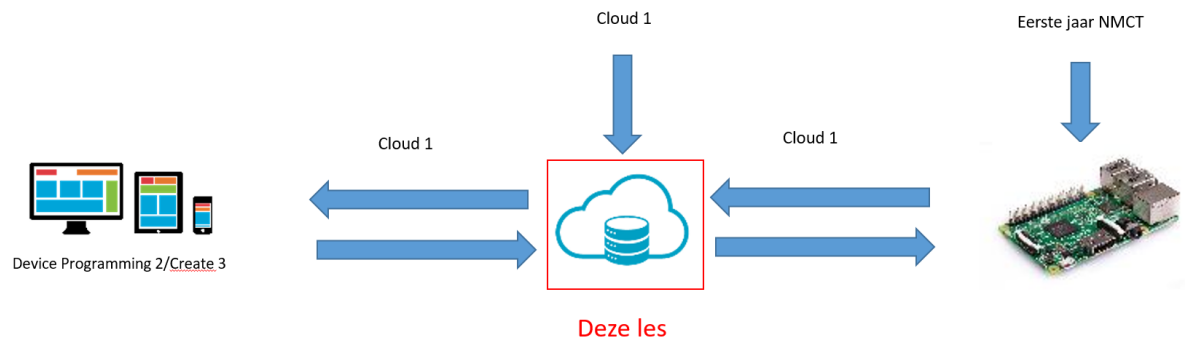


Les 1: Azure Cloud, VM, WebApp & SQL Database

Doelstellingen

- Kennis maken met Azure Cloud
- Oefening op SQL Database
- Zelfstandige VM opzetten
- Zelfstandig Azure Web App Opzetten
- Zelfstandig Alert opzetten voor Web App
- Relationale database opzetten en gebruiken binnen de Azure Cloud omgeving



Activatie

<https://azure.microsoft.com/en-us/education/>

Inloggen

Surf <https://portal.azure.com>

Daarna moet u inloggen met uw Azure gegevens.

Een eerste service

Het dashboard

Na het inloggen verschijnt de Azure Portal. Van hieruit kunnen we verschillende soorten services aanmaken. Links staan een lijst van de meest gebruikte services zodat we deze snel kunnen aanmaken. Klikken we op het + teken dan krijgen we toegang tot meer services. Het dashboard is overzicht die u zelf kan samenstellen met de voor u belangrijkste services. U kan meerdere dashboards aanmaken per account.

Microsoft Azure

Search resources, services, and docs (5/1/7)

Home > Resource groups

Resource groups

regio:west-Vlaanderen

+ Add | Edit columns | Refresh | Export to CSV | Assign tags

Subscriptions: 1 of 4 selected - Don't see a subscription? Open Directory + Subscription settings

Filter by name: OHK | All locations | All tags

NAME	SUBSCRIPTION	LOCATION
igamWinis	OHK	West Europe
cloud-shell-storage-eastus	OHK	East US
DAEChatBot	OHK	Central US
DAESTAFF	OHK	West Europe
DefaultResourceGroup-WEU	OHK	West Europe
DockerDieterDP	OHK	West Europe
Henk_Project	OHK	North Europe
Hoywaert_Eil_Cloud	OHK	West Europe
IoTCloud	OHK	West Europe
IPO	OHK	West Europe
LiesPinket	OHK	North Europe
NetworkWatcherRG	OHK	West Europe
Sajin_Walcarlus	OHK	West Europe
Vannieuwenhuysslohan	OHK	West Europe

Aan de rechter kan van de portal hebben we volgende opties: een zoekveld om te zoeken binnen onze aangemaakte resources. Enkele andere interessante opties kan u hieronder vinden



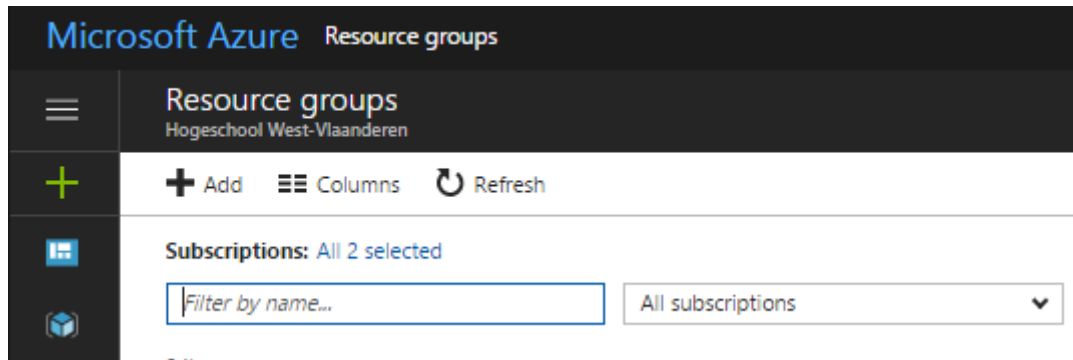
Notificaties, welke services zijn aan het uitrollen, foutmeldingen

Instellingen zoals taal, wij raden aan om in het Engels te werken

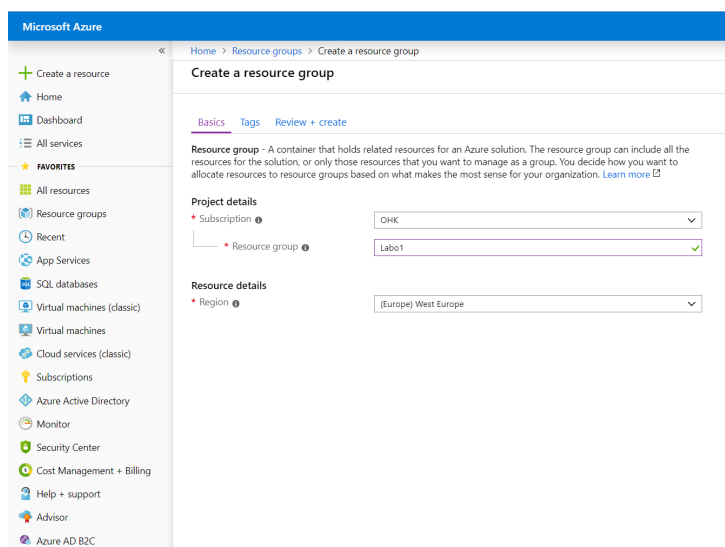
Bash/Powershell om via de commandline Azure services aan te maken of te beheren

De Resource Group

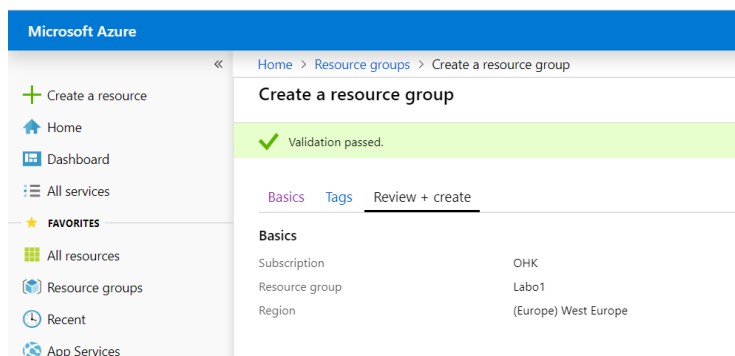
Waar een Azure subscription het hoogste niveau is binnen Azure, is de resource group een logische eenheid van organisatie binnen Azure. Binnen een resource maken we de verschillende services aan. U kan zelf kiezen hoe u de onderverdeling maakt. U kan bijvoorbeeld een resource maken per klant of project. Andere mogelijkheden zijn per type service vb.: een storage resource group, een webservice resource group. Nog een andere benadering is een test resource group en een productie resource group. Er zijn dus heel wat mogelijkheden om dit te configureren. U mag dit vrij kiezen. Het aanmaken van de resource groep doet u door te klikken op de blauwe kubus en daarna **Add** te kiezen.



Daarna krijgt u het volgende scherm waarin u de naam kiest, de subscription en de locatie van de resource group. Wij kiezen als locatie ergens in Europa. Indien u een project hebt in de VS of Azië dan kiest u best een locatie ergens in die regio's.

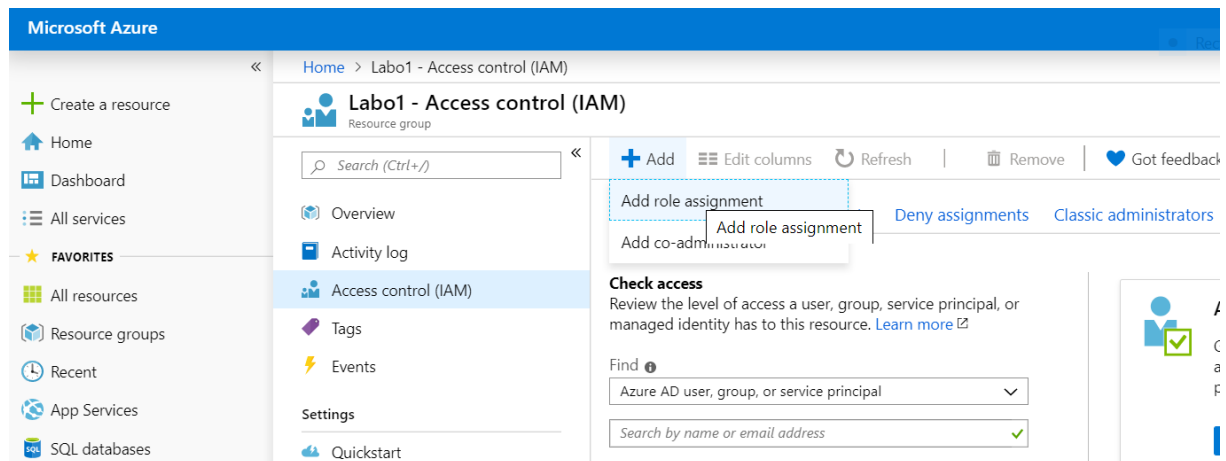


Daarna kiezen we **Review + Create** onderaan zodat het platform alles nog eens kan controleren.

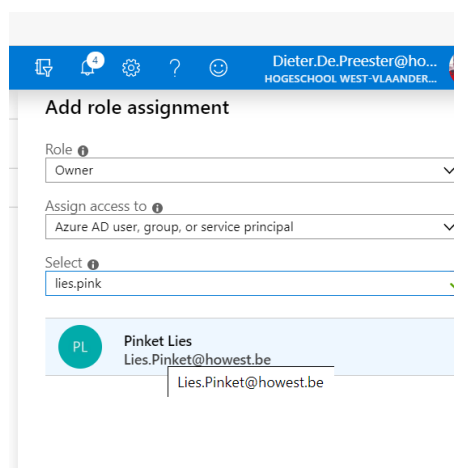


Daarna kiezen we **Create** en zal de resource group aangemaakt worden.

Als u derden toegang wil geven aan een resource group dan kan u dit doen door de resource group te openen en te kiezen voor **Access Control (IAM)**.

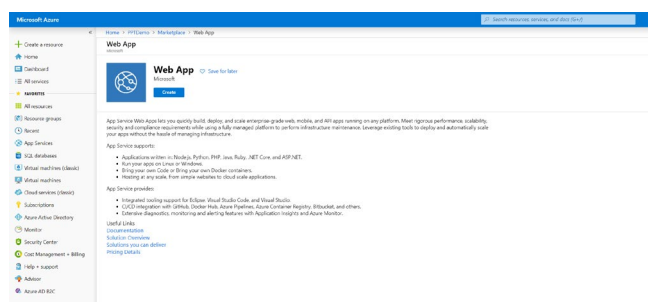


Via de knop **Add** ➔ **Role Assignment** kan u een andere gebruiker toegang geven tot de resource group. U moet een rol kiezen. Via de rol zal de gebruiker al dan niet bepaalde rechten hebben. U moet dit steeds goed nakijken in de documentatie wat de mogelijkheden zijn van de gekozen **role**: <https://docs.microsoft.com/en-us/azure/active-directory/role-based-access-built-in-roles>. U moet ook het e-mailadres opgeven. Als alles goed verloopt zal de gebruiker een e-mail ontvangen met de procedure voor activatie. Daarna kan deze gebruiker via <http://portal.azure.com> en zijn e-mail en paswoord inloggen op Azure en naargelang de role zaken uitvoeren in de resource group.

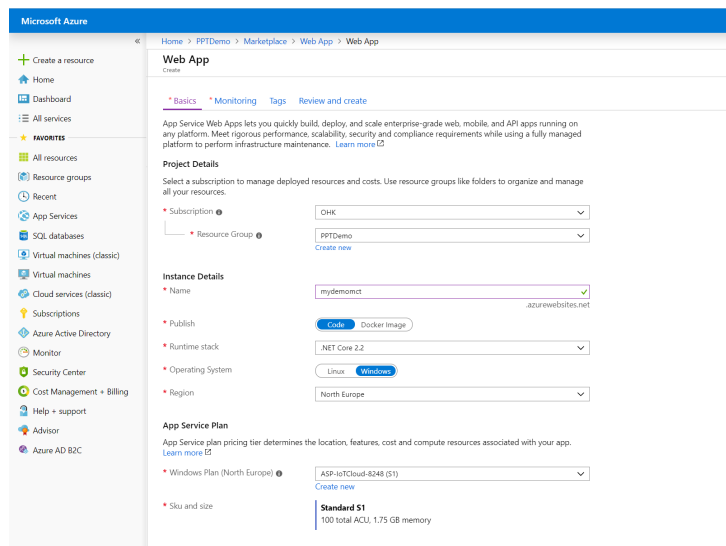


Een eerste service: Azure Web App

Na de aanmaak van de resource group zijn we klaar voor het aanmaken van een eerste service. Wij kiezen voor een Azure Web App. Een Azure Web App is een basis website die we hosten in de cloud. Druk links op het + symbool. Daarna zoek je op **“Web App”**



Bij de volgende stap vullen we de naam van een resource group in waarin we de app willen aanmaken. We kiezen ook een unieke naam. Deze zal eindigen op**azurewebsites.net**. Dit domein krijgen we er gratis bij. Bij **Publish** kiezen we **Code**. Bij **Runtime Stack** **.NET Core 2.2** (bekijk ook eens andere opties). **Het operating system** is **Windows**. De **region** is het datacenter waar we de applicatie plaatsen. Dit kiest je zo dicht mogelijk bij de klant. Wij kiezen voor ergens in Europa. De laatste stap **Sku and Size** is ook zeer belangrijk. Het is op basis van deze optie dat we moeten betalen. Kies hier voor **Change Size**.



Microsoft Azure

Home > PPTDemo > Marketplace > Web App > Web App

Web App

Overview

App Service Web Apps lets you quickly build, deploy, and scale enterprise-grade web, mobile, and API apps running on any platform. Meet rigorous performance, scalability, security and compliance requirements while using a fully managed platform to perform infrastructure maintenance. [Learn more](#)

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

- Subscription: CHK
- Resource Group: PPTDemo [Create new](#)

Instance Details

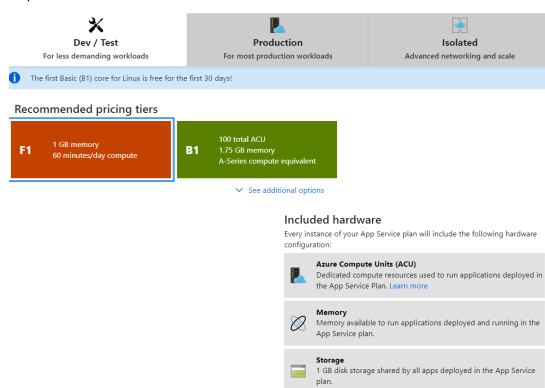
- Name: mydemo123 [azurewebsites.net](#)
- Publish: Code [Docker image](#)
- Runtime stack: .NET Core 2.2
- Operating System: Linux [Windows](#)
- Region: North Europe

App Service Plan

App Service plan pricing tier determines the location, features, cost and compute resources associated with your app. [Learn more](#)

- Windows Plan (North Europe): ASP-1a2Cloud-8248 (S1) [Create new](#)
- Sku and size: **Standard S1**
100 total ACU, 1.75 GB memory

Na het kiezen van **Change Size** krijgen we onderstaande scherm. Deze laatste maar zeer belangrijkste stap is het prijsniveau kiezen. Hoe hoger de prijs, hoe krachtiger de server, hoe sneller de CPU, hoe meer geheugen, etc.... Je kan ook kiezen voor de free tier. Dit wil zeggen dat je op een gedeelde server zit met andere klanten en beperkte datatrafiek per dag hebt. Een Shared is hetzelfde als free maar zonder datatrafiek beperking en de mogelijkheid om een domein (www.mijndomein.be) aan te koppelen. Wij kiezen voor F1 **Free**. Indien u dit wenst kan u dit achteraf aanpassen zonder problemen. Dat is één van de voordelen van de Cloud.



Dev / Test
For less demanding workloads

Production
For most production workloads

Isolated
Advanced networking and scale

The first Basic (B1) core for Linux is free for the first 30 days!

Recommended pricing tiers

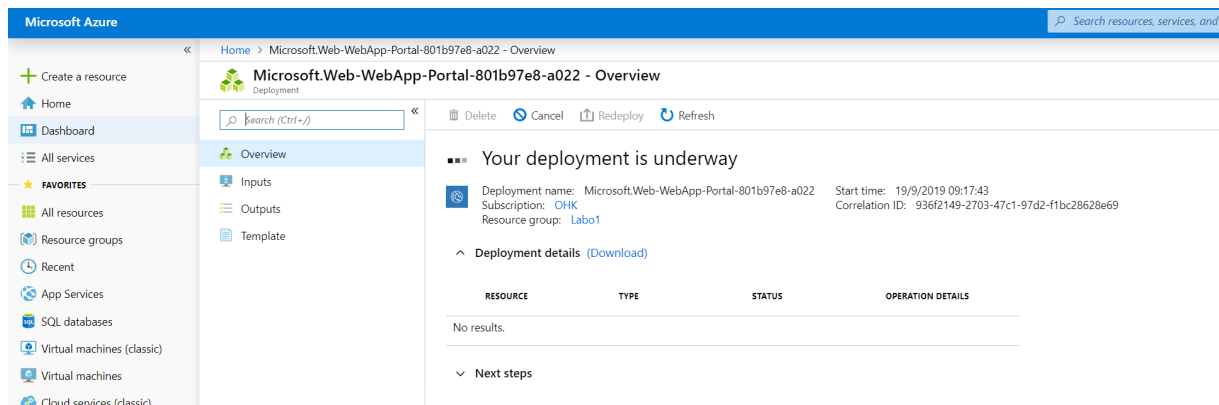
Tier	Configuration
F1	1 GB memory 60 minutes/day compute
B1	100 total ACU 1.75 GB memory A-Series compute equivalent

[See additional options](#)

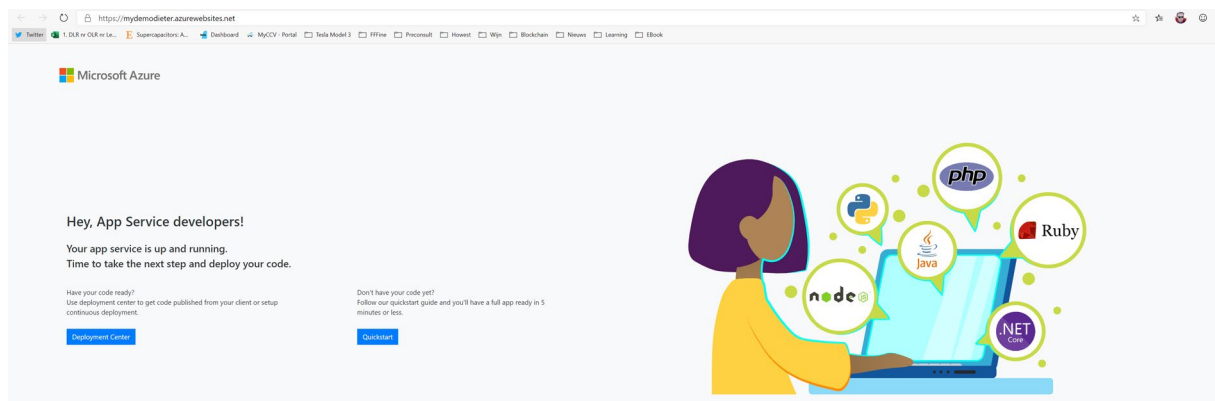
Included hardware

Every instance of your App Service plan will include the following hardware configuration:

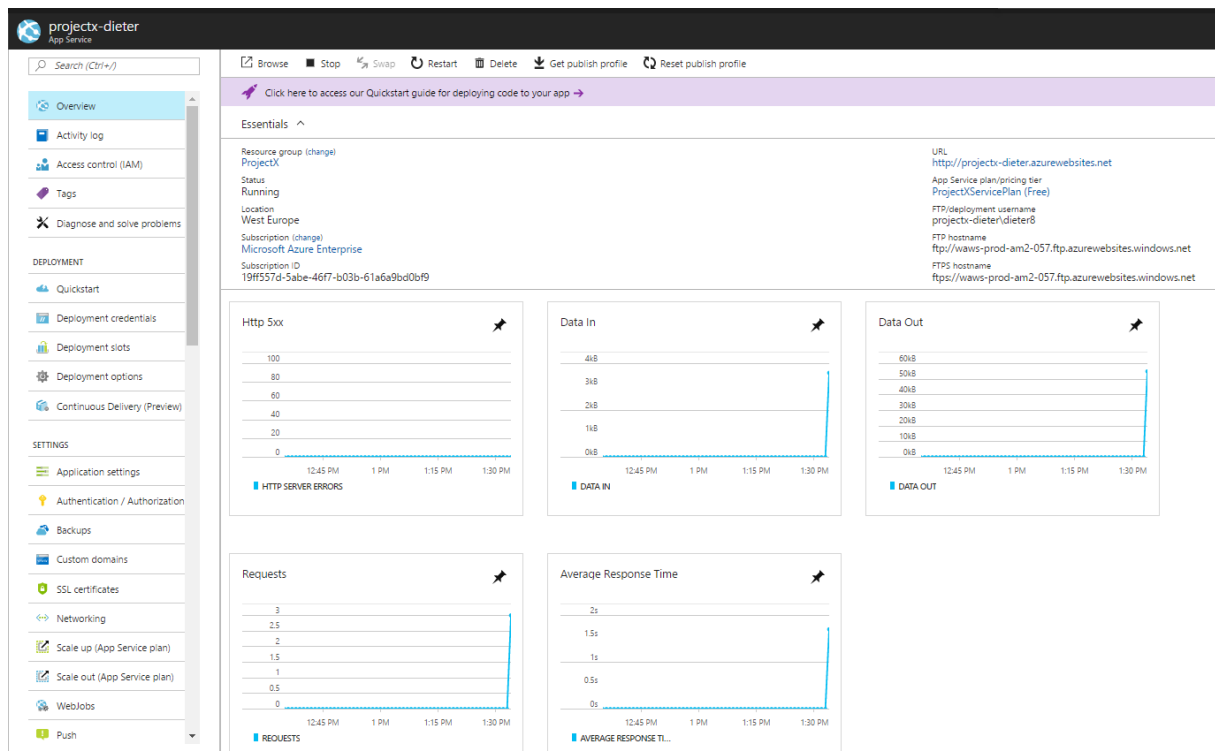
- Azure Compute Units (ACU)**
Dedicated compute resources used to run applications deployed in the App Service Plan. [Learn more](#)
- Memory**
Memory available to run applications deployed and running in the App Service plan.
- Storage**
1 GB disk storage shared by all apps deployed in the App Service plan.



Daarna kan je de web app laten aanmaken en een kleine minuut later zou deze actief moeten zijn op de door u gekozen URL.



Het beheer, de logging en de monitoring van de applicatie kunnen we volgen via het overzicht scherm van de web app. Ga naar de resource group en kies de web app. Het startscherm bevat een reeks interessante grafieken zoals http requests, data verbruik etc. Via het linker menu kan u opties wijzigen. We gaan deze nu niet allemaal in detail overlopen. U zal deze ten gepaste tijden leren kennen.



Mijn website online plaatsen

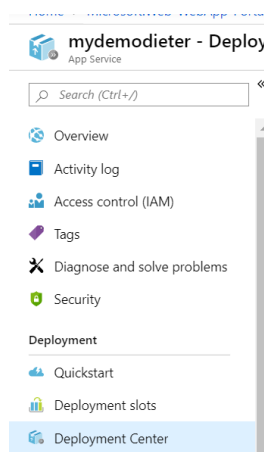
Nu we een web app hebben aangemaakt is het de bedoeling dat we onze eigen websites online kunnen plaatsen. Bij Azure Web App kan u verschillende soorten technologie gebruiken, waaronder zeer veel opensource en niet Microsoft technologieën.

- ASP.NET
- NodeJS
- PHP
- Python Flask
- Etc.

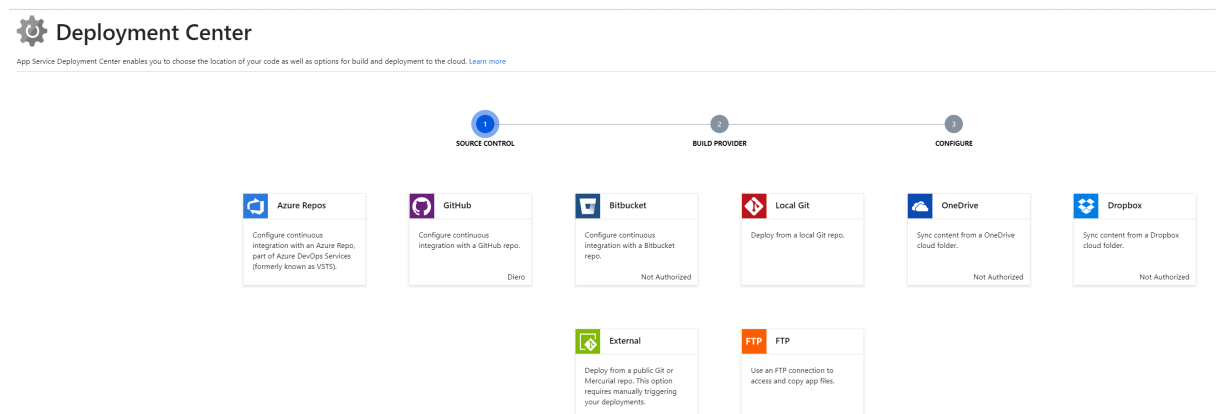
Voor dit labo gaan we een eenvoudige HTML-website online plaatsen. U kan zelf een eenvoudige HTML-pagina maken die u straks online kan plaatsen of u mag ook een bootstrap template downloaden op <https://startbootstrap.com/template-categories/all/>.

Een website online plaatsen op Azure kan op verschillende manieren. Wij kiezen voor de meeste professionele manier en dit is via Git. We zullen dus eerste onze source code op een git service moeten plaatsen. Ik kies voor GitHub (<http://www.github.com>). Maak een nieuwe repository aan in GitHub en probeer nu zelf de code online te plaatsen in deze repository. U zou Git moeten kennen uit de project module van 1 NMCT. **Indien u dit niet meer weet doen we dit samen in het labo en moet u dit noteren.**

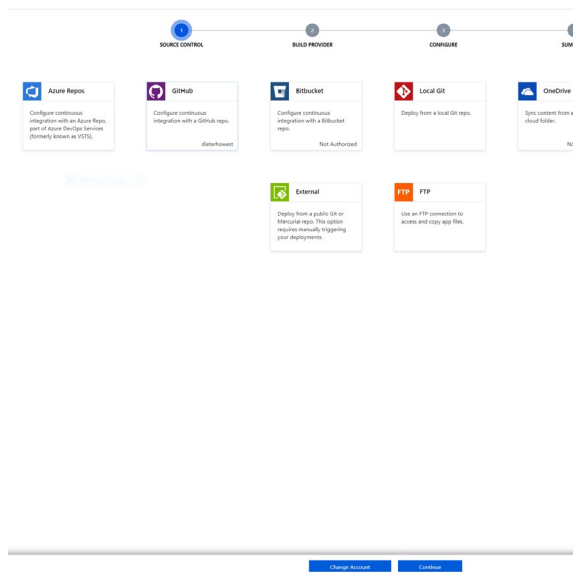
Als de code online staat dan moeten we de GitHub repository koppelen aan de Azure Web App. Open de Azure Web App en kies **Deployment Center**.



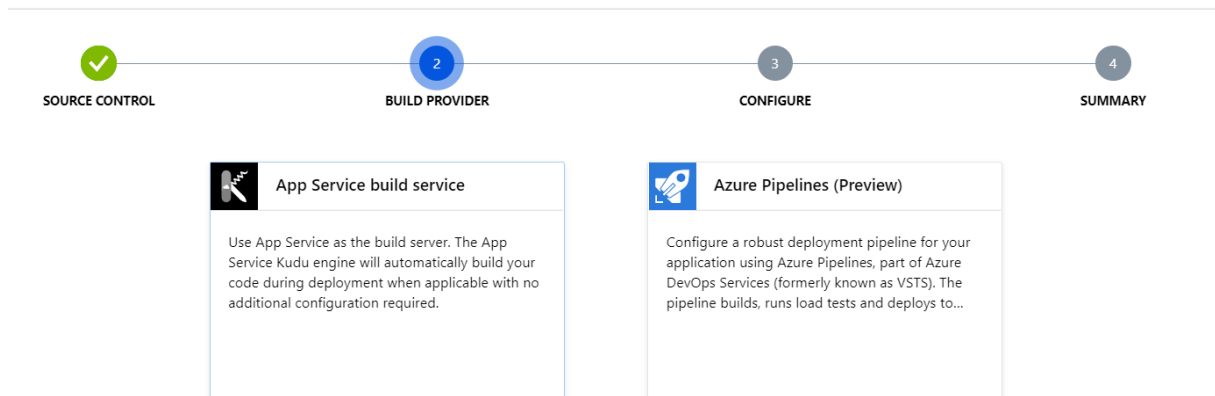
In het volgende scherm kan je kiezen van waar de source code van de applicatie komt. Wij kiezen voor GitHub maar er zijn ook andere interessante opties zoals OneDrive of DropBox.



Bij de volgende stap moet u inloggen met uw GitHub account zodat de koppeling kan plaatsvinden. Kies om in te loggen met uw GitHub Account. Daarna Kiest u onderaan **Continue**.



Nu kiezen we voor **App Service Build service**:



Daarna moet u eventueel de organisatie kiezen gevolgd door het project die je wenst te koppelen. De laatste stap is de **Branch**. Normaal is dit de master.

Code

Organization:

Repository:

Branch:

i If you can't find an organization or repository, you might need to enable additional permissions on GitHub. [?](#)

Daarna drukken we op **Finish** en zou de koppeling moeten plaats vinden. Daarnaast zal ook een eerste **deploy** plaatsvinden. Controleer nu op de publieke URL van de web app of je HTML-pagina's online staan.

Opdrachten Azure Web App:

1. Probeer nu ook eens iets te wijzigen in de HTML-pagina's op je pc. Daarna doe je een git **commit** en een **push**. Normaal gezien zouden de wijzigingen vrij snel online moeten staan.
2. Onze webapplicatie krijgt plots zeer veel bezoekers, hoe kan je zonder de web app opnieuw aan te maken deze laten draaien op een zwaardere server? Documenteer uw oplossing met screenshots zodat u dit later kan terugvinden.
3. Hoe kan je zonder de web app opnieuw aan te maken deze laten schalen. Met schalen bedoelen we de applicatie op 2 of 3 servers laten draaien zodat we meer trafiek kunnen verwerken? Welke instelling moeten we hiervoor wijzigen. Documenteer uw oplossing met screenshots zodat u dit later kan terugvinden.



Cloud Databases

In dit stuk gaan we dieper in op een ander belangrijk onderdeel binnen de Azure, de relationele databases. Binnen Azure hebben we twee relationele databases die we vlot kunnen aanmaken en gebruiken:

- SQL Database
- My SQL

Beide databases werken volgens het **managed services** concept. Dit wil zeggen dat we zeer eenvoudige de database kunnen aanmaken en terug verwijderen. We moeten zelf geen servers aanmaken, software installeren, **patches** en **backups** maken. Al deze zaken zal Azure voor ons doen. We noemen dit **managed services**. We betalen enkel voor wat we gebruiken en hoelang we de service gebruiken.

SQL Database

Aanmaken SQL Database + Server

SQL Azure is een **managed database** op basis van SQL. We kunnen deze vrij eenvoudig aanmaken en terug verwijderen en betalen voor de duur van het gebruik en de hoeveelheid opslagruimte voor de database. Ga naar de resource groep en kies **Add**. De naam van de service is **Azure SQL**. Na het kiezen van Azure SQL moeten we het soort service kiezen. Wij gaan altijd met SQL Databases werken. Voor we kunnen starten moeten we eerst een Database Server aanmaken. Deze moeten we aanmaken maar NIET beheren. Azure zal dit voor ons doen.

How do you plan to use the service?

SQL databases	SQL managed instances	SQL virtual machines
Best for modern cloud applications. Hyperscale and serverless options are available.	Best for most migrations to the cloud. Lift-and-shift ready.	Best for migrations and applications requiring OS-level access. Lift-and-shift ready.
Resource type Database server	Resource type Single instance	Image Please select an offer
Create Show details	Create Show details	Create Show details

Na de keuze krijgen we volgende scherm te zien waar we volgende eenvoudige zaken kunnen invullen:

Create SQL Database Server

Microsoft

[Basics](#) [Networking](#) [Additional settings](#) [Tags](#) [Review + create](#)

SQL database server is a logical container for managing databases and elastic pools. Complete the Basic tab, then go to Review + Create to provision with smart defaults, or visit each tab to customize. [Learn more](#)

Project details


Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

* Subscription 

* Resource group 
[Create new](#)


Server details


Enter required settings for this server, including providing a name and location.


* Server name  .database.windows.net

* Location

Administrator account

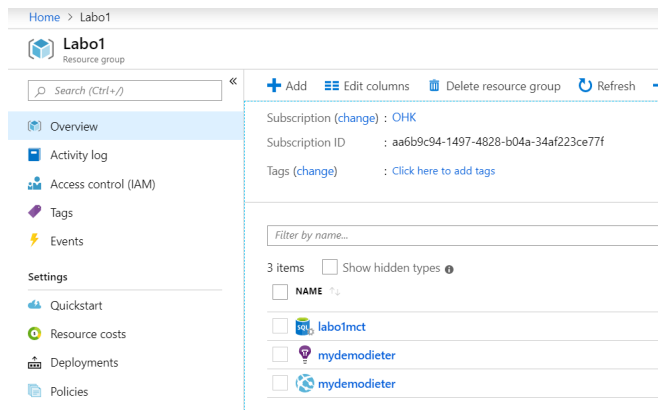
* Server admin login 

* Password 

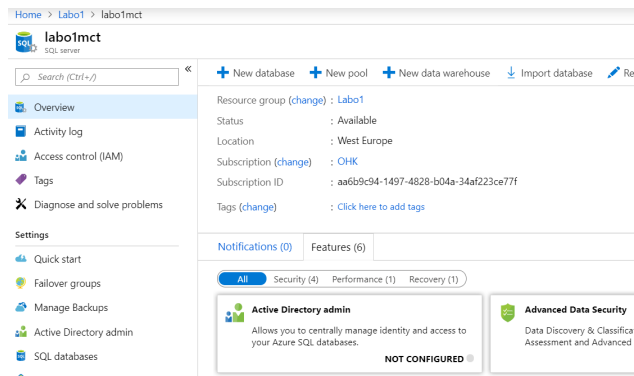
* Confirm password 

- Server naam
- De subscription
- De resource group
- Locatie
- De admin en zijn paswoord

Daarna zal de server worden aangemaakt. Achteraf kunnen we deze vinden in onze Resource Group



We kiezen de SQL server en krijgen onderstaande scherm waar we kiezen voor **New Database**:



SQL database □ ×

* Database name

* Select source ⓘ
 Blank database ▼

Server
 labo1mct (westeurope) 🔒

Want to use SQL elastic pool? ⓘ
☐ Yes ☒ Not now

* Pricing tier ⓘ
 General Purpose: Gen5, 2 vCores, ... >

* Collation ⓘ
 ✓

We kiezen als database een **Blank Database**. We zullen later zelf één toevoegen. Een belangrijke ander optie is de keuze van de server. Daarnaast moeten we ook nog de optie **SQL Elastic pool** op **Not now** plaatsen. De laatste optie is de **pricing tier**. Dit is een belangrijke keuze want deze zal enerzijds bepalen hoeveel de database ons kost en anderzijds wat de opslagcapaciteit is maar ook de snelheid. Wij kiezen voor een basic met **5 DTU** en **100MB Storage**. DTU staat voor **Database Transaction Unit** en is een mix van CPU-gebruik, geheugen en I/O doorvoer. Onze database kan 5 DTU gebruiken. Als we meer resources gebruiken dan 5 DTU dan zal er **throttling** plaatsvinden. Dit is het afbreken van verbindingen of **timeouts**. Dit zal blijven duren tot we minder **resources** gebruiken.

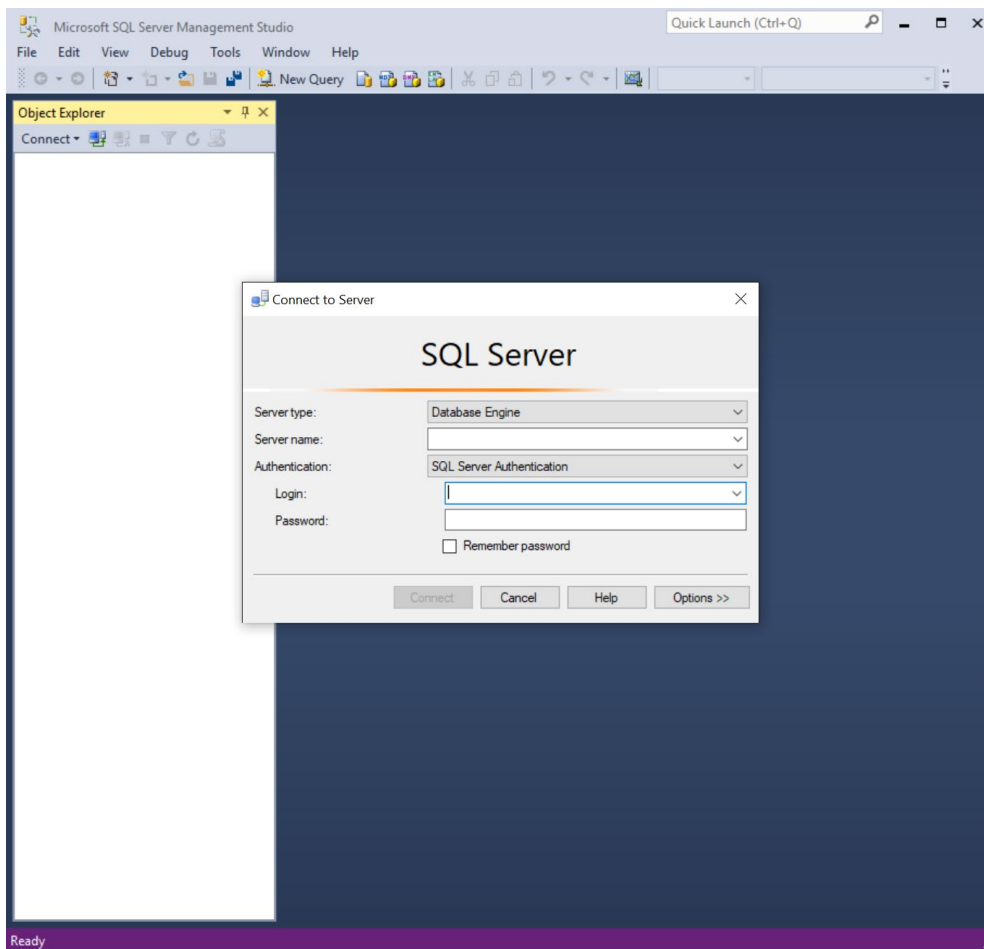
Na de aanmaak van de database kunnen de database en database server info terugvinden door te SQL Server te kiezen waarom de database staat.

Naam SQL Database

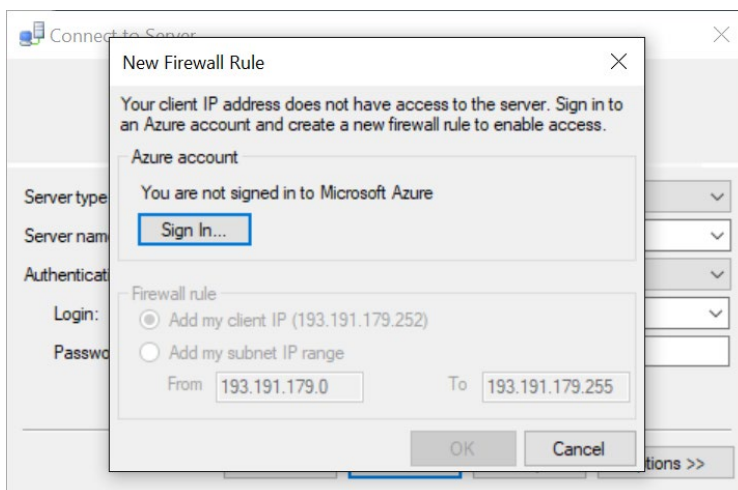
NAME	TYPE	LOCATION
labo1mct	SQL server	West Europe
demo01 (labo1mct/demo01)	SQL database	West Europe
mydemodieter	Application Insights	Central US
mydemodieter	App Service	Central US

Het connecteren met de SQL Database zal gebeuren via de **SQL Management Studio**. Deze kan u downloaden via <https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms>.

Na de installatie start u de SQL Management Studio en krijgt u het volgende scherm:



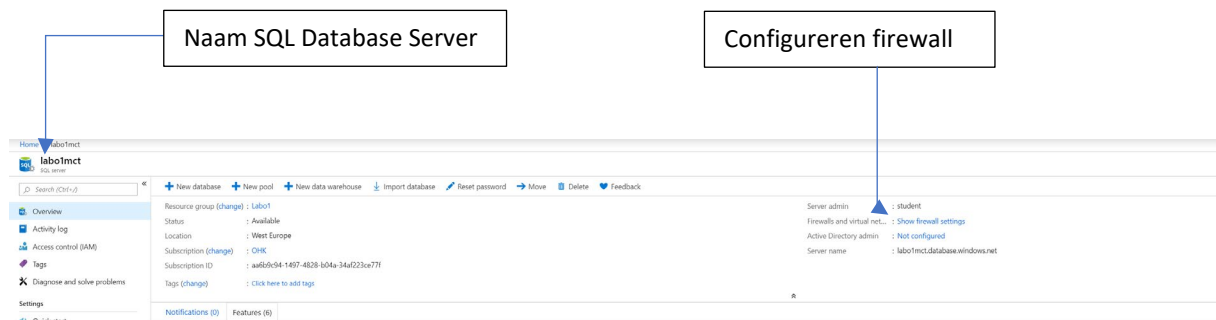
Bij Server name vullen we de naam van onze server (die kan terugvinden in de Portal) in gevolgd door database.windows.net vb. Projectx-dieter.database.windows.net. Daarna moeten we nog de **admin user** en **password** invullen. Dit hebben we moeten opgeven bij de aanmaak van de database. Daarna proberen we te connecteren en krijgen we het volgende scherm:



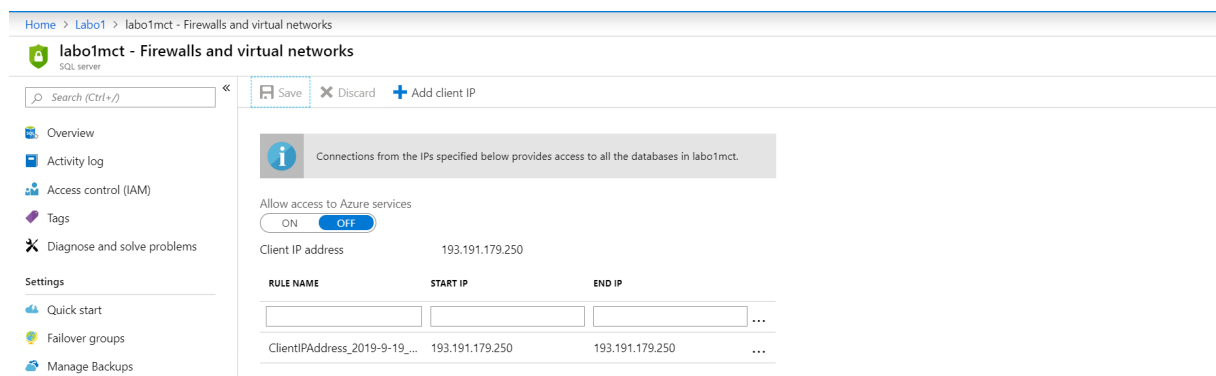
Bovenstaande scherm is een melding dat we niet kunnen connecteren omdat de **SQL Database Firewall** het **IP adres** niet kent van de **client** (wij) die wil connecteren. Dit is een veiligheidsmaatregel want anders kan iedereen zomaar op onze database connecteren. Via **Sign In** moeten we inloggen met onze **Azure Account**. Daarna zal het voorstel komen om één IP Adres of een range van adressen toegang te geven. Je kan kiezen voor 1 adres. Daarna zal de **management studio** verbinden met de server.

Opmerking: Indien u dit thuis zou proberen dan zal u ook deze melding krijgen en moet u het IP-adres van thuis toevoegen.

Een andere manier om de firewall in te stellen is via de Azure Portal. In het overzicht scherm van de SQL Server heb je een link **Show Firewall Settings**.

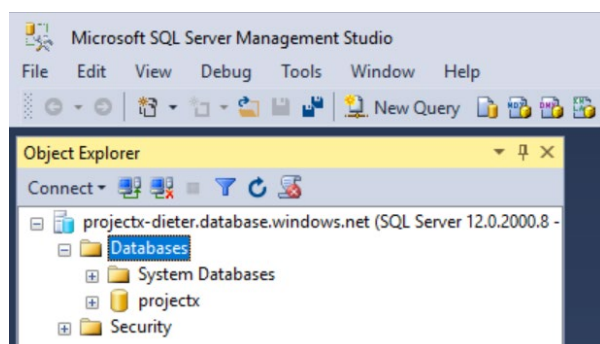


Daarna komt je op het onderstaande scherm terecht waar je de IP-adressen kan toevoegen aan de firewall. Kies hiervoor **Add Client IP**.



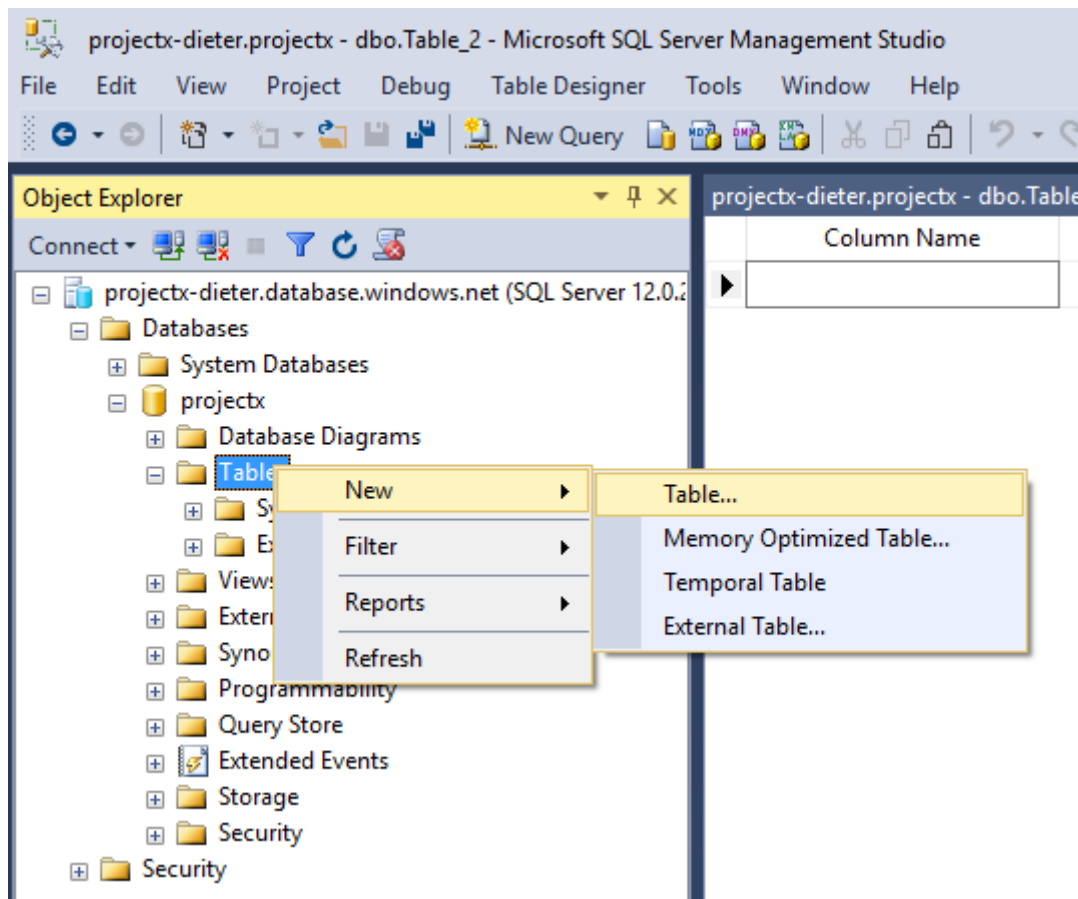
Beheren van de database

Via de SQL Management Studio kunnen we onze database in de Cloud beheren. Na het connecteren krijgen we een overzicht van de verschillende databases op de Cloud server.



Een nieuw tabel aanmaken.

Kies eerst de database waarop je wenst te werken. Daarna kies je voor **Tables** en via de rechtermuis kiezen we voor **New Table**.



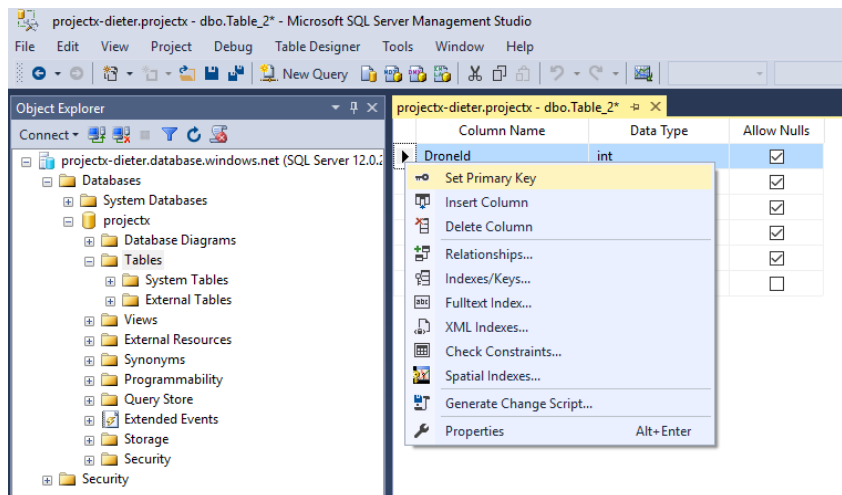
Rechts verschijnt de **table editor**. Waarin we de kolommen kunnen aanmaken. Volgende zaken moeten we opgeven:

- Naam van de kolom
- Datatype (welk soort data kan ik opslaan in deze kolom), een overzicht van de datatypes voor SQL Database kan je hier vinden <https://docs.microsoft.com/en-us/sql/t-sql/data-types/data-types-transact-sql>
- Kan de kolom niets bevatten (null waarde)

Voeg nu volgende kolommen toe met hun juiste datatype.

Column Name	Data Type	Allow Nulls
Droneld	int	<input type="checkbox"/>
DroneType	nvarchar(50)	<input checked="" type="checkbox"/>
Price	decimal(18, 2)	<input checked="" type="checkbox"/>
InStock	bit	<input checked="" type="checkbox"/>
ImageName	nvarchar(270)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

De **primary key** kan je instellen via rechtermuis en den **Set Primary Key**.



Als laatste stellen we de kolom **Droneld** in op **Identity**. Dit wil zeggen dat de database zelf een unieke waarde zal kiezen oplopend. Selecteer de kolom **Droneld** en kies onderaan voor **(Is Identity)** Yes.

projectx-dieter.projectx - dbo.Drones* SQLQuery3.sql - proj...objectx (dieter (90)) SQLQuery2.3

	Column Name	Data Type	Allow Nulls
🔑	Droneld	int	<input type="checkbox"/>
	DroneType	nvarchar(50)	<input checked="" type="checkbox"/>
	Price	decimal(18, 2)	<input checked="" type="checkbox"/>
▶	InStock	bit	<input checked="" type="checkbox"/>
	ImageName	nvarchar(270)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Column Properties

(General)	
(Name)	InStock
Allow Nulls	Yes
Data Type	bit
Default Value or Binding	
Table Designer	
Collation	< database default >
Computed Column Specification	
Condensed Data Type	bit
Description	
Deterministic	Yes
DTS-published	No
Full-text Specification	No
Has Non-SQL Server Subscriber	No
Identity Specification	No
(Is Identity)	No
Identity Increment	
Identity Seed	

Nu kunnen we de tabel opslaan en een naam geven.

Toevoegen van data

Het toevoegen van data doen we met behulp van de TSQL-taal. De basis hebt u reeds gezien in 1 NMCT. U zou reeds moeten weten wat een INSERT, UPDATE, DELETE zijn.

Probeer nu zelf volgende drones in te voegen via een INSERT Statement

Drone Type	Prijs	Stock	Image Name
DJI Phantom 4 Pro+ Drone	799	Ja	Drone1.jpg
Yuneec Typhoon H inkl. Wärmebildkamera Hexacopte	299	Neen	Drone2.jpg
Hubsan X4 FPV Drone RTF First Person View	599	Ja	Drone3.jpg

Opdrachten SQL Database

We hebben volgende probleemstelling:

Ontwikkel de database voor een klein toegangscontrolesysteem. Het systeem bestaat uit lezers. Iedere lezer heeft een naam en is van een bepaald merk. Een lezer is gekoppeld aan een controller. Een controller bevat een omschrijving en is aanwezig in één gebouw. De controller kan een Raspberry Pi 2 of 3 zijn. Maar later kan het misschien nog een ander soort hardware zijn. De controller bevat een IP-adres. Een controller staat in een gebouw. Een gebouw heeft een naam en een adres en een globaal telefoonnr. Er zitten ook gebruikers in het systeem. Deze gebruikers hebben een naam, voornaam en e-mail en zijn gekoppeld aan één of meerder badges. Een badge bestaat uit een nummer en kan al dan niet geblokkeerd zijn over het volledige systeem. Een badge heeft toegang tot 1 of meerdere lezers. Als iemand ergens een lezer gebruikt dan zal er een transactie weggeschreven worden. In de transactie loggen we de locatie, de lezer, de controller de gebruiker, heeft de gebruiker toegang of niet en een tijdstip.

- 1) Teken het ER schema op papier en stem af met de docent
- 2) Maak een Azure SQL Database aan en voeg de tabellen en relaties toe
 - a. Wat zal u gebruiken als datatype voor het autonummer veld?
 - b. Bekijk het datatype **uniqueidentifier** <https://docs.microsoft.com/en-us/sql/t-sql/data-types/uniqueidentifier-transact-sql>
 - c. Wijzig een paar tabellen zodat deze een **uniqueidentifier** gebruiken
- 3) Voeg in iedere tabel wat records toe zodat er in iedere tabel data aanwezig is.

Opdrachten Azure Virtual Machines

- 1) Maak een Resource Group **MyVM**. In deze group maken we een eigen Windows 10 Pro aan. Na het aanmaken probeer je in te loggen in deze machine via RDP (Remote Desktop). Maak screenshots en hou deze bij zodat u achteraf nog weet hoe u dit moet doen.
- 2) Naast de VM zal Azure nog een aantal andere zaken aanmaken in de resource group, bekijk deze kort, wat zijn deze zaken?
- 3) Hoe kan ik zien wat de kosten zijn/zullen zijn van mijn resource group?
- 4) Verwijder nu de **volledige Resource Group MyVM met één actie**

Opdrachten Azure Web App

- 1) Maak een resource group **MyBlog**
- 2) Zoek nu of het mogelijk is een Azure Web App te maken waar Wordpress geïnstalleerd is zodat we deze niet zelf meer moeten installeren
- 3) Zoek hoe je een **Alert** kan doen afgaan wanneer een gebruiker surft naar een pagina die NIET bestaat op onze Wordpress site. Je kan dit instellen via de portal. Documenteer terug zodat u dit later nog weet.
- 4) Hoe kan ik ervoor zorgen dat ik deze resource group niet per ongeluk kan verwijderen? Hoe kunnen we dit beveiligen?
- 5) Verwijder nu de **volledige Resource Group MyBlog met één actie**

Opdracht thuis

- 1) Maak een **Azure Database for MySQL** meer info kan u vinden op <https://docs.microsoft.com/en-us/azure/mysql/quickstart-create-mysql-server-database-using-azure-portal>
- 2) Zoek een tool op het Internet waarmee je kan connecteren naar een MySQL database : vb MySQL Workbench
- 3) Probeer connectie te maken via deze tool met de MySQL op Azure.
- 4) Maak nu de tabellen en relaties aan zoals uit de vorige opgave. Zorg er ook voor dat er in ieder tabel data zit.

