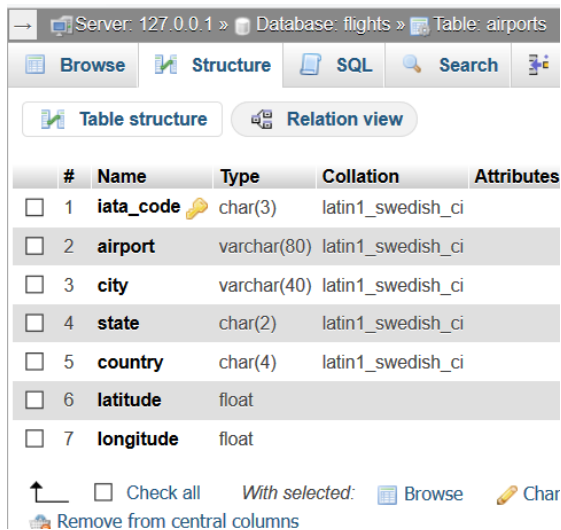
Datatype	Description
CHAR	Fixed length with maximum length of 8,000 characters

VARCHAR	Variable length storage with maximum length of 8,000 characters
VARCHAR(max)	Variable length storage with provided max characters, not supported in MySQL
TEXT	Variable length storage with maximum size of 2GB data

Mijn eerste queries - deel II

Dit zijn nog een paar opgaven die allemaal betrekking hebben op de tabel airports in de database flights.

Als het goed is ziet je tabel er zo uit:



#	Name	Type	Collation	Attributes
<input type="checkbox"/> 1	iata_code	char(3)	latin1_swedish_ci	
<input type="checkbox"/> 2	airport	varchar(80)	latin1_swedish_ci	
<input type="checkbox"/> 3	city	varchar(40)	latin1_swedish_ci	
<input type="checkbox"/> 4	state	char(2)	latin1_swedish_ci	
<input type="checkbox"/> 5	country	char(4)	latin1_swedish_ci	
<input type="checkbox"/> 6	latitude	float		
<input type="checkbox"/> 7	longitude	float		

☐ Check all With selected: ☐ Browse ☐ Char
☐ Remove from central columns

1. Voer nu de onderstaande query uit:
`SELECT count(*) FROM `airports``
Wat is het resultaat en wat betekent dat?
2. Voer de onderstaande query uit:
`SELECT count(*) FROM `airports` WHERE country = 'USA'`
wat is het resultaat, en wat betekent dat?
3. Vergelijk het antwoord van vraag 1 en vraag 2, wat valt je op en waarom is dat zo?
4. Met welke query krijg je een lijst van alle vliegvelden in de state Hawaii?
5. Heeft de staat Nevada (NV) meer of minder vliegvelden dan Hawaii? Met welke query onderzoek je dat?
6. Maar één query die alle vliegvelden op Hawaii en in Nevada laat zien (tip: denk aan OR).

7. Welk vliegveld heeft een naam die begint met 'Cherry'? Met welke query onderzoek je dit?

Database aanpassen

Tabelnaam aanpassen

```
RENAME TABLE `t1` TO `t2`;
```

(Use ` backticks if name contains spaces)

Kolomnaam aanpassen

Kolom aanpassen kan ook via PHPmyAdmin via de GUI. **Database**, In de regel van de tabel, **structure**, kolom aanvinken en **change** kiezen.

```
# Kolomnaam city hernoemen naar plaats.
```

```
ALTER TABLE airports  
CHANGE COLUMN city plaats varchar(40);
```

(Use ` backticks if name contains spaces)

Table airlines

```
DROP TABLE IF EXISTS `airlines`;  
CREATE TABLE `airlines` (  
  `code` char(2) NOT NULL,  
  `airline` varchar(40) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--
```

```
-- Dumping data for table `airlines`
```

```
--
```

```
INSERT INTO `airlines` (`code`, `airline`) VALUES  
( 'AA', 'American Airlines Inc.' ),  
( 'AS', 'Alaska Airlines Inc.' ),  
( 'B6', 'JetBlue Airways' ),
```

```
('DL', 'Delta Air Lines Inc.'),
('EV', 'Atlantic Southeast Airlines'),
('F9', 'Frontier Airlines Inc.'),
('HA', 'Hawaiian Airlines Inc.'),
('IA', 'AIRLINE'),
('MQ', 'American Eagle Airlines Inc.'),
('NK', 'Spirit Air Lines'),
('OO', 'Skywest Airlines Inc.'),
('Q1', 'Quorum Fly'),
('QQ', 'Quantum Airlines'),
('UA', 'United Air Lines Inc.'),
('US', 'US Airways Inc.'),
('VX', 'Virgin America'),
('WN', 'Southwest Airlines Co.');
```

Table airports

```
-- Database: `flights`
-- Table structure for table `airports`

DROP TABLE IF EXISTS `airports`;
CREATE TABLE `airports` (
  `iata_code` char(3) NOT NULL,
  `airport` varchar(80) NOT NULL,
  `city` varchar(40) DEFAULT NULL,
  `state` char(2) NOT NULL,
  `country` char(4) NOT NULL,
  `latitude` float NOT NULL,
  `longitude` float NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `airports`
--

INSERT INTO `airports` (`iata_code`, `airport`, `city`, `state`, `country`, `latitude`, `longitude`) VALUES
('ABE', 'Lehigh Valley International Airport', 'Allentown', 'PA', 'USA', 40.6524, -75.4404),
('ABI', 'Abilene Regional Airport', 'Abilene', 'TX', 'USA', 32.4113, -99.6819),
('ABQ', 'Albuquerque International Sunport', 'Albuquerque', 'NM', 'USA', 35.0402, -106.609),
```

('ABR', 'Aberdeen Regional Airport', 'Aberdeen', 'SD', 'USA', 45.4491, -98.4218),
('ABY', 'Southwest Georgia Regional Airport', 'Albany', 'GA', 'USA', 31.5355, -84.1945),
('ACK', 'Nantucket Memorial Airport', 'Nantucket', 'MA', 'USA', 41.2531, -70.0602),
('ACT', 'Waco Regional Airport', 'Waco', 'TX', 'USA', 31.6113, -97.2305),
('ACV', 'Arcata Airport', 'Arcata/Eureka', 'CA', 'USA', 40.9781, -124.109),
('ACY', 'Atlantic City International Airport', 'Atlantic City', 'NJ', 'USA', 39.4576, -74.5772),
('ADK', 'Adak Airport', 'Adak', 'AK', 'USA', 51.878, -176.646),
('ADQ', 'Kodiak Airport', 'Kodiak', 'AK', 'USA', 57.75, -152.494),
('AEX', 'Alexandria International Airport', 'Alexandria', 'LA', 'USA', 31.3274, -92.5486),
('AGS', 'Augusta Regional Airport (Bush Field)', 'Augusta', 'GA', 'USA', 33.37, -81.9645),
('AKN', 'King Salmon Airport', 'King Salmon', 'AK', 'USA', 58.6768, -156.649),
('ALB', 'Albany International Airport', 'Albany', 'NY', 'USA', 42.7481, -73.803),
('ALO', 'Waterloo Regional Airport', 'Waterloo', 'IA', 'USA', 42.5571, -92.4003),
('AMA', 'Rick Husband Amarillo International Airport', 'Amarillo', 'TX', 'USA', 35.2194, -101.706),
('ANC', 'Ted Stevens Anchorage International Airport', 'Anchorage', 'AK', 'USA', 61.1743, -149.996),
('APN', 'Alpena County Regional Airport', 'Alpena', 'MI', 'USA', 45.0781, -83.5603),
('ASE', 'Aspen-Pitkin County Airport', 'Aspen', 'CO', 'USA', 39.2232, -106.869),
('ATL', 'Hartsfield-Jackson Atlanta International Airport', 'Atlanta', 'GA', 'USA', 33.6404, -84.4269),
('ATW', 'Appleton International Airport', 'Appleton', 'WI', 'USA', 44.2574, -88.5195),
('AUS', 'Austin-Bergstrom International Airport', 'Austin', 'TX', 'USA', 30.1945, -97.6699),
('AVL', 'Asheville Regional Airport', 'Asheville', 'NC', 'USA', 35.4362, -82.5418),
('AVP', 'Wilkes-Barre/Scranton International Airport', 'Wilkes-Barre/Scranton', 'PA', 'USA', 41.3382, -75.7243),
('AZO', 'Kalamazoo/Battle Creek International Airport', 'Kalamazoo', 'MI', 'USA', 42.2349, -85.5521),
('BDL', 'Bradley International Airport', 'Windsor Locks', 'CT', 'USA', 41.9389, -72.6832),
('BET', 'Bethel Airport', 'Bethel', 'AK', 'USA', 60.7798, -161.838),
('BFL', 'Meadows Field', 'Bakersfield', 'CA', 'USA', 35.4336, -119.057),
('BGM', 'Greater Binghamton Airport', 'Binghamton', 'NY', 'USA', 42.2085, -75.9796),
('BGR', 'Bangor International Airport', 'Bangor', 'ME', 'USA', 44.8074, -68.8281),
('BHM', 'Birmingham-Shuttlesworth International Airport', 'Birmingham', 'AL', 'USA', 33.5629, -86.7535),
('BIL', 'Billings Logan International Airport', 'Billings', 'MT', 'USA', 45.8077, -108.543),
('BIS', 'Bismarck Municipal Airport', 'Bismarck', 'ND', 'USA', 46.7741, -100.747),
('BJI', 'Bemidji Regional Airport', 'Bemidji', 'MN', 'USA', 47.5094, -94.9337),
('BLI', 'Bellingham International Airport', 'Bellingham', 'WA', 'USA', 48.7928, -122.538),
('BMI', 'Central Illinois Regional Airport at Bloomington-Normal', 'Bloomington', 'IL', 'USA', 40.478, -88.9159),
('BNA', 'Nashville International Airport', 'Nashville', 'TN', 'USA', 36.1245, -86.6782),
('BOI', 'Boise Airport (Boise Air Terminal)', 'Boise', 'ID', 'USA', 43.5644, -116.223),
('BOS', 'Gen. Edward Lawrence Logan International Airport', 'Boston', 'MA', 'USA', 42.3643, -71.0052),
('BPT', 'Jack Brooks Regional Airport (Southeast Texas Regional)', 'Beaumont/Port Arthur', 'TX', 'USA', 29.9508, -94.0207),
('BQK', 'Brunswick Golden Isles Airport', 'Brunswick', 'GA', 'USA', 31.259, -81.4663),

('BQN', 'Rafael Hernández Airport', 'Aguadilla', 'PR', 'USA', 18.4949, -67.1294),
('BRD', 'Brainerd Lakes Regional Airport', 'Brainerd', 'MN', 'USA', 46.3979, -94.1372),
('BRO', 'Brownsville/South Padre Island International Airport', 'Brownsville', 'TX', 'USA', 25.9068, -97.4259),
('BRW', 'Wiley Post-Will Rogers Memorial Airport', 'Barrow', 'AK', 'USA', 71.2855, -156.766),
('BTM', 'Bert Mooney Airport', 'Butte', 'MT', 'USA', 45.9548, -112.497),
('BTR', 'Baton Rouge Metropolitan Airport', 'Baton Rouge', 'LA', 'USA', 30.5332, -91.1496),
('BTV', 'Burlington International Airport', 'Burlington', 'VT', 'USA', 44.473, -73.1503),
('BUF', 'Buffalo Niagara International Airport', 'Buffalo', 'NY', 'USA', 42.9405, -78.7322),
('BUR', 'Bob Hope Airport (Hollywood Burbank Airport)', 'Burbank', 'CA', 'USA', 34.2006, -118.358),
('BWI', 'Baltimore-Washington International Airport', 'Baltimore', 'MD', 'USA', 39.1754, -76.6682),
('BZN', 'Bozeman Yellowstone International Airport (Gallatin Field Airport)', 'Bozeman', 'MT', 'USA', 45.7769, -111.153),
('CAE', 'Columbia Metropolitan Airport', 'Columbia', 'SC', 'USA', 33.9388, -81.1195),
('CAK', 'Akron-Canton Regional Airport', 'Akron', 'OH', 'USA', 40.9163, -81.4425),
('CDC', 'Cedar City Regional Airport', 'Cedar City', 'UT', 'USA', 37.701, -113.099),
('CDV', 'Merle K. (Mudhole) Smith Airport', 'Cordova', 'AK', 'USA', 60.4918, -145.478),
('CEC', 'Del Norte County Airport (Jack McNamara Field)', 'Crescent City', 'CA', 'USA', 41.7802, -124.237),
('CHA', 'Chattanooga Metropolitan Airport (Lovell Field)', 'Chattanooga', 'TN', 'USA', 35.0353, -85.2038),
('CHO', 'Charlottesville-Albemarle Airport', 'Charlottesville', 'VA', 'USA', 38.1386, -78.4529),
('CHS', 'Charleston International Airport/Charleston AFB', 'Charleston', 'SC', 'USA', 32.8987, -80.0405),
('CID', 'The Eastern Iowa Airport', 'Cedar Rapids', 'IA', 'USA', 41.8846, -91.7109),
('CIU', 'Chippewa County International Airport', 'Sault Ste. Marie', 'MI', 'USA', 46.2508, -84.4724),
('CLD', 'McClellan-Palomar Airport', 'San Diego', 'CA', 'USA', 33.1272, -117.279),
('CLE', 'Cleveland Hopkins International Airport', 'Cleveland', 'OH', 'USA', 41.4109, -81.8494),
('CLL', 'Easterwood Airport', 'College Station', 'TX', 'USA', 30.5886, -96.3638),
('CLT', 'Charlotte Douglas International Airport', 'Charlotte', 'NC', 'USA', 35.214, -80.9431),
('CMH', 'Port Columbus International Airport', 'Columbus', 'OH', 'USA', 39.998, -82.8919),
('CMI', 'University of Illinois - Willard Airport', 'Champaign/Urbana', 'IL', 'USA', 40.0392, -88.2781),
('CMX', 'Houghton County Memorial Airport', 'Hancock', 'MI', 'USA', 47.1684, -88.4891),
('CNY', 'Canyonlands Field', 'Moab', 'UT', 'USA', 38.755, -109.755),
('COD', 'Yellowstone Regional Airport', 'Cody', 'WY', 'USA', 44.5202, -109.024),
('COS', 'City of Colorado Springs Municipal Airport', 'Colorado Springs', 'CO', 'USA', 38.8058, -104.7),
('COU', 'Columbia Regional Airport', 'Columbia', 'MO', 'USA', 38.8181, -92.2196),
('CPR', 'Natrona County International Airport', 'Casper', 'WY', 'USA', 42.9084, -106.464),
('CRP', 'Corpus Christi International Airport', 'Corpus Christi', 'TX', 'USA', 27.7704, -97.5012),
('CRW', 'Yeager Airport', 'Charleston', 'WV', 'USA', 38.3731, -81.5932),
('CSG', 'Columbus Metropolitan Airport', 'Columbus', 'GA', 'USA', 32.5163, -84.9389),
('CVG', 'Cincinnati/Northern Kentucky International Airport', 'Covington', 'KY', 'USA', 39.0461, -84.6622),
('CWA', 'Central Wisconsin Airport', 'Mosinee', 'WI', 'USA', 44.7776, -89.6668),
('DAB', 'Daytona Beach International Airport', 'Daytona Beach', 'FL', 'USA', 29.1799, -81.0581),

('DAL', 'Dallas Love Field', 'Dallas', 'TX', 'USA', 32.8471, -96.8518),
('DAY', 'James M. Cox Dayton International Airport', 'Dayton', 'OH', 'USA', 39.9024, -84.2194),
('DBQ', 'Dubuque Regional Airport', 'Dubuque', 'IA', 'USA', 42.403, -90.7092),
('DCA', 'Ronald Reagan Washington National Airport', 'Arlington', 'VA', 'USA', 38.8521, -77.0377),
('DEN', 'Denver International Airport', 'Denver', 'CO', 'USA', 39.8584, -104.667),
('DFW', 'Dallas/Fort Worth International Airport', 'Dallas-Fort Worth', 'TX', 'USA', 32.896, -97.0372),
('DHN', 'Dothan Regional Airport', 'Dothan', 'AL', 'USA', 31.3213, -85.4496),
('DIK', 'Dickinson Theodore Roosevelt Regional Airport', 'Dickinson', 'ND', 'USA', 46.7974, -102.802),
('DLG', 'Dillingham Airport', 'Dillingham', 'AK', 'USA', 59.0454, -158.503),
('DLH', 'Duluth International Airport', 'Duluth', 'MN', 'USA', 46.8421, -92.1936),
('DRO', 'Durango-La Plata County Airport', 'Durango', 'CO', 'USA', 37.1515, -107.754),
('DSM', 'Des Moines International Airport', 'Des Moines', 'IA', 'USA', 41.5349, -93.6607),
('DTW', 'Detroit Metropolitan Airport', 'Detroit', 'MI', 'USA', 42.2121, -83.3488),
('DVL', 'Devils Lake Regional Airport', 'Devils Lake', 'ND', 'USA', 48.1143, -98.9088),
('EAU', 'Chippewa Valley Regional Airport', 'Eau Claire', 'WI', 'USA', 44.8653, -91.4851),
('ECP', 'Northwest Florida Beaches International Airport', 'Panama City', 'FL', 'USA', 0, 0),
('EGE', 'Eagle County Regional Airport', 'Eagle', 'CO', 'USA', 39.6426, -106.918),
('EKO', 'Elko Regional Airport', 'Elko', 'NV', 'USA', 40.8249, -115.792),
('ELM', 'Elmira/Corning Regional Airport', 'Elmira', 'NY', 'USA', 42.1599, -76.8914),
('ELP', 'El Paso International Airport', 'El Paso', 'TX', 'USA', 31.8067, -106.378),
('ERI', 'Erie International Airport', 'Erie', 'PA', 'USA', 42.082, -80.1762),
('ESC', 'Delta County Airport', 'Escanaba', 'MI', 'USA', 45.7227, -87.0937),
('EUG', 'Eugene Airport (Mahlon Sweet Field)', 'Eugene', 'OR', 'USA', 44.1233, -123.219),
('EVV', 'Evansville Regional Airport', 'Evansville', 'IN', 'USA', 38.038, -87.5306),
('EWN', 'Coastal Carolina Regional Airport (Craven County Regional)', 'New Bern', 'NC', 'USA', 35.073, -77.0429),
('EWR', 'Newark Liberty International Airport', 'Newark', 'NJ', 'USA', 40.6925, -74.1687),
('EYW', 'Key West International Airport', 'Key West', 'FL', 'USA', 24.5561, -81.7596),
('FAI', 'Fairbanks International Airport', 'Fairbanks', 'AK', 'USA', 64.8137, -147.86),
('FAR', 'Hector International Airport', ' Fargo', 'ND', 'USA', 46.9193, -96.815),
('FAT', 'Fresno Yosemite International Airport', 'Fresno', 'CA', 'USA', 36.7762, -119.718),
('FAY', 'Fayetteville Regional Airport', 'Fayetteville', 'NC', 'USA', 34.9915, -78.88),
('FCA', 'Glacier Park International Airport', 'Kalispell', 'MT', 'USA', 48.3114, -114.255),
('FLG', 'Flagstaff Pulliam Airport', 'Flagstaff', 'AZ', 'USA', 35.1385, -111.671),
('FLL', 'Fort Lauderdale-Hollywood International Airport', 'Ft. Lauderdale', 'FL', 'USA', 26.0726, -80.1527),
('FNT', 'Bishop International Airport', 'Flint', 'MI', 'USA', 42.9655, -83.7435),
('FSD', 'Sioux Falls Regional Airport', 'Sioux Falls', 'SD', 'USA', 43.5813, -96.7417),
('FSM', 'Fort Smith Regional Airport', 'Fort Smith', 'AR', 'USA', 35.3366, -94.3674),
('FWA', 'Fort Wayne International Airport', 'Fort Wayne', 'IN', 'USA', 40.9785, -85.1952),
('GCC', 'Gillette-Campbell County Airport', 'Gillette', 'WY', 'USA', 44.3489, -105.539),
('GCK', 'Garden City Regional Airport', 'Garden City', 'KS', 'USA', 37.9275, -100.724),

('GEG', 'Spokane International Airport', 'Spokane', 'WA', 'USA', 47.6199, -117.534),
('GFK', 'Grand Forks International Airport', 'Grand Forks', 'ND', 'USA', 47.9493, -97.1761),
('GGG', 'East Texas Regional Airport', 'Longview', 'TX', 'USA', 32.3849, -94.7117),
('GJT', 'Grand Junction Regional Airport (Walker Field)', 'Grand Junction', 'CO', 'USA', 39.1224, -108.527),
('GNV', 'Gainesville Regional Airport', 'Gainesville', 'FL', 'USA', 29.6901, -82.2718),
('GPT', 'Gulfport-Biloxi International Airport', 'Gulfport-Biloxi', 'MS', 'USA', 30.4073, -89.0701),
('GRB', 'Green Bay-Austin Straubel International Airport', 'Green Bay', 'WI', 'USA', 44.4851, -88.1296),
('GRI', 'Central Nebraska Regional Airport', 'Grand Island', 'NE', 'USA', 40.9675, -98.3086),
('GRK', 'Killeen-Fort Hood Regional Airport', 'Killeen', 'TX', 'USA', 31.0649, -97.8278),
('GRR', 'Gerald R. Ford International Airport', 'Grand Rapids', 'MI', 'USA', 42.8808, -85.5228),
('GSO', 'Piedmont Triad International Airport', 'Greensboro', 'NC', 'USA', 36.0978, -79.9373),
('GSP', 'Greenville-Spartanburg International Airport', 'Greer', 'SC', 'USA', 34.8957, -82.2189),
('GST', 'Gustavus Airport', 'Gustavus', 'AK', 'USA', 58.4244, -135.707),
('GTF', 'Great Falls International Airport', 'Great Falls', 'MT', 'USA', 47.482, -111.371),
('GTR', 'Golden Triangle Regional Airport', 'Columbus-Starkville-West Point', 'MS', 'USA', 33.4503, -88.5914),
('GUC', 'Gunnison-Crested Butte Regional Airport', 'Gunnison', 'CO', 'USA', 38.534, -106.933),
('GUM', 'Guam International Airport', 'Agana', 'GU', 'USA', 13.4834, -144.796),
('HDN', 'Yampa Valley Airport (Yampa Valley Regional)', 'Hayden', 'CO', 'USA', 40.4812, -107.218),
('HIB', 'Range Regional Airport (Chisholm-Hibbing Airport)', 'Hibbing', 'MN', 'USA', 47.3866, -92.839),
('HLN', 'Helena Regional Airport', 'Helena', 'MT', 'USA', 46.6068, -111.983),
('HNL', 'Honolulu International Airport', 'Honolulu', 'HI', 'USA', 21.3187, -157.922),
('HOB', 'Lea County Regional Airport', 'Hobbs', 'NM', 'USA', 32.6875, -103.217),
('HOU', 'William P. Hobby Airport', 'Houston', 'TX', 'USA', 29.6454, -95.2789),
('HPN', 'Westchester County Airport', 'White Plains', 'NY', 'USA', 41.067, -73.7076),
('HRL', 'Valley International Airport', 'Harlingen', 'TX', 'USA', 26.2285, -97.6544),
('HSV', 'Huntsville International Airport', 'Huntsville', 'AL', 'USA', 34.6404, -86.7731),
('HYA', 'Barnstable Municipal Airport', 'Hyannis', 'MA', 'USA', 41.6693, -70.2804),
('HYS', 'Hays Regional Airport', 'Hays', 'KS', 'USA', 38.8449, -99.274),
('IAD', 'Washington Dulles International Airport', 'Chantilly', 'VA', 'USA', 38.9445, -77.4558),
('IAG', 'Niagara Falls International Airport', 'Niagara Falls', 'NY', 'USA', 43.1073, -78.9454),
('IAH', 'George Bush Intercontinental Airport', 'Houston', 'TX', 'USA', 29.9805, -95.3397),
('IAT', 'AIRPORT', 'CITY', 'ST', 'COUN', 0, 0),
('ICT', 'Wichita Dwight D. Eisenhower National Airport (Wichita Mid-Continent Airport)', 'Wichita', 'KS', 'USA', 37.65, -97.4331),
('IDA', 'Idaho Falls Regional Airport', 'Idaho Falls', 'ID', 'USA', 43.5146, -112.07),
('ILG', 'Wilmington Airport', 'Wilmington', 'DE', 'USA', 39.6787, -75.6065),
('ILM', 'Wilmington International Airport', 'Wilmington', 'NC', 'USA', 34.2706, -77.9026),
('IMT', 'Ford Airport', 'Iron Mountain/Kingsford', 'MI', 'USA', 45.8184, -88.1145),
('IND', 'Indianapolis International Airport', 'Indianapolis', 'IN', 'USA', 39.7173, -86.2944),
('INL', 'Falls International Airport', 'International Falls', 'MN', 'USA', 48.5662, -93.4031),

('ISN', 'Sloulin Field International Airport', 'Williston', 'ND', 'USA', 48.1779, -103.642),
('ISP', 'Long Island MacArthur Airport', 'Islip', 'NY', 'USA', 40.7952, -73.1002),
('ITH', 'Ithaca Tompkins Regional Airport', 'Ithaca', 'NY', 'USA', 42.491, -76.4584),
('ITO', 'Hilo International Airport', 'Hilo', 'HI', 'USA', 19.7203, -155.048),
('JAC', 'Jackson Hole Airport', 'Jackson', 'WY', 'USA', 43.6073, -110.738),
('JAN', 'Jackson-Evers International Airport', 'Jackson', 'MS', 'USA', 32.3112, -90.0759),
('JAX', 'Jacksonville International Airport', 'Jacksonville', 'FL', 'USA', 30.4941, -81.6879),
('JFK', 'John F. Kennedy International Airport (New York International Airport)', 'New York', 'NY', 'USA', 40.6398, -73.7789),
('JLN', 'Joplin Regional Airport', 'Joplin', 'MO', 'USA', 37.1518, -94.4983),
('JMS', 'Jamestown Regional Airport', 'Jamestown', 'ND', 'USA', 46.9297, -98.6782),
('JNU', 'Juneau International Airport', 'Juneau', 'AK', 'USA', 58.355, -134.576),
('KOA', 'Kona International Airport at Keahole', 'Kailua/Kona', 'HI', 'USA', 19.7388, -156.046),
('KTN', 'Ketchikan International Airport', 'Ketchikan', 'AK', 'USA', 55.3556, -131.714),
('LAN', 'Capital Region International Airport (Lansing Capital City)', 'Lansing', 'MI', 'USA', 42.7787, -84.5874),
('LAR', 'Laramie Regional Airport', 'Laramie', 'WY', 'USA', 41.312, -105.675),
('LAS', 'McCarran International Airport', 'Las Vegas', 'NV', 'USA', 36.0804, -115.152),
('LAW', 'Lawton-Fort Sill Regional Airport', 'Lawton', 'OK', 'USA', 34.5677, -98.4166),
('LAX', 'Los Angeles International Airport', 'Los Angeles', 'CA', 'USA', 33.9425, -118.408),
('LBB', 'Lubbock Preston Smith International Airport', 'Lubbock', 'TX', 'USA', 33.6636, -101.823),
('LBE', 'Arnold Palmer Regional Airport', 'Latrobe', 'PA', 'USA', 40.2759, -79.4048),
('LCH', 'Lake Charles Regional Airport', 'Lake Charles', 'LA', 'USA', 30.1261, -93.2234),
('LEX', 'Blue Grass Airport', 'Lexington', 'KY', 'USA', 38.037, -84.6054),
('LFT', 'Lafayette Regional Airport', 'Lafayette', 'LA', 'USA', 30.2053, -91.9877),
('LGA', 'LaGuardia Airport (Marine Air Terminal)', 'New York', 'NY', 'USA', 40.7772, -73.8726),
('LGB', 'Long Beach Airport (Daugherty Field)', 'Long Beach', 'CA', 'USA', 33.8177, -118.152),
('LIH', 'Lihue Airport', 'Lihue', 'HI', 'USA', 21.976, -159.339),
('LIT', 'Bill and Hillary Clinton National Airport (Adams Field)', 'Little Rock', 'AR', 'USA', 34.7294, -92.2243),
('LNK', 'Lincoln Airport (Lincoln Municipal)', 'Lincoln', 'NE', 'USA', 40.851, -96.7592),
('LRD', 'Laredo International Airport', 'Laredo', 'TX', 'USA', 27.5437, -99.4615),
('LSE', 'La Crosse Regional Airport', 'La Crosse', 'WI', 'USA', 43.8794, -91.2565),
('LWS', 'Lewiston-Nez Perce County Airport', 'Lewiston', 'ID', 'USA', 46.3745, -117.015),
('MAF', 'Midland International Airport', 'Midland', 'TX', 'USA', 31.9425, -102.202),
('MBS', 'MBS International Airport', 'Saginaw', 'MI', 'USA', 43.5329, -84.0797),
('MCI', 'Kansas City International Airport', 'Kansas City', 'MO', 'USA', 39.2976, -94.7139),
('MCO', 'Orlando International Airport', 'Orlando', 'FL', 'USA', 28.4289, -81.316),
('MDT', 'Harrisburg International Airport', 'Harrisburg', 'PA', 'USA', 40.1935, -76.7634),
('MDW', 'Chicago Midway International Airport', 'Chicago', 'IL', 'USA', 41.786, -87.7524),
('MEI', 'Meridian Regional Airport', 'Meridian', 'MS', 'USA', 32.3331, -88.7512),
('MEM', 'Memphis International Airport', 'Memphis', 'TN', 'USA', 35.0424, -89.9767),

('MFE', 'McAllen-Miller International Airport (McAllen Miller International)', 'McAllen', 'TX', 'USA', 26.1758, -98.2386),

('MFR', 'Rogue Valley International Airport', 'Medford', 'OR', 'USA', 42.3742, -122.873),

('MGM', 'Montgomery Regional Airport', 'Montgomery', 'AL', 'USA', 32.3006, -86.394),

('MHK', 'Manhattan Regional Airport', 'Manhattan', 'KS', 'USA', 39.141, -96.6708),

('MHT', 'Manchester-Boston Regional Airport', 'Manchester', 'NH', 'USA', 42.9345, -71.4371),

('MIA', 'Miami International Airport', 'Miami', 'FL', 'USA', 25.7932, -80.2906),

('MKE', 'General Mitchell International Airport', 'Milwaukee', 'WI', 'USA', 42.9472, -87.8966),

('MKG', 'Muskegon County Airport', 'Muskegon', 'MI', 'USA', 43.1695, -86.2382),

('MLB', 'Melbourne International Airport', 'Melbourne', 'FL', 'USA', 28.1028, -80.6458),

('MLI', 'Quad City International Airport', 'Moline', 'IL', 'USA', 41.4485, -90.5075),

('MLU', 'Monroe Regional Airport', 'Monroe', 'LA', 'USA', 32.5109, -92.0377),

('MMH', 'Mammoth Yosemite Airport', 'Mammoth Lakes', 'CA', 'USA', 37.6241, -118.838),

('MOB', 'Mobile Regional Airport', 'Mobile', 'AL', 'USA', 30.6914, -88.2428),

('MOT', 'Minot International Airport', 'Minot', 'ND', 'USA', 48.2594, -101.28),

('MQT', 'Sawyer International Airport', 'Marquette', 'MI', 'USA', 46.3536, -87.3954),

('MRY', 'Monterey Regional Airport (Monterey Peninsula Airport)', 'Monterey', 'CA', 'USA', 36.587, -121.843),

('MSN', 'Dane County Regional Airport', 'Madison', 'WI', 'USA', 43.1399, -89.3375),

('MSO', 'Missoula International Airport', 'Missoula', 'MT', 'USA', 46.9163, -114.091),

('MSP', 'Minneapolis-Saint Paul International Airport', 'Minneapolis', 'MN', 'USA', 44.8806, -93.2169),

('MSY', 'Louis Armstrong New Orleans International Airport', 'New Orleans', 'LA', 'USA', 29.9934, -90.258),

('MTJ', 'Montrose Regional Airport', 'Montrose', 'CO', 'USA', 38.5089, -107.894),

('MVY', 'Martha's Vineyard Airport', 'Marthas Vineyard', 'MA', 'USA', 41.393, -70.6143),

('MYR', 'Myrtle Beach International Airport', 'Myrtle Beach', 'SC', 'USA', 33.6797, -78.9283),

('OAJ', 'Albert J. Ellis Airport', 'Jacksonville', 'NC', 'USA', 34.8292, -77.6121),

('OAK', 'Oakland International Airport', 'Oakland', 'CA', 'USA', 37.7213, -122.221),

('OGG', 'Kahului Airport', 'Kahului', 'HI', 'USA', 20.8986, -156.43),

('OKC', 'Will Rogers World Airport', 'Oklahoma City', 'OK', 'USA', 35.3931, -97.6007),

('OMA', 'Eppley Airfield', 'Omaha', 'NE', 'USA', 41.3025, -95.8942),

('OME', 'Nome Airport', 'Nome', 'AK', 'USA', 64.5122, -165.445),

('ONT', 'Ontario International Airport', 'Ontario', 'CA', 'USA', 34.056, -117.601),

('ORD', 'Chicago O'Hare International Airport', 'Chicago', 'IL', 'USA', 41.9796, -87.9045),

('ORF', 'Norfolk International Airport', 'Norfolk', 'VA', 'USA', 36.8946, -76.2012),

('ORH', 'Worcester Regional Airport', 'Worcester', 'MA', 'USA', 42.2673, -71.8757),

('OTH', 'Southwest Oregon Regional Airport (North Bend Municipal)', 'North Bend', 'OR', 'USA', 43.4171, -124.246),

('OTZ', 'Ralph Wien Memorial Airport', 'Kotzebue', 'AK', 'USA', 66.8847, -162.599),

('PAH', 'Barkley Regional Airport', 'Paducah', 'KY', 'USA', 37.0608, -88.7738),

('PBG', 'Plattsburgh International Airport', 'Plattsburgh', 'NY', 'USA', 0, 0),

('PBI', 'Palm Beach International Airport', 'West Palm Beach', 'FL', 'USA', 26.6832, -80.0956),

('PDX', 'Portland International Airport', 'Portland', 'OR', 'USA', 45.5887, -122.598),
('PHF', 'Newport News/Williamsburg International Airport', 'Newport News', 'VA', 'USA', 37.1319, -76.493),
('PHL', 'Philadelphia International Airport', 'Philadelphia', 'PA', 'USA', 39.8719, -75.2411),
('PHX', 'Phoenix Sky Harbor International Airport', 'Phoenix', 'AZ', 'USA', 33.4342, -112.008),
('PIA', 'General Wayne A. Downing Peoria International Airport', 'Peoria', 'IL', 'USA', 40.6642, -89.6933),
('PIB', 'Hattiesburg-Laurel Regional Airport', 'Hattiesburg-Laurel', 'MS', 'USA', 31.4671, -89.3371),
('PIH', 'Pocatello Regional Airport', 'Pocatello', 'ID', 'USA', 42.9113, -112.596),
('PIT', 'Pittsburgh International Airport', 'Pittsburgh', 'PA', 'USA', 40.4915, -80.2329),
('PLN', 'Pellston Regional Airport of Emmet County', 'Pellston', 'MI', 'USA', 45.5709, -84.7967),
('PNS', 'Pensacola International Airport (Pensacola Gulf Coast Regional Airport)', 'Pensacola', 'FL', 'USA', 30.4733, -87.1874),
('PPG', 'Pago Pago International Airport (Tafuna Airport)', 'Pago Pago', 'AS', 'USA', 14.331, -170.711),
('PSC', 'Tri-Cities Airport', 'Pasco', 'WA', 'USA', 46.2647, -119.119),
('PSE', 'Mercedita Airport', 'Ponce', 'PR', 'USA', 18.0083, -66.563),
('PSG', 'Petersburg James A. Johnson Airport', 'Petersburg', 'AK', 'USA', 56.8017, -132.945),
('PSP', 'Palm Springs International Airport', 'Palm Springs', 'CA', 'USA', 33.8292, -116.506),
('PUB', 'Pueblo Memorial Airport', 'Pueblo', 'CO', 'USA', 38.2891, -104.497),
('PVD', 'Theodore Francis Green State Airport', 'Providence', 'RI', 'USA', 41.724, -71.4282),
('PWM', 'Portland International Jetport', 'Portland', 'ME', 'USA', 43.6462, -70.3087),
('RAP', 'Rapid City Regional Airport', 'Rapid City', 'SD', 'USA', 44.0453, -103.057),
('RDD', 'Redding Municipal Airport', 'Redding', 'CA', 'USA', 40.509, -122.293),
('RDM', 'Redmond Municipal Airport (Roberts Field)', 'Redmond', 'OR', 'USA', 44.2541, -121.15),
('RDU', 'Raleigh-Durham International Airport', 'Raleigh', 'NC', 'USA', 35.8776, -78.7875),
('RHI', 'Rhineland-Oneida County Airport', 'Rhineland', 'WI', 'USA', 45.6312, -89.4675),
('RIC', 'Richmond International Airport', 'Richmond', 'VA', 'USA', 37.5052, -77.3197),
('RKS', 'Rock Springs-Sweetwater County Airport', 'Rock Springs', 'WY', 'USA', 41.5942, -109.065),
('RNO', 'Reno/Tahoe International Airport', 'Reno', 'NV', 'USA', 39.4986, -119.768),
('ROA', 'Roanoke Regional Airport (Woodrum Field)', 'Roanoke', 'VA', 'USA', 37.3255, -79.9754),
('ROC', 'Greater Rochester International Airport', 'Rochester', 'NY', 'USA', 43.1189, -77.6724),
('ROW', 'Roswell International Air Center', 'Roswell', 'NM', 'USA', 33.3016, -104.531),
('RST', 'Rochester International Airport', 'Rochester', 'MN', 'USA', 43.9088, -92.498),
('RSW', 'Southwest Florida International Airport', 'Ft. Myers', 'FL', 'USA', 26.5362, -81.7552),
('SAF', 'Santa Fe Municipal Airport', 'Santa Fe', 'NM', 'USA', 35.6168, -106.088),
('SAN', 'San Diego International Airport (Lindbergh Field)', 'San Diego', 'CA', 'USA', 32.7336, -117.19),
('SAT', 'San Antonio International Airport', 'San Antonio', 'TX', 'USA', 29.5337, -98.4698),
('SAV', 'Savannah/Hilton Head International Airport', 'Savannah', 'GA', 'USA', 32.1276, -81.2021),
('SBA', 'Santa Barbara Municipal Airport (Santa Barbara Airport)', 'Santa Barbara', 'CA', 'USA', 34.4262, -119.84),
('SBN', 'South Bend International Airport (South Bend Regional)', 'South Bend', 'IN', 'USA', 41.709, -86.3185),
('SBP', 'San Luis Obispo County Regional Airport (McChesney Field)', 'San Luis Obispo', 'CA', 'USA', 35.2371, -120.642),

('SCC', 'Deadhorse Airport (Prudhoe Bay Airport)', 'Deadhorse', 'AK', 'USA', 70.1948, -148.465),
('SCE', 'University Park Airport', 'State College', 'PA', 'USA', 40.8512, -77.8463),
('SDF', 'Louisville International Airport (Standiford Field)', 'Louisville', 'KY', 'USA', 38.1744, -85.736),
('SEA', 'Seattle-Tacoma International Airport', 'Seattle', 'WA', 'USA', 47.449, -122.309),
('SFO', 'San Francisco International Airport', 'San Francisco', 'CA', 'USA', 37.619, -122.375),
('SGF', 'Springfield-Branson National Airport', 'Springfield', 'MO', 'USA', 37.2443, -93.3869),
('SGU', 'St. George Regional Airport', 'St George', 'UT', 'USA', 37.0906, -113.593),
('SHV', 'Shreveport Regional Airport', 'Shreveport', 'LA', 'USA', 32.4466, -93.8256),
('SIT', 'Sitka Rocky Gutierrez Airport', 'Sitka', 'AK', 'USA', 57.0471, -135.362),
('SJC', 'Norman Y. Mineta San José International Airport', 'San Jose', 'CA', 'USA', 37.3619, -121.929),
('SJT', 'San Angelo Regional Airport (Mathis Field)', 'San Angelo', 'TX', 'USA', 31.3577, -100.496),
('SJU', 'Luis Muñoz Marín International Airport', 'San Juan', 'PR', 'USA', 18.4394, -66.0018),
('SLC', 'Salt Lake City International Airport', 'Salt Lake City', 'UT', 'USA', 40.7884, -111.978),
('SMF', 'Sacramento International Airport', 'Sacramento', 'CA', 'USA', 38.6954, -121.591),
('SMX', 'Santa Maria Public Airport (Capt G. Allan Hancock Field)', 'Santa Maria', 'CA', 'USA', 34.8993, -120.458),
('SNA', 'John Wayne Airport (Orange County Airport)', 'Santa Ana', 'CA', 'USA', 33.6757, -117.868),
('SPI', 'Abraham Lincoln Capital Airport', 'Springfield', 'IL', 'USA', 39.8439, -89.6776),
('SPS', 'Wichita Falls Municipal Airport/Sheppard AFB', 'Wichita Falls', 'TX', 'USA', 33.9888, -98.4919),
('SRQ', 'Sarasota-Bradenton International Airport', 'Sarasota', 'FL', 'USA', 27.3953, -82.5541),
('STC', 'St. Cloud Regional Airport', 'St Cloud', 'MN', 'USA', 45.5453, -94.0583),
('STL', 'St. Louis International Airport at Lambert Field', 'St Louis', 'MO', 'USA', 38.7477, -90.36),
('STT', 'Cyril E. King Airport', 'Charlotte Amalie', 'VI', 'USA', 18.3373, -64.9734),
('STX', 'Henry E. Rohlsen Airport', 'Christiansted', 'VI', 'USA', 17.7019, -64.7986),
('SUN', 'Friedman Memorial Airport', 'Hailey', 'ID', 'USA', 43.5048, -114.297),
('SUX', 'Sioux Gateway Airport', 'Sioux City', 'IA', 'USA', 42.4026, -96.3844),
('SWF', 'Stewart International Airport', 'Newburgh', 'NY', 'USA', 41.5041, -74.1048),
('SYR', 'Syracuse Hancock International Airport', 'Syracuse', 'NY', 'USA', 43.1112, -76.1063),
('TLH', 'Tallahassee International Airport', 'Tallahassee', 'FL', 'USA', 30.3965, -84.3503),
('TOL', 'Toledo Express Airport', 'Toledo', 'OH', 'USA', 41.5868, -83.8078),
('TPA', 'Tampa International Airport', 'Tampa', 'FL', 'USA', 27.9755, -82.5332),
('TRI', 'Tri-Cities Regional Airport', 'Bristol', 'TN', 'USA', 36.4752, -82.4074),
('TTN', 'Trenton Mercer Airport', 'Trenton', 'NJ', 'USA', 40.2767, -74.8135),
('TUL', 'Tulsa International Airport', 'Tulsa', 'OK', 'USA', 36.1984, -95.8882),
('TUS', 'Tucson International Airport', 'Tucson', 'AZ', 'USA', 32.1161, -110.941),
('TVC', 'Cherry Capital Airport', 'Traverse City', 'MI', 'USA', 44.7414, -85.5822),
('TWF', 'Magic Valley Regional Airport (Joslin Field)', 'Twin Falls', 'ID', 'USA', 42.4818, -114.488),
('TXK', 'Texarkana Regional Airport (Webb Field)', 'Texarkana', 'AR', 'USA', 33.4537, -93.991),
('TYR', 'Tyler Pounds Regional Airport', 'Tyler', 'TX', 'USA', 32.3541, -95.4024),
('TYS', 'McGhee Tyson Airport', 'Knoxville', 'TN', 'USA', 35.8125, -83.9929),
('UST', 'Northeast Florida Regional Airport (St. Augustine Airport)', 'St. Augustine', 'FL', 'USA', 0, 0),

('VEL', 'Valdez Airport', 'Vernal', 'UT', 'USA', 40.4409, -109.51),
('VLD', 'Valdosta Regional Airport', 'Valdosta', 'GA', 'USA', 30.7825, -83.2767),
('VPS', 'Destin-Fort Walton Beach Airport/Eglin AFB', 'Valparaiso', 'FL', 'USA', 30.4832, -86.5254),
('WRG', 'Wrangell Airport', 'Wrangell', 'AK', 'USA', 56.4843, -132.37),
('WYS', 'Westerly State Airport', 'West Yellowstone', 'MT', 'USA', 44.6884, -111.118),
('XNA', 'Northwest Arkansas Regional Airport', 'Fayetteville/Springdale/Rogers', 'AR', 'USA', 36.2819, -94.3068),
('YAK', 'Yakutat Airport', 'Yakutat', 'AK', 'USA', 59.5034, -139.66),
('YUM', 'Yuma International Airport', 'Yuma', 'AZ', 'USA', 32.6566, -114.606);

SQL Queries met opdrachten

SQL

Als je gegevens uit een RDBMS wilt halen dan moet je gebruik maken van SQL (Strucutered Query Language).

SQL is een gestandaliseerde taal die voor alle relationele databases (zoals MySQL, MariaDB, Postgress, Oracle, Microsoft SQL server) hetzelfde is¹.

String of getal?

Strings staan tussen " en geven aan dat iets geen getal is. Tabel- en kolomnamen staan in principe niet tussen ". Indien tabelnamen of kolomnamen wel een spatie bevatten dan moeten die tussen speciale `` worden gezet. Dit zijn de zognaamde back-ticks, ene quote die achterover leunt.

Select

Als je een met SQL gegevens wilt raadplegen dan begin je altijd met *SELECT*.

Na de select komt altijd de *FROM*. De *FROM* geeft aan uit welke tabel je je gegevens haalt. Voorlopig halen we onze gegevens uit één tabel.

Als je niet alle rows wilt zien

dan moet je een *WHERE* toevoegen. Daarbij specificeer je dan welke rows je wilt zien.

Voorbeelden

Query	Uitleg
SELECT voornaam, achternaam FROM persoon WHERE voornaam = 'Skander'	Laat voornaam en achternaam zien van iedereen die als voornaam Skander heeft. Let op de " rondom Skander. Dit geeft aan dat Skander een string is.
SELECT count(*) FROM persoon WHERE voornaam like 'K%'	Count(*) laat alleen het aantal zien. Dus bij deze query worden alle personen geteld van wie de voornaam met een K begint.

SELECT count(*) FROM persoon WHERE voornaam like '%'	Deze query lijkt op de vorige maar hier staat alleen een '%' en dat betekent elk karakter nul één of meer keren. Dus hier worden alle voornamen geselecteerd. De query kan dus ook zonder WHERE kunnen worden uitgevoerd.
SELECT * FROM vakken WHERE score > 6	Laat alle informatie zien van alle vakken waarvoor geldt dat de score groter is dan 6. Let op dat 6 niet tussen '' staat; het is immers een getal.
SELECT count(*) FROM vakken WHERE score > 6 AND score < 8	Laat het aantal vakken zien waarvoor de score groter is dan 6 en kleiner is dan 8 (er dus tussenin ligt).
SELECT * FROM vakken WHERE score = 6 OR score = 8	Laat alle informatie van alle vakken zien waarvoor een score van een 6 of een 8 is behaald.
SELECT * FROM vakken WHERE vaknaam <> 'DB Introduction' ORDER BY vaknaam	Laat alle informatie zien van alle vakken behalve het vak 'DB Introduction' en sorteer de lijst van vakken op achternaam.

Opdrachten

Beschrijf nu wat de volgende queries doen:

Query Nr.	Query	Uitleg
1	SELECT voornaam, achternaam FROM persoon WHERE lengte = 192	
2	SELECT voornaam, achternaam FROM persoon WHERE lengte >= 192	
3	SELECT voornaam, achternaam FROM persoon WHERE lengte >= 180 AND lengte <= 190	
4	SELECT voornaam, achternaam FROM persoon WHERE lengte < 180 AND lengte > 190	
5	SELECT count(*) FROM persoon WHERE lengte < 180 AND lengte > 190	
6	SELECT voornaam, achternaam FROM persoon WHERE voornaam = 'Mo'	

7	SELECT voornaam, achternaam FROM persoon WHERE voornaam like 'Mo%'	
8	SELECT count(*) FROM persoon WHERE voornaam like 'Mo%'	
9	SELECT voornaam, achternaam FROM persoon WHERE voornaam like 'Mo%' OR voornaam like 'Ma%'	
10	SELECT voornaam, achternaam FROM persoon WHERE voornaam like 'Mo%' AND voornaam like 'Ma%'	
11	SELECT voornaam, achternaam FROM persoon WHERE voornaam = 'Mo' ORDER by achternaam	
12	SELECT voornaam, achternaam FROM persoon WHERE voornaam = 'Mo' ORDER by achternaam desc	

¹⁾ Er zijn tussen de verschillende databases soms wel kleine verschillen in SQL.

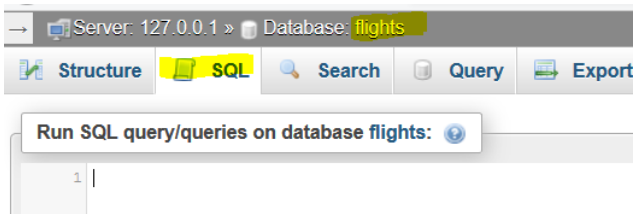
Huiswerk

week 6

Zorg ervoor dat XAMPP is geïnstalleerd en Apache en MySQL (de bovenste twee opties) aan staan.

Ga in de browser naar <http://127.0.0.1/myphpadmin>

Kies onder database de flight database en kies tabje SQL boven in je scherm staat nu:



Om er zeker voor te zijn dat je database goed is kun je de file *flights-dropDB-airlines_en_airports.sql* in Teams of hiernaast downloaden (onder files/bestanden) en deze draaien. Dat doe je door in phpmyadmi nonder het import tabje de file in te lezen en uit te voeren.

1. Voer de volgende query uit: `SELECT * FROM airlines`
Wat zie je en hoe heten de kolommen?
2. Voer de volgende query uit: `SELECT * FROM airlines WHERE code='AA'`
Wat zie je en hoe heet de Airline met de code AA?
3. Voer de volgende query uit: `SELECT * FROM airlines WHERE code='AA' OR code='OO'`
Hoeveel regels worden er getoond en wat zie je?
4. Voer de volgende query uit: `SELECT * FROM airlines WHERE code like 'A%' OR code like 'O%'`
Hoeveel regels worden er getoond en wat zie je?
5. Voer de volgende query uit: `SELECT * FROM airports`
Wat zie je en hoe heten de kolommen (onthoud deze kolommen voor later)?
6. Voer de volgende query uit: `SELECT * FROM airports WHERE city = 'Dallas'`
Wat zie je en wat is de code van de staat waarin Dallas ligt?
7. Voer de volgende query uit: `SELECT * FROM airports WHERE city = 'New York'`
Wat zie je? Hoeveel vliegvelden heeft New York?
8. Voer de volgende query uit: `SELECT state FROM airports where city = 'Las Vegas'`
Hoeveel kolommen zie je nu en waarom is dat?

9. Bedenk nu zelf een query die alle vliegvelden laat zien in de plaats Chicago
Schrijf de query op.
 10. Bedenk nu zelf een query die alle vliegvelden laat zien waarvan de naam begint met City
Hoeveel vliegvelden vind je? Schrijf de query en de uitkomst op.
 11. Hoeveel vliegvelden heeft de staat Nevada (code = NV)?
Schrijf de uitkomst (getal) en de query op.
 12. Hoeveel airlines (regels) zitten er totaal in de tabel airlines?
Weet je nog dat je hiervoor count(*) kan gebruiken? Schrijf de query op.
 13. Bonusvraag: Hoeveel airlines hebben een 'y' of een 'Y' in de naam?
Schrijf de query op.
-

week 5

Vertel wat de queries doen:

1. `SELECT achternaam FROM werknemers WHERE salaris > 2100 order by achternaam`
2. `SELECT achternaam FROM werknemers WHERE salaris >= 2100 AND salaris <= 2200`
3. `SELECT count(*) FROM werknemers WHERE salaris=0`
4. `SELECT * FROM vakken WHERE vaknaam = 'PHP'`
5. `SELECT vaknaam FROM vakken WHERE vaknaam like '%db%'`
6. `SELECT bestellingsnummer, bedrag FROM bestellingen WHERE besteldatum > '31-12-2018'`
7. `SELECT bestellingsnummer, bedrag FROM bestellingen WHERE besteldatum > '31-12-2018' AND bedrag <> 0`
8. `SELECT count(*) FROM reizen WHERE bestemming = 'Spanje' AND reisdatum >= '01-07-2018' AND reisdatum <= '31-07-2018'`
9. `SELECT count(*) FROM reizen WHERE reisdatum >= '01-07-2018' AND reisdatum <= '31-07-2018' AND vluchtijd > '18:00'`
10. `SELECT count(*) FROM reizen WHERE bestemming = 'Griekenland' OR bestemming = 'Malta'`
11. `SELECT count(*) FROM reizen WHERE bestemming = 'Griekenland' AND bestemming = 'Malta'`

12. SELECT count(*) FROM reizen WHERE bestemming <> 'Griekenland' AND bestemming <> 'Malta'

Joins (les 2, periode 2)

In deze les leer je hoe je de gegevens uit twee tabellen uit de deatabase kunt combineren

Uitleg

In een relationele database zitten meerdere tabellen en met behulp van relaties kun je deze gegevens combineren. In deze les leren we op een hele eenvoudige manier hoe je dat kunt doen.

Stel je hebt twee tabellen: *persoon* en *stad*.

De tabel *persoon* bestaat uit 4 rows en 5 kolommen.

ID	voornaam	achternaam	stad
1	Jane	Collinson	101
2	Mohamed	Lisram	103
3	Leo	Thompson	101
4	Montano	Drew	104
5	Greg	Lewinky	105

De tabel *stad* bestaat uit 5 rows en 2 kolommen

ID	naam
101	Amsterdam
102	Diemen
103	Abcoude
105	Utrecht
106	Bovenkerk

Vraag 1: weet jij waar Jane Collinson woont (goed kijken)?

Als je ziet waar Jane woont dan heb je waarschijnlijk gezien dat de kolom *stad* uit de tabel *persoon* verwijst naar de kolom *ID* uit de tabel *stad*. Of in het kort:

Primary Key en Foreign Key

De kolom `persoon.stad` verwijst naar een andere tabel dat heet een foreign key (vreemde key, een key van een andere tabel).

De kolom `ID` in de tabel `stad` is een primary key. Een hoofd sleutel. Een primary key heeft voor elke regel een unieke waarde en je kan er dus altijd precies één rij mee aanwijzen.

Een foreign key verwijst naar een aan andere primary key maar hoeft niet uniek te zijn. Kijk maar in het voorbeeld, Jane en Leo wonen allebij in stad 101 (Amsterdam) en hebben dus dezelfde foreign key.

Opdracht (in de les)

Maak een nieuwe database aan in phpmyadmin en noem de database `persoon`.

We gaan nu een database en twee tabellen maken in phpmyadmin.

1. Download het bestandje [persoon.sql](#) aan de linkerkant van deze pagina.
2. Start XAMPP (Apache en MySQL).
3. Kies in het linker gedeelte 'new' om een nieuwe database aan te maken.
4. Maak een database met de naam `persoon` aan.
5. Ga naar het tabje import en selecteer browse; ga naar het bestand `persoon.sql` en importeer deze.

(*optioneel*) Als je wilt kun je de tabellen ook handmatig aanmaken. Je moet daarvoor de volgende twee tabellen maken:

1. **persoon** met de kolommen **ID** (integer), **voornaam** (varchar(80)), **achternaam** (varchar(80)), en **stad** (integer)
2. **stad** met de kolommen **ID** (integer) en **naam** (varchar(80)).
3. vul de tabellen nu met de waarden zoals hierboven aangegeven.

Voer nu de volgende query uit:


```
SELECT *  
FROM persoon  
JOIN stad  
ON persoon.stad = stad.id
```

Opdracht (huiswerk)

Start XAMPP (Apache en SQL server) en ga naar phpmySQL <http://localhost/phpmyadmin/>. Ga naar de database persoon (linkerkant van scherm) en ga naar het tabje SQL.

Tip: voor opgave 4 t/m 8 moet je een join gebruiken.

1. Maak een query die alle plaatsnamen uit de database (persoon) laat zien.
Schrijf de query op je antwoordenblad.
2. Maak een query die alle voornamen en achternamen van alle personen uit de database afdrukt.
Schrijf de query op je antwoordenblad.
3. Maak een query die alle voornamen en achternamen laat zien van alle personen waarvan de voornaam met een M begint.
Schrijf de query op je antwoordenblad.
4. Maak een query van alle voornamen en achternamen en woonplaatsen van alle personen uit de database.
Schrijf de query op je antwoordenblad.
5. Maak een query die alle voornamen afdrukt van alle personen uit Amsterdam.
Zoek op op welke kolomnaam je moet zoeken; waar vind je de plaatsnaam en hoe heet die? Let ook op dat Amsterdam in de database met een kleine letter begint.
Schrijf de query op je antwoordenblad.
6. Maak een query die alle voornamen en achternamen afdrukt van alle personen uit Abcoude.
Schrijf de query op je antwoordenblad.
7. Maak een query die telt hoeveel personen er uit Amsterdam komen.
Schrijf de query op je antwoordenblad.

8. Maak een query die telt hoeveel personen er uit Amsterdam of Diemen komen.
Schrijf de query op je antwoordenblad.

--

Samenvatting - Leren!

Tabel, kolommen en regels (table, columns and rows).

Een tabel bevat informatie over iets, bijvoorbeeld over een persoon. Een **tabel** heet bijvoorbeeld persoon want het bevat gegevens over een persoon.

voornaam	achternaam	adres	woonplaats	telefoonnummer
Anouar	Jackson	De Kluivert 3	Amsterdam	06 3434 1212
Jessie	Loo	Grote Markt 1a	Amsterdam	06 2125 7845
Maggie	Mellow	Kleine Gracht 12	Amsterdam	06 1035 2900

Deze table heeft 5 **kolommen/columns**, voornaam, achternaam, adres, woonplaats en telefoonnummer. Dit zijn allemaal eigenschappen van de persoon.

Deze table heeft drie **regels/rows**. Elke regel bevat informatie over één persoon.

SQL Query

Met SQL kun je de database een vraag stellen, de meest eenvoudige query ziet er uit als volg:

```
SELECT ...  
FROM tabelnaam  
WHERE ...
```

Na de select

Na de SELECT	
één of meer kolomnamen	toont de inhoud van de kolomnamen
*	toont de inhoud van alle kolommen van deze tabel
count(*)	laat zien hoeveel regels er geselecteerd zijn

Na de where

Na de where kun je een vergelijking of selectie maken, bijvoorbeeld:

Na de WHERE	
-------------	--

kolomnaam = 'xyz'	Toont alle regels waarvan de kolom de 'xyz' heeft (waarde is string).
kolomnaam like 'x%'	Toont alle regels waarvan de kolom met x begint.
kolomnaam like '%x%'	Toont alle regels waarvan de kolom een x bevat.
kolomnaam like '%x'	Toont alle regels waarvan de kolom met x eindigt.
kolomnaam = 12	Toont alle regels waarvan de kolom 12 is (waarde is nummer).
kolmnaam > 12	Toont alle regels waarvan de kolom groter dan 12 is.
kolmnaam < 12	Toont alle regels waarvan de kolom kleiner dan 12 is.
kolmnaam >= 12	Toont alle regels waarvan de kolom groter dan of gelijk aan 12 is.
kolmnaam <= 12	Toont alle regels waarvan de kolom kleiner dan of gelijk 12 is.
kolomnaam <> 12	Toont alle regels waarvan de kolom ongelijk aan 12 is.

Na de where kun je ook **twee** vergelijkingen hebben, deze moet je dan met een AND of een OR aan elkaar koppelen, bijvoorbeeld:

AND - OR	
kolom >12 AND kolom < 20	Kolom zit tussen 12 en 20; is groter dan 12 AND kleiner dan 20.
kolom <12 OR kolom > 20	Kolom is kleiner dan 12 OR kolom is groter dan 20.
naam like 'A%' OR naam like 'B%'	de kolom naam begint met een A OR met een B.
naam like 'A%' AND naam like 'B%'	de kolom naam begint met een A AND met een B; <i>dit kan niet!</i>

Resultaat sorteren

Bij een select kun je de resultaten sorteren. Dat doe je door een ORDER BY aan het einde van de query toe te voegen.

Na de ORDER BY geef je de kolomnaam waarop je wilt sorteren. Sorteren kan op twee manieren:

ORDER BY	Betekent	Sorteer volgorde
ASC	Ascending (stijgen)	Oplopend: A..Z of 0..9
DESC	Descending (dalen)	Aflopend: Z..A of 9..0

Voorbeeld:

```
SELECT voornaam, achternaam  
FROM leerling  
ORDER BY achternaam ASC
```

(sorteer de namen op achternaam oplopend)

That's all!

Lesplan DB Intro (8ste les)

Klas: **OITIBO9B** 10:45 - 11:45, lokaal 251

Dit is een doorstroomklas van niveau 3 en dit is de eerste keer dat deze lessenserie wordt gegeven. Voor mij was het erg zoeken naar het instapniveau.

Achtergrond

In de laatste les is er een test afgenomen. De resultaten waren laag. Gemiddeld een 5.5 nadat de norm is aangepast van 1-10 naar 3-10

Lesdoel

Doel is om de klas voor te bereiden voor de test van volgende week.

Mijn leerdoel

Klassenmanagement is een uitdaging in deze groep. Ik hoor graag hoe ik dat beter kan doen.

Lesplanning

De kern van de les zal zijn het eerste zelf doornemen van de leerstof. Dit is en samenvatting van alle behandelde theorie. Vervolgens gaan we aan de hand van de theorie zelf de test nakijken. We doen dit per groepje vragen (2-5) maar ik laat dit afhangen van de planning.

0 - 7 m.	Binnenkomst, presentielijst, rondlopen.
7 - 10 m.	Uitleggen wat we gaan doen.
10-15 m.	Zelf doorlezen theorie
15 - 55 m. (~40m.)	6x3 vragen (5 min) plus 4 open vragen (10 min)
55 - 60 m.	Afronden

--

Huiswerk is theorie leren.

De Wereld-Database

In deze les gaan we een database met drie tabellen gebruiken. De database bevat een tabel city (plaats), een tabel country (land) en een tabel countrylanguage (taal van het land).

We downloaden de file [world.zip](#) en unzippen het bestand. Je ziet een nieuw bestand met de naam world.sql. Onthoud waar het staat.

We maken een database zoals we dat eerder hebben gedaan. We starten XAMPP Apache en MySQL en gaan naar 127.0.0.1/phpmyadmin. We maken daar een nieuwe database en noemen die world (wereld). Daarna importeren we het script world.sql. Na een paar seconden is de database aangemaakt.

Laten we met controleren of de database goed is aangemaakt.

Ga naar het tabje SQL in phpmyadmin en voer de volgende queries uit. De getallen tussen haakjes zijn de getallen die je zou moeten zien als je de query uitvoert.

```
SELECT count(*) FROM city (4079)
```

```
SELECT count(*) FROM country (239)
```

```
SELECT count(*) FROM countrylanguage (984)
```

Als we de city tabel en country tabel bekijken dan zien we het volgende:

TABEL City		
Kolomnaam	Beschrijving	Voorbeeld
ID	Primary Key (nummer)	19
Name	Naam van de stad	Amsterdam
CountryCode	Foreign Key naar Country	NLD
District	Provincie of Regio	Noord-Holland
Population	Bevolkingsgrootte	731200

TABEL Country	
---------------	--

Kolomnaam	Beschrijving	Voorbeeld
Code	Primary Key (Country Code)	NLD
Name	Naam van het land	Netherlands
Continent	Continent	Europe
Region	Regio	Western Europe
SurfaceArea	Oppervlakte (in KM ²)	41526.00
IndepYear	Jaar van onafhankelijkheid	1581
Population	Bevolkingsgrootte, inwoneraantal	15864000
LifeExpectancy	Levensverwachting	78.3
GNP	Bruto Nationaal Product	371362.00
GNPOld	Oud Bruto Nationaal Product	360478.00
LocalName	Lokale naam	Nederland
GovernmentForm	Regeringsvorm	Constitutional Monarchy
HeadOfState	'Baas' van het land	Alexander
Capital	Foreign Key naar City	5
Code2	Officiële landsafkorting (bijv NL)	2

Opgaven

Opgave 1

Kijk of je Nederland kunt vinden en druk alle kolommen af uit de tabel Country. Zoek daarvoor naar alle rijen waarvan naam begint met 'Net' (van Netherlands).

Zet de query die je gebruikt hebt op je antwoordenblad.

Opgave 2

Zoek nu alle landen die meer dan 100000000 (100 miljoen) inwoners hebben. Dat is een 1 met 9 nullen. Hoeveel landen hebben meer dan 100 miljoen inwoners?

Zet de query die je hebt gebruikt op je antwoordenblad.

Opgave 3

Hoeveel inwoners heeft het land Belgium?

Zet de query die je hebt gebruikt op je antwoordenblad.

Opgave 4

Hoeveel inwoners heeft het land Australia (Australë)? Het inwoneraantal staat in de kolom *Population* in de tabel *country*.

Zet de query die je gebruikt hebt op je antwoordenblad.

Opgave 5

We gaan de twee tabellen koppelen met de join. En we drukken alle plaatsen (city) af die in Nederland (Netherland) liggen en die in de database staan. Laten we eerste de join gaan maken. Maak de onderstaande query af en koppel de twee tabellen met de juiste keys aan elkaar.

```
SELECT *  
FROM city  
JOIN country  
on ... = ...  
WHERE country.Name='Netherlands'
```

Als het goed is zie je nu alle steden in Nederland die in de database staan. Verander de query nu zo dat je niet alle steden ziet maar dat je gewoon een aantal krijgt.

Opgave 6

Hoeveel *steden* uit Belgium (België) staan er in de database?

Zet de query die je gebruikt hebt op je antwoordenblad.

Opgave 7

Hoeveel *steden* uit France (Frankrijk) staan er in de database?

Zet de query die je gebruikt hebt op je antwoordenblad.

Opgave 8

Hoeveel *steden* uit Morocco (Marocco) staan er in de database?

Zet de query die je gebruikt hebt op je antwoordenblad.

Opgave 9

Hoeveel *landen* van het Continent Europe (Europa), staan er in de database?

Zet de query die je gebruikt hebt op je antwoordenblad.

Opgave 10

Hoeveel *steden* die onderdeel uitmaken van het Continent Europe (Europa), staan er in de database?

Zet de query die je gebruikt hebt op je antwoordenblad.

Opgave 11

Hoeveel *steden* die onderdeel uitmaken van het Continent Africa (Afrika), staan er in de database?

Zet de query die je gebruikt hebt op je antwoordenblad.

Opgave 12

We gaan nu twee tabellen joinen (samenvoegen). We willen namelijk opzoeken wat de hoofdstad is van Nigeria.

Dat doen we door uit de tabel city de naam van de city te halen. Dat zie je in de eerste twee regels van de query. Dan gaan we de tabel samenvoegen join-en met de country tabel. En als laatste stap selecteren we alleen het juiste land in het gedeelte achter de WHERE.

De query ziet er dus zo uit:

```
SELECT city.Name  
FROM city  
join country  
on ... = ...  
WHERE country.Name = 'Nigeria'
```

Maak de query af door op de plaats van de ... de juiste primary key en foreign key te zetten.

Zet de query die je hebt gebruikt op je antwoordenblad.

Opgave 13

In de World database staat nog een derde tabel en die heet *countrylanguage*. In deze tabel staat de CountryCode. Dit is een foreign key naar de Code kolom in de tabel country.

Maak nu een de voglende query af door op de plaats van de puntjes de juiste tabelnaam of kolomnamen in te vullen:

```
SELECT country.Continent, country.Name, countrylanguage.Language, countrylanguage.Percentage
FROM countrylanguage
JOIN .....
ON ..... = .....
WHERE countrylanguage.Language='Dutch'
ORDER BY countrylanguage.Percentage desc
```

Kijk goed naar de uitkomst van de query. Als het goed is kun je zien hoeveel procent van de inwoners van België Nederlands (Dutch) praten. Zorg dat alles regels van de query begrijpt voordat je verder gaat naar de voglende opgave.

Zet de werkende query op je antwoordenblad.

Opgave 14

Voor de volgende vraag kun je de query van opgave 12 als basis gebruiken.

Bepaal welke talen er in België worden gesproken en hoeveel procent van de inwoners elke taal spreekt.

```
SELECT country.Continent, country.Name, countrylanguage.Language, countrylanguage.Percentage,
countrylanguage.IsOfficial
FROM countrylanguage
FROM .....
JOIN .....
ON ..... = .....
WHERE ..... = 'Belgium'
ORDER BY countrylanguage.Percentage desc
```

In de laatste kolom 'IsOfficial' kun je zien of het een officiële taal van het land is. (T=true en F=false).

Zet de werkende query op je antwoordenblad.

Opgave 15

In de vorige query kon je in de laatste kolom 'IsOfficial' zien of het een officiële taal van het land is. (T=true en F=false). Verander nu de query van Opgave 13 zodat je alleen de officiële talen te zien krijgt.

Zet de werkende query op je antwoordenblad.

Opgave 16

Hoeveel officiële talen heeft 'South Africa' ?

Zet de query waarmee je dit antwoord hebt bepaald op je antwoordenblad.

Opgave 17

Hoeveel niet-officiële talen heeft 'India'?

Zet de query waarmee je dit antwoord hebt bepaald op je antwoordenblad.

Opgave 18

Hoeveel talen (officieel en niet-officieel) worden er op het continent 'Europe' gesproken?

Zet de query waarmee je dit antwoord hebt bepaald op je antwoordenblad.

Opgave 19

Hoeveel officiële talen worden er op het continent 'Europe' gesproken?

Zet de query waarmee je dit antwoord hebt bepaald op je antwoordenblad.

--

De Wereld-Database - Distinct

Deze les leren we wat distinct betekent en hoe je het kunt gebruiken.

In de afgelopen les hebben we ook gekeken naar de tabel `countrylanguage`.

TABEL <code>countrylanguage</code>		
Kolomnaam	Beschrijving	Voorbeeld
CountryCode	Code	NLD
IsOfficial	t of f	t = true en f = false (dus officiële taal dan true)
Language	taal	Dutch
Percentage	Percentage dat de taal spreekt	9.5

Stel je nu eens voor dat wel alle talen die er zijn zouden willen afdrukken, dan kunnen we met de volgende query alle talen afdrukken:

```
SELECT Language
FROM countrylanguage
```

Als je dit doet dan zie je dat er 984 regels (rows) worden getoond. Zijn er dan echt 984 talen?

Sorteer je resultaat van de vorige query en kijk nog eens goed naar het resultaat. Wat zie je?

Je kunt een query ontdebellen met het statement *distinct*.

Voer de volgende query uit:

```
SELECT distinct Language
FROM countrylanguage
order by Language asc
```

Wat zie je nu, hoeveel regels worden er getoond? Zie je nu wat er gebeurt door het toevoegen van *distinct* ?

Opdracht 1

Bekijk de volgende twee queries:

```
SELECT count(Language)
FROM countrylanguage
order by Language asc
```

en

```
SELECT count(distinct Language)
FROM countrylanguage
order by Language asc
```

Wat is komt er uit beide queries? Schrijf het antwoord op je antwoordenblad. Beide queries hebben een ander uitkomst. Leg in je eigen woorden uit wat het verschil is en hoe het wordt veroorzaakt.

Opdracht 2

Hoeveel continenten staan er in de database? Schrijf de query die je hebt gebruikt op je antwoordenblad.

Opdracht 3

Hoeveel GovernmentForm staan er in de World Database? Schrijf de query die je hebt gebruikt op je antwoordenblad.

Opdracht 4

Hoeveel talen (officieel en niet-officieel) worden er op het continent 'Europe' gesproken? Deze vraag heb je in de vorige les ook al beantwoord. Kijk nog eens naar deze query? Zou je hier een *distinct* kunnen gebruiken? Wat is nu je antwoord en waarom verschilt dat van je eerdere antwoord?

Zet de (nieuwe) query waarmee je dit antwoord hebt bepaald op je antwoordenblad. Noteer ook de uitkomst.

Opdracht 5

Bekijk de volgende twee queries waarin alle hoofdsteden uit de tabel *country* worden geteld.

```
SELECT count(distinct Capital) from country
```

en

```
SELECT count(Capital) from country
```

De uitkomst van beide queries is gelijk. Beschrijf waarom dat zo is op je antwoordenblad.

Opdracht 6

Stel je hebt een tabel `leerling` waarin de cijfers staan van alle leerlingen voor het vak SQL. Stel de tabel heeft 200 leerlingen.

Met de query

```
select cijfer from leerling
```

zie je alle cijfers die er zijn gehaald. Deze query geeft 24 regels als resultaat.

Stel je maakt de volgende query

```
select distinct cijfer from leerling
```

Enigszins verbaasd zie je dat het resultaat van deze query 10 is. Je ziet dus één regel en de waarde is 10.

Wat betekent dit? Schrijf je antwoord op het antwoordenblad.

Opgave 7

Hoeveel waarden kan de kolom `IsOfficial` uit de tabel `countrylanguage` hebben? Welke waarden zijn dat?

Schrijf je query en het resultaat op je antwoordenblad.

Opgave 8

Voer de volgende query uit:

```
SELECT distinct substring(Name,1,1)
from Country
order by Name asc
```

Let op, `substring(Name,1,1)` geeft de eerste letter van `Name` terug. Kijk goed naar het antwoord, mis je wat? Leg uit wat het antwoord betekent; welke conclusie kun je hieruit trekken?

Schrijf het antwoord op je antwoordenblad.

--