

Blok 7 - Intro Frameworks en Deployment

- [Prompt Engineering 3](#)
- [Deployment](#)
- [CMS](#)
- [Yii Introduction](#)

Prompt Engineering 3

1, *Introductie*

We hebben in prompt engineering 1 en 2 geleerd waaraan een goede prompt moest voldoen.

Dit zijn de basis kenmerken van een goede prompt. De eerste drie kenmerken moet je prompt **altijd** aan voldoen!

1. **Context** - een goede prompt heeft voldoende contextrt.
2. **Details/Specifiek** - een goed prompt heeft voldoende details en is zo specifiek mogelijk.
3. **Duidelijkheid** - een goede prompt is concreet en bevat geen vage termen zoals (mooi, groot,), zei [hier](#).
4. **Doelgericht** - een goede prompt is doelgericht.
5. **Vorm** - in een goede prompt kan je de output in een bepaalde vorm vragen.
6. **Toon** - door in de prompt de toon op te nemen, bepaal je de vorm van het antwoord.

In deze module gaan we nog meer **advanced prompt-technieken** leren. Deze technieken heb je niet altijd nodig maar het is handig om deze technieken te kennen.

Bovendien zijn de meeste technieken ook toepasbaar in als '**problem solving**' technieken.

1. **Gebruik een rol (role prompting)**
Laat de AI een specifieke rol of expertise aannemen (bijv. leraar, programmeur, leerling).
2. **Vraag om alternatieven of variaties**
Stimuleer de AI om meerdere ideeën of invalshoeken te geven.
3. **Iteratief verbeteren**
Vraag de AI om haar eigen antwoord te verbeteren of herschrijven.
4. **Gebruik prompt templates**
Werk met vaste structuren zoals: context, doel, outputvorm, toon.

5. **Vraag om zelfvragen (self-questioning)**

Laat de AI zelf bedenken welke vragen nodig zijn om het probleem goed op te lossen.

6. **Gebruik kritiek als leermoment (prompt debugging)**

Laat de AI uitleggen waarom een prompt niet goed werkte en hoe je die kunt verbeteren.

1. *Role Prompting*

□□ Uitleg

Je kunt de AI vragen om te reageren **alsof ze een bepaalde rol heeft**. Bijvoorbeeld:

- een ervaren programmeur
- een beginnende student
- een HTML-docent
- een code-reviewer

Door een rol te geven, pas je automatisch de **toon, moeilijkheidsgraad en stijl** van het antwoord aan. Dit helpt je om antwoorden te krijgen die beter passen bij jouw doelgroep of bij jouw situatie.

□□ Voorbeeld

Prompt

Leg uit wat een `<form>`-element is in HTML, alsof je een docent bent die het uitlegt aan een groep mbo-studenten zonder programmeerervaring.

Verwachte AI-reactie

Een `<form>` is een stuk HTML-code waarmee gebruikers informatie kunnen invullen, zoals hun naam of e-mailadres. Denk aan een online formulier dat je invult om iets te bestellen. De informatie die je invult wordt verstuurd naar de website zodat er iets mee gedaan kan worden.

Laat de AI vragen stellen als een docent

Je kunt ook aan de AI vragen om **deelvragen** te stellen, zoals een goede docent dat zou doen.

Prompt

Speel de rol van een HTML-docent. Stel mij eerst drie vragen waarmee je kan inschatten hoeveel ik al weet over formulieren in HTML. Geef daarna pas een uitleg op mijn niveau.

Wat dit doet

- De AI stelt vragen zoals: “Wat denk je dat een `<form>` doet?”, “Heb je eerder inputvelden gebruikt?”
- De AI past daarna de uitleg aan op basis van jouw antwoorden (of een inschatting daarvan)

Opdracht

Je wilt dat de AI uitleg geeft over een klein stukje code, maar dan **in een specifieke rol**. Bijvoorbeeld: als docent, als programmeermaatje, of als code-reviewer.

1. Je bent een student die uitleg wil over hoe een formulier werkt in HTML
Je wilt dat de AI zich gedraagt als een docent die eerst **deelvragen** stelt voordat hij iets uitlegt
2. **Schrijf een prompt** waarin je duidelijk de rol beschrijft die de AI moet aannemen.

Inleveren

1. Je volledige prompt
2. Het AI-antwoord

2. *Vraag om alternatieven of variaties*

Uitleg

De AI geeft standaard vaak **slechts één oplossing**. Maar in de praktijk zijn er meestal meerdere manieren om iets te doen — zeker in programmeertalen als HTML, CSS, PHP of JavaScript.

Je kunt dus expliciet vragen om **meerdere oplossingen** of **varianties**, zoals:

- “Geef 3 manieren om een knop te centreren met CSS.”
- “Laat 2 varianten zien van hetzelfde formulier: één minimalistisch en één met veel opmaak.”
- “Bedenk 3 manieren om input te valideren in JavaScript.”

Voordelen

- Je leert alternatieve technieken kennen
- Je kunt zelf kiezen welke stijl of methode bij jouw project past
- Je voorkomt dat je AI-antwoorden blind overneemt zonder kritisch te zijn

▯▯ Voorbeeld

Prompt

Geef 3 manieren om een knop in het midden van het scherm te tonen met behulp van CSS.

Verwachte AI-reactie (beknopt)

1. `text-align: center` op de container + `display: inline-block` op de knop
2. `margin: 0 auto` bij een block-level knop met vaste breedte
3. Flexbox op de container met `justify-content: center` en `align-items: center`

👉 Opdracht

Je wil verschillende manieren zien om een veelvoorkomende webtaak uit te voeren.

1. Kies één van deze taken:
 - Validatie toevoegen aan een inputveld, of
 - Een PHP-script schrijven dat data toont

2. **Schrijf een prompt** waarin je de AI vraagt om **minstens 3 alternatieven of varianten** van die oplossing.
3. Gebruik in je prompt woorden als:
 - “Geef meerdere mogelijkheden...”
 - “Laat minstens 3 manieren zien...”
 - “Toon varianten van deze code...”
 - “Beschrijf van elke variant de voor- en nadelen....”
4. Zorg voor een goede opbouw in je prompt met:
 - **Context:** wat ben je aan het doen en welke technieken gebruik je?
 - **Details:** wat zijn de randvoorwaarden?
 - **Duidelijkheid:** hoeveel oplossingen wil je?

☐☐ Inleveren

1. Je volledige prompt
2. De gegenereerde alternatieven van de AI
3. Een korte reflectie (reflectie.txt): welke oplossing vind jij het beste, en waarom?

3. Iteratief verbeteren

☐☐ Uitleg

De AI geeft vaak een eerste versie van een antwoord, maar dat hoeft niet perfect te zijn. Je kunt de AI daarom vragen om **het eigen antwoord te beoordelen en te verbeteren**.

Dit heet *iteratief verbeteren*.

Bijvoorbeeld

- “Herschrijf dit zodat het eenvoudiger is voor een beginner.”
- “Geef dezelfde code, maar nu met uitleg erbij.”
- “Kun je deze oplossing optimaliseren voor leesbaarheid?”

Je leert zo hoe je feedback kunt geven aan de AI, en je oefent tegelijkertijd **zelfkritisch denken over codekwaliteit**.

▣▣ Voorbeeld

Prompt

*Geef me een formulier in HTML waarmee iemand zich kan inschrijven voor een nieuwsbrief.
Voeg daarna uitleg toe over hoe elk onderdeel werkt.*

Vervolgprompt

Herschrijf je antwoord zodat het ook begrijpelijk is voor iemand die nog nooit HTML heeft gezien.

👉 Opdracht

Je wilt dat de AI **het eigen werk verbetert** — of dat jij feedback geeft, en dat de AI dat verwerkt.

1. Vraag de AI eerst om een korte opdracht uit te voeren. Bijvoorbeeld:
 - Een stukje JavaScript schrijven dat controleert of een naam de juiste hoofdletters en punten heeft. (Dus goede namen zijn: M. Bisschop en J. K. van der Velden, onjuiste namen zijn M Bisschop (punt mist) en k. van der Velden (k is gene hoofdletter)).
2. Lees het antwoord goed door. Stel daarna een vervolgprompt waarin je de AI vraagt om:
 - het duidelijker te maken,
 - het aan te passen voor beginners,
 - het te herschrijven met uitleg/commentaar,
 - of het korter, leesbaarder of efficiënter te maken.
3. **Schrijf beide prompts uit:**
 - de eerste vraag, gebruik hierbij de few-shot prompt techniek (voorbeelden).
 - de verbeteropdracht
4. Zorg dat je feedback duidelijk is: **wat wil je precies anders of beter?**
5. **Zorg ervoor dat je de code begrijpt.**

Inleveren

1. Je eerste prompt
2. Het eerste AI-antwoord
3. Je tweede prompt (de verbeteropdracht)
4. Het verbeterde AI-antwoord

Deze opdracht laat je zien aan een docent. Je laat zien wat je eerste- en tweede eprompt is en je kunt zelf uitleggen hoe de code werkt.

---- vanaf hier nog corrigeren/aanvullen ----

4 Gebruik *prompt templates*

Uitleg

Als je regelmatig AI gebruikt voor soortgelijke taken, is het handig om te werken met een **vaste structuur**: een *prompt template*.

Een template helpt je om:

- niets te vergeten in je vraag;
- sneller en consistentere te werken;
- betere output te krijgen van de AI.

Een goede prompt bestaat vaak uit onderdelen zoals:

- **Context** - wat is de situatie?
- **Specifiek** - wat wil precies je bereiken?
- **Duidelijk** - zorg dat je prompt geen 'vage' aanwijzingen heeft
- **Outputvorm** - in welke vorm wil je het antwoord?
- **Toon** - formeel, informeel, eenvoudig, technisch?

Voorbeeld van een prompt template

Context: Ik werk aan een HTML-formulier

Doel: Ik wil dat de AI me helpt met het maken van een formulier voor e-mailinvoer

Outputvorm: Alleen HTML-code, zonder uitleg

Toon: Neutraal en duidelijk

Voorbeeldprompt:

Gebruik onderstaande context en geef alleen de code:

Ik wil een formulier voor e-mailadres, zonder uitleg.

Context: HTML-formulier

Opdracht

Je gaat zelf een prompt schrijven met behulp van een prompt-template.

1. Gebruik het volgende sjabloon en vul de velden in:

Context:
Doel:
Outputvorm:
Toon:

2. Schrijf daarna een **volledige prompt** op basis van jouw ingevulde template.
3. Stuur je prompt naar de AI en bekijk het resultaat:
 - Krijg je precies wat je bedoelde?
 - Is de toon en vorm correct?
 - Zou je iets willen toevoegen aan je template?
4. (Optioneel) Pas je template aan en probeer het nog eens.

Inleveren

1. Je ingevulde template
2. De volledige prompt
3. De output van de AI
4. Een korte reflectie: werkte het goed? Wat zou je verbeteren aan het sjabloon?

5. Vraag om zelfvragen (self-questioning)

Uitleg

Soms weet je nog niet goed **wat je precies moet vragen**, of wil je dat de AI eerst **nadenkt over het probleem** vóórdat het met een oplossing komt.

Dan kun je de AI vragen om eerst **zelf vragen te stellen** over de situatie of opdracht.

Deze techniek helpt om:

- een probleem beter te begrijpen
- complexe opdrachten op te breken
- geen belangrijke dingen over het hoofd te zien

Deze techniek lijkt op *chain-of-thought prompting*, maar het verschil is:

□ **De AI stelt eerst vragen aan zichzelf of aan jou** om het probleem helder te krijgen.

Voorbeeld

Prompt:

Ik wil een inschrijfformulier bouwen voor mijn website.

Stel jezelf eerst 3 vragen zodat je weet wat je precies moet bouwen.

Beantwoord die vragen.

Bouw daarna pas het formulier in HTML.

Verwachte AI-antwoord:

1. Welke gegevens moet de gebruiker invullen?
2. Moet de invoer gevalideerd worden?
3. Wat moet er gebeuren na het verzenden?

Daarna volgt uitleg én pas daarna de code.

Opdracht

Je wilt dat de AI een simpele programmeeropdracht uitvoert, maar eerst **zelf nadenkt over wat er nodig is**, door zichzelf (of jou) vragen te stellen.

1. Kies een opdracht, bijvoorbeeld:

- Maak een contactformulier
 - Valideer een inputveld met JavaScript
 - Toon resultaten in PHP
2. **Schrijf een prompt** waarin je de AI vraagt om:
 - eerst 3 vragen te stellen over de opdracht,
 - die vragen zelf te beantwoorden,
 - daarna pas te starten met uitleg en/of code
 3. Je kunt de prompt afsluiten met: *“Geef de code pas nadat je de vragen hebt beantwoord.”*

Inleveren

1. Je volledige prompt
2. De 3 vragen + antwoorden van de AI
3. De gegenereerde uitleg en code
4. Korte toelichting: hielp dit om het probleem beter te begrijpen?

6. Gebruik kritiek als leermoment (prompt debugging)

Uitleg

Niet elke prompt die je aan de AI geeft, levert direct het gewenste resultaat op. Soms krijg je:

- een onvolledig antwoord,
- verkeerde code,
- te algemene uitleg,
- of net niet wat je bedoelde.

In plaats van gefrustreerd te raken, kun je deze momenten gebruiken om **van je fout te leren**:

☐ Laat de AI zelf uitleggen waarom de prompt niet goed werkte, en hoe je die beter kunt formuleren.

Zo leer je hoe kleine verschillen in formulering grote impact kunnen hebben.

Voorbeeld

Slechte prompt:

Maak een mooie website

Verbeterde prompt:

Maak een eenvoudige, moderne HTML-pagina met een grote titel, een navigatiebalk bovenaan, en drie contentblokken onder elkaar. Gebruik alleen HTML en CSS, zonder externe libraries.

Prompt debugging:

*Waarom gaf de AI een vage reactie op “maak een mooie website”?
Wat zou ik anders moeten vragen?*

Opdracht

Je oefent met het herkennen en verbeteren van een slechte prompt.

1. **Schrijf eerst bewust een vage of slechte prompt**, zoals:
 - “Maak een website”
 - “Schrijf een formulier”
 - “Geef uitleg over PHP”
2. **Vraag vervolgens aan de AI:**
 - Waarom deze prompt niet optimaal is
 - Wat er beter zou kunnen aan deze prompt
 - Hoe je de prompt kunt herschrijven
3. **Schrijf daarna een verbeterde versie** van je prompt, en test het resultaat.
4. **Reflecteer:**
 - Wat is het verschil in output?
 - Wat leer je hiervan over duidelijkheid en specificiteit in prompts?

Inleveren

1. De oorspronkelijke (slechte) prompt
2. De analyse van de AI
3. De verbeterde prompt
4. De nieuwe output
5. Korte reflectie: wat is het belangrijkste inzicht?

Deployment

1 Toegang tot PLESK

Ga na bij een Software Development docent en meld dat je met deze module start.
De docent zal alles voor je klaarzetten!

Om te beginnen heb je toegang nodig tot PLESK. Hiervoor ga je met je browser naar het volgende adres:

<https://jouw.website/>

Vervolgens krijg je een inlogscherf. Klik hier op 'Wachtwoord vergeten?' om een link te krijgen voor het aanmaken van een wachtwoord. Je inlognaam is je studentnummer.

Om een wachtwoord te resetten (of op te vragen) gebruik je je email adres

studentnummer@talnet.nl, bijvoorbeeld : 123456@talnet.nl Vergeet niet je SPAM-box te checken.

Vergeet niet je SPAM-box te checken.

Als je bent ingelogd in PLESK, kom je gelijk in de beheeromgeving van jouw website. De url van jouw website is:

<jouw studentnummer>.jouw.website

Als je deze url voor de eerste keer intyped zie je een standaard (tijdelijke) pagina. Deze ga je in de volgende onderdelen vervangen door de door jou gemaakte website- / applicatie. Veel browsers zullen deze pagina niet direct laten zien. Dit komt omdat de site nog niet beveiligd is met een SSL certificaat (slotje bij je browser url).

Inleveren

- Screenshot van je browser met de PLESK-omgeving en zichtbare domeinnaam/URL.

2 Beveilig het domein met een SSL-certificaat

Start met het bekijken van je domein in de browser: `<studentNummer>.jouw.website`. Let op het slot-icoon in de adresbalk.

Stap 1

Voer jouw **domeinnaam** in, in je browser. Er kunnen afhankelijk van je browser, computer, virusscanner verschillende dingen gebeuren. Kijk wat je ziet. En kijk ook of er naast de domeinnaam / url in je adresbalk een slotje te zien is. Hoe ziet dat er uit?

Wat was ook alweer een domeinnaam?

Een **domeinnaam** is het adres van een website dat je intoetst in de browser, zoals:

www.google.com

Stap 2

Log in op je PLESK-omgeving en ga naar het onderdeel 'SSL/TLS-certificaten' onder 'Beveiliging'. Hier zie je verschillende mogelijkheden die je hebt om certificaten te installeren. Dit is per beheersysteem verschillend en ligt er aan hoe een beheersysteem is ingericht. Wij maken gebruik van het soort certificaat van 'Let's Encrypt'. Dit is een gratis open source certificaat dat vaak is geïnstalleerd op een beheersysteem (PLESK)

Wat is een certificaat?


Een **certificaat** op internet is een soort digitaal paspoort voor een website. Het bewijst dat de website een veilige verbinding maakt.

☐ Bijvoorbeeld:

Als je `https://` ziet en een slotje in de adresbalk, dan gebruikt de website een certificaat.

“ Een certificaat zorgt ervoor dat de verbinding tussen jou en de website veilig is (versleuteld), maar het garandeert **niet** dat de website zelf te vertrouwen is.

 Een nepwebsite (zoa `ing-bank-login.xyz`) kan ook gewoon een geldig certificaat hebben en een slotje tonen. Daarom moet je **altijd goed naar het adres kijken** en alleen websites vertrouwen die je kent of via een betrouwbare link bezoekt.

ound or type unknown

Stap 3: Klik onder aan de certificatenpagina op ' **Installeren** ' bij het onderdeel 'Let's Encrypt'

ound or type unknown

Hierna krijg je opties voor wat je op je domein wilt beveiligen. Het belangrijkste voor nu is de domeinnaam. Maar bijvoorbeeld mail kun je hier ook laten versleutelen met een certificaat.

Stap 4

Klik aan wat je nodig hebt. Je kunt alles aanvinken behalve de 'Wildcard'optie. Hiervoor zijn meerdere handelingen nodig.

Wat is een wildcard?

Een wildcard-optie betekent dat je met één instelling o meerdere variaties in één keer dekt.

Een * staat bij een wildcard voor "alles" of "maakt niet uit wat hier staat".

Stap 5

Bekijk de site nogmaals in je browser (eventueel refreshen) en kijk of je verschil ziet en of het domein is beveiligd.

Stap 6

Maak een screenshot van je browser met hier in de adresbalk waar jou domeinnaam + het beveiligingsslotje goed zichtbaar is. Stuur ook jouw url in.

Reflectie – SSL-certificaat installeren

- Wat is het doel van een SSL-certificaat?
- Wat verandert er zichtbaar in je browser als een site beveiligd is?
- Wat kan er misgaan als een website géén SSL-certificaat heeft (zoek dit eventueel op)?
- Welk type certificaat heb je gekozen en waarom was de 'Wildcard' optie niet geschikt?

☐☐ Inleveren

- Lever je reflectie in als .txt of .pdf-bestand.
- Screenshot van de browser met het beveiligde domein (slotje zichtbaar).
- URL van je domein.

3 Maak een FTP-account aan

Maak via PLESK een FTP-account met zelfgekozen gebruikersnaam en wachtwoord. Stel de hoofdmap in op de root-directory van je (live) website.

☐☐ Reflectie – FTP-account aanmaken

- Wat is het doel van een FTP-account bij het beheren van je website?
- Welke gegevens moet je onthouden of noteren bij het aanmaken van een FTP-account?

☐☐ Inleveren

- Lever je reflectie in als .txt of .pdf-bestand.

4 FTP-verbinding maken en bestanden plaatsen

↓ FTP-client downloaden en installeren

Je mag natuurlijk alle programma's die je fijn vindt gebruiken. In de uitleg gaan we kijken naar de client van FileZilla (zelfde organisatie als FireFox). Dit is een gratis programma dat je hier kunt downloaden:

<https://filezilla-project.org/download.php?type=client>

! let er op dat je de **clientversie** gebruikt.

Installeer dit programma op je computer.

📁FTP-verbinding maken

Nu je de FTP-client hebt geïnstalleerd gaan we een verbinding maken met jouw domein op de server. Om dit te doen heb je 3 gegevens nodig:

- **Host** (server)
Hier kun je je domeinnaam invoeren: <studentnummer>.jouw.website
- **Gebruiker**
(zoals je deze hebt aangemaakt bij het maken van een ftp-account op de server)
- **Wachtwoord**
(zoals je deze hebt aangemaakt bij het maken van een ftp-account op de server)

Volg onderstaande stappen om een verbinding te maken:

- open FileZilla
- open 'sitebeheer' via:
bestanden -> sitebeheer
- klik op 'nieuwe site'
- geef de site een naam in het linker overzichtsscherm
- Voer je host, je gebruikersnaam en wachtwoord in
- klik je op 'verbinden'

filezilla.png
image not found or type unknown

Als het goed is, ben je nu ingelogd en zie je de inhoud van je servermap waar je een site kunt plaatsen.

filezilla2.png
image not found or type unknown

Verwijder de bestanden die je ziet op de server.

Maak een eigen index.html bestand aan op je computer. Maak hier een leuke pagina van met html-code. Plaats deze vervolgens op de server via FTP. Als je FileZilla gebruikt zie je in het linker gedeelte van je scherm, bestanden op jouw computer en rechts die op de server.

Bekijk je nieuwe pagina in je browser door naar je site te gaan!

Mooi, he!?

☐☐ Reflectie – FTP-account aanmaken

- Wat kan er gebeuren als je een verkeerde hoofdmap instelt bij je FTP-account?
- Wat zou je een volgende keer anders aanpakken bij het instellen van FTP-toegang?

☐☐ Inleveren

- URL van jouw domein met de nieuwe indexpagina.
- Lever je reflectie in als .txt of .pdf-bestand.

5 Database aanmaken

Maak in PLESK een nieuwe database én gebruiker aan:

Geef de database de naam:

Je ziet de database in het overzicht; open phpMyAdmin daarvoor.

☐☐ Inleveren

- Screenshot van phpMyAdmin in je browser inclusief adresbalk en database-naam.

6 Webapplicatie installeren

In de [CRUD Challenge](#) heb je een applicatie gemaakt, waarmee geregistreerd kon worden wie te laat in de klas kwam en de reden daarvan. Weet je nog!? Dit ga je nu live zetten op jouw server.

image.png
image not found or type unknown

Nu is het tijd om jouw applicatie 'live' te zetten op de server, zodat iedereen via Internet er gebruik van kan maken. ! Hou er rekening mee dat het kan zijn dat je code moet aanpassen om een goede verbinding te maken met de database. Tip: Net als in xampp, kun je als hostname / servername 'localhost' gebruiken. Inleveren : De url van jouw (werkende) applicatie

☐ Let op: je moet mogelijk de database-connectie in je code aanpassen (bijv. hostname = 'localhost').

Inleveren

- URL van je live, werkende applicatie.


7 Test je Kennis

Wat is Plesk?

Plesk is een soort **bedieningspaneel** voor je website. Je kunt het vergelijken met het dashboard van een auto, maar dan voor je webserver.

Wat kun je ermee doen?

- Je eigen **domeinnaam** instellen (zoals `www.jouwsite.nl`)
- **Bestanden uploaden** van je website
- Een **database aanmaken**
- Een **SSL-certificaat** installeren (voor de  in je adresbalk)
- **E-mailadressen beheren** (zoals `info@jouwsite.nl`)

 Plesk maakt het dus makkelijk om je website online te zetten en te beheren, zonder dat je moeilijke servercommando's hoeft te leren.

Wat doet een SSL-certificaat en waarom is dat belangrijk?

Een SSL-certificaat zorgt ervoor dat gegevens tussen bezoeker en website versleuteld zijn.

Je herkent het aan de  in de adresbalk. Het beschermt tegen af luisteren en verhoogt vertrouwen, maar zegt niets over de betrouwbaarheid van de site zelf.

Wat betekent 'wildcard'-optie bij certificaten en waarom kies je die niet meteen?

Een wildcard-certificaat dekt alle subdomeinen (bijv. `*.jouw.website`). Voor eenvoudige sites heb je die vaak niet nodig, en het vereist extra configuratie, dus kies je meestal alleen jouw domein.

Waarom moet je een FTP-account aanmaken en waar is dit goed voor?

Met een FTP-account kun je verbinding maken met de server en bestanden (zoals je website) uploaden of bijwerken. Het is dé manier om jouw code 'live' te zetten.

Waarom is het nodig de database-connectie opnieuw in te stellen bij deployment van je CRUD-applicatie?

In je lokale development omgeving gebruik je meestal gebruiker 'root' zonder wachtwoord (lekker makkelijk!).

Op de productieserver moet je de het wachtwoord en liefst ook de gebruikersnaam aanpassen om te voorkomen dat je database makkelijker wordt gehakkeld en dat de informatie in verkeerde handen komt.

CMS

Status: helemaal doorlopen/testn, plaatjes zijn mogelijk niet goed overgenomen?

1 Wat is een CMS?

□□ Leerdoelen

- Je weet wat een CMS is.
- Je kunt uitleggen wat je met een CMS zoals WordPress kunt doen.
- Je begrijpt waarom veel websites met een CMS gebouwd worden.

□□ Uitleg

CMS staat voor *Content Management System*. Een CMS is een programma waarmee je op een eenvoudige manier een website kunt maken en beheren.

Je kunt met een CMS internetpagina's aanmaken, plaatjes toevoegen, navigatiemenu's maken en zelfs extra functies installeren – allemaal zonder te programmeren.

Een CMS bestaat meestal uit een basispakket (de 'kern') en uitbreidingen. Je kunt bijvoorbeeld thema's toevoegen om het uiterlijk aan te passen, of plugins om functionaliteit toe te voegen zoals een webshop, contactformulier of nieuwsbrief.

Populaire voorbeelden van CMS-systemen zijn **WordPress**, **Drupal** en **Joomla**.

In deze module leer je werken met het meest gebruikte CMS ter wereld: **WordPress**.

WordPress is open-source software. Dat betekent dat je het gratis mag gebruiken en aanpassen. Je kunt WordPress zien als een grote doos met Lego-blokken:

- De basisblokken vormen je website: pagina's, menu's, afbeeldingen, enzovoort.
- Met uitbreidingen (plugins en thema's) geef je de site een nieuw uiterlijk, of voeg je functies toe zoals een online winkel.

WordPress maakt het dus mogelijk om heel snel een professionele website te bouwen – zonder dat je technische kennis nodig hebt.

2 WordPress installeren op je eigen laptop

□□ Leerdoelen

- Je kunt zelfstandig WordPress installeren op je eigen laptop.
- Je kunt een database aanmaken met phpMyAdmin.
- Je begrijpt welke gegevens je nodig hebt tijdens de installatie.

□□ Uitleg

In dit onderdeel ga je een **lokale** WordPress-website maken op jouw laptop met behulp van `XAMPP`. Dit betekent dat je website alleen werkt op jouw eigen laptop. Voor het maken van een WordPress-site zijn een paar stappen nodig. In eerdere modules heb je al met XAMPP gewerkt, dus we gaan ervan uit dat deze al geïnstalleerd is op je computer.

Stap 1 - Database aanmaken

Open **phpMyAdmin** via <localhost/phpmyadmin>. Maak een nieuwe database aan. Bedenk zelf een geschikte naam (bijv. `wp-jan`).

Stap 2 - WordPress downloaden

Download de laatste versie van WordPress van de officiële site:

<https://wordpress.org/download/>

Let op: Download het NIET van wordpress.com!

Sla het ZIP-bestand op in je `htdocs`-map van XAMPP (meestal: `C:\xampp\htdocs`).

□□ Opdracht 1 – WordPress map instellen

1. Pak het ZIP-bestand uit dat je hebt gedownload.
2. Er ontstaat een map met de naam `wordpress`.
3. Hernoem deze map naar `wp-<jouwnaam>` (vervang `<jouwnaam>` door jouw eigen naam, zonder spaties!).

Voorbeeld: `wp-ayoub` of `wp-nina`

We gaan nu verder met het installatieproces. Surf in je browser naar: `http://localhost/wp-jouwnaam`

Tijdens de installatie moet je deze gegevens invullen:

- **Naam van de database:** de naam die je net hebt aangemaakt
- **Gebruikersnaam database:** `root`
- **Wachtwoord database:** laat dit veld **leeg**
- **Host database:** `localhost`
- **Tabelvoorvoegsel:** laat dit staan zoals het is (`wp_`)
- **Titel van de site:** bijv. “De website van Ayoub”
- **Gebruikersnaam en wachtwoord** om in te loggen op je WordPress-site
- **Je e-mailadres:** nodig om je wachtwoord later te herstellen

☐ Reflectie

- Wat was het makkelijkste onderdeel van deze installatie?
- Wat vond je lastig of verwarrend?
- Wat is volgens jou het voordeel van lokaal werken ten opzichte van een online installatie?

☐ Inleveren

- Maak een lijstje met de gegevens die jij hebt gebruikt tijdens de installatie (zie hierboven).
Sla dit op als `installatiegegevens-jouwnaam.pdf` (bijv. `installatiegegevens-ayoub.pdf`) en lever die in.

- Reflectie met daarin de antwoorden op de drie vragen (txt of PDF).

3 WordPress installatie uitvoeren

Leerdoelen

- Je kunt het installatieproces van WordPress starten via je browser.
- Je weet hoe je Apache en MySQL moet starten in XAMPP.
- Je kunt inloggen op je eigen WordPress-site via de juiste URL.

Uitleg

Je bent nu klaar om WordPress te installeren. Dit doe je in je browser, maar eerst moet je controleren of de juiste onderdelen van XAMPP actief zijn:

- Zorg dat **Apache** en **MySQL** zijn gestart in de XAMPP-console. Ze moeten groen zijn.

image.png
image not found or type unknown

Open nu je browser en ga naar:

`http://localhost/wp-<jouwnaam>`

Voorbeeld: `http://localhost/wp-ayoub`

Je ziet nu het installatieproces van WordPress in je browser. Lees alles zorgvuldig en vul de gevraagde gegevens correct in. Gebruik hierbij de informatie die je eerder hebt genoteerd (zie opdracht 1).

Na de installatie:

- Je krijgt een bevestiging te zien dat WordPress is geïnstalleerd.
- Je kunt direct inloggen met de gebruikersnaam en het wachtwoord die je tijdens de installatie hebt opgegeven.
- Als je deze link niet meer ziet, kun je inloggen via:

`http://localhost/wp-<jouwnaam>/wp-admin`

Opdracht – Installatie controleren

1. Log in op WordPress via `http://localhost/wp-jouwnaam/wp-admin`.
2. Maak een screenshot van je browser, waarop de WordPress-beheerderspagina zichtbaar is, met jouw naam in de adresbalk.
3. Open ook [phpMyAdmin](#) en klik op de database die je net hebt aangemaakt.
4. Maak een screenshot van de tabellenstructuur van jouw WordPress-database.

Inleveren

- `inloggen-gelukt-jouwnaam.jpg`: Screenshot van jouw WordPress-dashboard met jouw naam zichtbaar in de adresbalk.
- `phpmyadmin-jouwnaam.jpg`: Screenshot van jouw WordPress-database in phpMyAdmin met tabellen zichtbaar.

4 Frontend en Backend in WordPress

Leerdoelen

- Je begrijpt het verschil tussen de frontend en backend van een WordPress-site.
- Je weet hoe je kunt inloggen op de backend (beheeromgeving).
- Je kunt schakelen tussen frontend en backend via de admin-balk.

Uitleg

In WordPress werk je in twee verschillende omgevingen:

1. De backend

De backend is het beheergedeelte van je site. Hier kun je onder andere:

- Nieuwe pagina's aanmaken
- Functies toevoegen via plugins
- Gebruikers beheren
- Thema's (templates) installeren voor het uiterlijk van je site

Je hebt een account nodig om toegang te krijgen tot de backend. Dat account heb je aangemaakt tijdens de installatie.

De standaard URL om in te loggen op de backend is:

```
http://sitenaam.nl/wp-admin
```

Als je lokaal werkt is dit bijvoorbeeld:

```
http://localhost/wp-jouwnaam/wp-admin
```

2. De frontend

De frontend is de website zoals bezoekers deze zien. Iedereen kan deze omgeving bekijken – je hebt geen wachtwoord nodig, tenzij je dat instelt.

Switchen tussen frontend en backend

Als je bent ingelogd, zie je bovenaan een zwarte balk. Deze admin-balk is zichtbaar op alle pagina's, zowel in de backend als de frontend.

Met deze balk kun je makkelijk wisselen tussen beide omgevingen.

☐ Manier 1 – Naar de homepage of het dashboard

- Klik linksboven in de zwarte balk op het icoon naast het WordPress-logo.
- Om naar de **homepage** te gaan: klik op het **huis-icoontje**.
- Om naar het **dashboard** te gaan: klik op het **dashboard-icoontje**.

image.png
image not found or type unknown

☐ Probeer dit zelf een paar keer:

Wissel tussen de backend en frontend van jouw site met behulp van de zwarte balk.

☐ Manier 2 – Direct naar het onderdeel dat je wilt bewerken

Als je bijvoorbeeld een pagina bekijkt en je wilt de inhoud wijzigen, kun je via de zwarte balk direct naar de bewerkingspagina in de backend gaan.

Deze functie maakt het snel en makkelijk om onderdelen van je site aan te passen terwijl je ze bekijkt.

Hoe dit precies werkt, kom je later in deze module nog uitgebreid tegen!

☐☐ Controlevragen

- Wat is het verschil tussen frontend en backend?
- Waarvoor gebruik je de zwarte admin-balk?
- Waarom is het handig om direct vanuit de frontend naar het bewerkvenster te gaan?

☐☐ Inleveren

- Antwoord op de controlevragen (pdf of txt).

Welke onderdelen van Wordpress gaan we in deze module gebruiken?

image.png and or type unknown

5 Pagina's maken en aanpassen in WordPress

☐☐ Leerdoelen

- Je weet hoe pagina's werken in WordPress.
- Je kunt pagina's bekijken en bewerken vanuit de frontend én de backend.

- Je begrijpt het verschil tussen pagina's in WordPress en losse HTML-bestanden.

□□ Uitleg

Als software developer heb je waarschijnlijk al vaak webpagina's gemaakt met HTML en CSS. In WordPress kan dat ook, maar het hoeft niet!

In plaats van losse HTML-pagina's en CSS-bestanden gebruikt WordPress een andere manier. Je maakt en bewerkt pagina's gewoon via de backend, met een visuele editor. Je hoeft dus geen code te schrijven – al mag dat natuurlijk wel als je wilt!

Belangrijk om te onthouden:

- Je hoeft niet technisch te zijn om een mooie website te maken in WordPress.
- Maar als je wél technische kennis hebt, kun je die gebruiken voor extra aanpassingen.

Frontend ↔ Backend gebruiken

Via de zwarte admin-balk kun je gemakkelijk wisselen tussen de frontend en backend van je site. Zoals je eerder hebt geleerd, kun je dan heen en weer gaan tussen de homepage (frontend) en het dashboard (backend).

Maar soms wil je direct naar een specifieke pagina in de backend, bijvoorbeeld als je een stuk tekst wilt aanpassen op een bepaalde pagina.

Direct bewerken via de frontend

Als je bent ingelogd en je bekijkt een pagina via de frontend, dan zie je bovenaan een zwarte balk. Deze bevat een knop waarmee je direct naar het bewerkvenster van die pagina in de backend kunt gaan.

Dit is handig, want zo hoef je de pagina niet eerst handmatig op te zoeken in het dashboard.

Voorbeeldpagina bij installatie

Bij de installatie van WordPress wordt er automatisch een voorbeeldpagina aangemaakt. Die kun je bekijken via de volgende URL:

`http://localhost/wp-<jouw-naam>/voorbeeld-pagina`

Bijvoorbeeld: `http://localhost/wp-ayoub/voorbeeld-pagina`

Opdracht – Voorbeeldpagina bekijken en bewerken

1. Ga naar `http://localhost/wp-jouwnaam/voorbeeld-pagina`.
2. Bekijk de pagina zoals een bezoeker deze zou zien.
3. Klik op de zwarte balk bovenin op **Bewerken** om de backend-versie van de pagina te openen.
4. Pas de titel van de pagina aan naar bijvoorbeeld: “Welkom op mijn WordPress-site”.
5. Pas ook de tekst op de pagina aan (bijvoorbeeld je eigen naam toevoegen).
6. Klik op **Bijwerken** om je aanpassingen op te slaan.
7. Bekijk de pagina opnieuw aan de frontend: zijn de wijzigingen zichtbaar?

Controle vragen

- Wat is het voordeel van werken met WordPress-pagina's ten opzichte van losse HTML-bestanden?
- Wat vond je makkelijker: bewerken via de frontend of backend? Waarom?
- Welke rol speelt de zwarte admin-balk bij het onderhouden van pagina's?

Inleveren

- Maak een screenshot van jouw aangepaste voorbeeldpagina in de frontend (met jouw aanpassing zichtbaar).
- Sla het bestand op als `voorbeeldpagina-jouwnaam.jpg`.

6 Opdracht – Standaard pagina aanpassen

Opdracht 3 – Voorbeeldpagina aanpassen

1. Ga naar de voorbeeldpagina van jouw site:
`http://localhost/wp-<jouw-naam>/voorbeeld-pagina`
2. Klik bovenaan in de zwarte balk op **Pagina bewerken**.
3. Je komt nu in het bewerkingsscherm van de pagina. Klik op de tekst en blokken en vervang de standaardtekst door een **introductieverhaal over jezelf als student software developer**.
4. Pas ook de **titel** van de pagina aan naar iets origineels, bijvoorbeeld:
“Welkom op mijn portfolio” of “Dit ben ik – Toekomstig developer”
5. Gebruik de editor en probeer zelf te ontdekken hoe je tekstblokken wijzigt.

image.png
image not found or type unknown

☐ Zelf proberen!

Probeer eerst zelf uit te zoeken hoe je de tekst aanpast. Kom je er echt niet uit, kijk dan op internet of vraag hulp.

Vergeet niet op de knop **Bijwerken** te klikken zodat je wijzigingen worden opgeslagen!

Als je klaar bent, kun je de editor afsluiten door linksboven op het zwart/witte WordPress-logo te klikken.

image.png
image not found or type unknown

Ga daarna weer naar:

`http://localhost/wp-<jouw-naam>/voorbeeld-pagina`

Als het goed is zie je daar nu jouw eigen tekst en titel terug!

☐ Inleveren

- Maak een screenshot van de aangepaste pagina in je browser, met de URL zichtbaar waarin jouw naam voorkomt.
- Sla de afbeelding op als: `voorbeeld-pagina-<jouw-naam>.jpg`
- Lever dit bestand in via het LMS of per mail.

7 Pagina verder aanpassen

Opdracht 4 – Afbeeldingen toevoegen aan je pagina

We gaan de voorbeeldpagina nog mooier maken door afbeeldingen toe te voegen.

1. Open je pagina opnieuw via:
`http://localhost/wp-<jouw-naam>/voorbeeld-pagina`
2. Klik bovenaan op **Pagina bewerken**.
3. Voeg minstens **drie afbeeldingen** toe aan de pagina. Je mag zelf kiezen welke, zolang ze bij jouw introductieverhaal passen.
4. Upload afbeeldingen vanaf je computer via de WordPress editor.

Een afbeelding toevoegen in de tekst

Zet je cursor op de plek waar je een afbeelding wilt plaatsen. Klik dan op het pijltje omlaag bij een bestaand blok en kies **Inline afbeelding**. Selecteer daarna een afbeelding van je computer en voeg deze toe.

image.png
image.png and or type unknown

Een nieuw blok maken met een afbeelding

Je kunt ook een nieuw blok toevoegen. Klik op het **blauwe plusje** onder aan de pagina of tussen bestaande blokken.

image.png
image.png and or type unknown

Kies dan voor **Afbeelding** en upload een bestand vanaf je computer.

Inleveren

- Maak een screenshot van je voorbeeldpagina nadat je de drie afbeeldingen hebt toegevoegd.
- Zorg dat de URL zichtbaar is in de screenshot, met daarin jouw naam.
- Sla de afbeelding op als: `voorbeeld-pagina-<jouw-naam>.jpg`
- Lever het bestand in via het LMS of mail het naar je docent.

8 Nieuwe pagina's aanmaken

Opdracht 5 – Contact en Homepage aanmaken

Je gaat nu zelf twee nieuwe pagina's maken:

- Een pagina met de titel **Homepage**
- Een pagina met de titel **Contact**

Stap voor stap:

1. Zorg dat je bent ingelogd op je site.
2. Klik in de zwarte balk bovenin op **Nieuw → Pagina**. 
3. Geef de pagina de juiste titel: "Homepage" of "Contact".
4. Voeg minimaal **twee tekstblokken** toe.
5. Voeg daarnaast ook **twee andere soorten blokken** toe. Dit mogen bijvoorbeeld zijn: een afbeelding, knop, lijst, video, citaat, enzovoort.
6. Klik op **Publiceren** of **Bijwerken** om de pagina op te slaan.

Herhaal deze stappen voor de tweede pagina.

Inleveren

- Maak een screenshot van elke nieuwe pagina met de URL zichtbaar in de adresbalk.
- Bestandsnamen:
 - `contact-<jouw-naam>.jpg`
 - `homepage-<jouw-naam>.jpg`
- Lever deze twee afbeeldingen in via het LMS of per mail.
- Stuur ook de URL's van beide pagina's mee. Bijvoorbeeld:
 - `http://localhost/wp-jouwnaam/contact`
 - `http://localhost/wp-jouwnaam/homepage`

9 Site-instellingen en Homepage instellen

□□ Leerdoelen

- Je weet waar je de algemene instellingen van je WordPress-site kunt vinden.
- Je kunt de homepage van je website instellen via het menu “Lezen”.
- Je begrijpt het verschil tussen de standaard blogweergave en een statische pagina als homepage.

□□ Uitleg

Via de instellingen in de **backend** kun je veel algemene eigenschappen van je site beheren.

1. Algemeen

Via **Instellingen** → **Algemeen** kun je onder andere:

- De **titel** en ondertitel van je site aanpassen
- De **taal** van je WordPress-installatie wijzigen
- Het **datumformaat** en **tijdformaat** instellen

2. Lezen

Via **Instellingen** → **Lezen** kun je:

- Bepalen wat de **homepage** van je site is
- Instellen of **zoekmachines** (zoals Google) jouw site mogen indexeren

Homepage instellen

Je hebt nu drie pagina's aangemaakt: een voorbeeldpagina, een homepage en een contactpagina. Standaard laat WordPress de *blogberichten* zien op de hoofdpagina van je site. Dat gaan we nu aanpassen.

Als iemand naar jouw hoofdpagina surft (bijvoorbeeld `http://localhost/wp-<jouw-naam>/`) wil je natuurlijk dat hij jouw zelfgemaakte **Homepage**-pagina ziet!

Opdracht 6 – Homepage instellen

1. Ga naar de backend van je WordPress-site.
2. Navigeer naar **Instellingen → Lezen**.
3. Kies bij **“Je homepage toont”** voor: `Een statische pagina`.
4. Selecteer bij **Homepage** jouw zelfgemaakte pagina met de titel “Homepage”.
5. Sla je wijzigingen op met de knop **Wijzigingen opslaan**.

image.png
image not found or type unknown

Inleveren

- Surf naar `http://localhost/wp-<jouw-naam>/` en controleer of jouw homepage daar zichtbaar is.
- Maak een screenshot van deze pagina, mét zichtbare URL in de adresbalk.
- Sla het bestand op als: `homepage-instelling-<jouw-naam>.jpg`
- Lever de afbeelding in.

Vervolgopdracht – Zet jouw gemaakte homepage actief

Zorg dat de door jou gemaakte pagina met de titel **Homepage** (opdracht 5) daadwerkelijk wordt ingesteld als de homepage van jouw site.

Hiervoor gebruik je opnieuw:

- **Instellingen → Lezen** in de backend
- Selecteer bij **“Je homepage toont”** de pagina “Homepage”

Inleveren

- Ga naar: `http://localhost/wp-<jouw-naam>/`
- Controleer of daar nu jouw eigen gemaakte homepage zichtbaar is.
- Maak een screenshot van deze pagina, met de URL zichtbaar in de adresbalk.
- Sla de afbeelding op als: `homepage-<jouw-naam>.jpg`
- Lever het bestand in via het LMS of per e-mail aan je docent.

2 WordPress uitbreiden met een thema

□□ Leerdoelen

- Je begrijpt wat een WordPress-thema is.
- Je weet hoe je een ander thema kiest en installeert in WordPress.
- Je kunt het uiterlijk van je site aanpassen met een thema.

□□ Uitleg

In de vorige opdracht heb je een WordPress-site gemaakt en drie pagina's aangemaakt. Misschien vond je de site er nog een beetje saai uitzien. Geen zorgen: daar hebben we **thema's** voor!

Het **thema** bepaalt hoe jouw website eruitziet: het ontwerp, de kleuren, de lettertypes en de layout.

Je kunt WordPress zien als een bouwpakket met *Lego-blokken*. De software zelf is het grote basisblok. Een thema is één van die bouwblokken die je kunt toevoegen om je site mooier en persoonlijker te maken.

Er zijn drie manieren om een thema toe te voegen:

- Je maakt zelf een thema en uploadt het.
- Je koopt een professioneel thema en installeert dat.
- Je kiest een gratis thema uit de *thema-bibliotheek* binnen WordPress.

Standaard is er al een thema actief als je WordPress installeert. Meestal heet dat bijvoorbeeld `Twenty Twenty-Three` of `Twenty Twenty-Four`. Ook zijn er vaak al een paar andere thema's geïnstalleerd, die je meteen kunt activeren.

Zo activeer je een ander thema:

1. Log in op de **WordPress-backend**.
2. Ga in het linkermenu naar `Weergave` → `Thema's`.
3. Je ziet een overzicht van de beschikbare thema's.
4. Beweeg je muis over een thema en klik op **Activeren**.

Je website verandert dan direct van uiterlijk. De indeling verandert, de kleuren kunnen anders zijn, en het lettertype wordt aangepast. Ook worden andere CSS-bestanden gebruikt.

Thema aanpassen met de customizer

Bij het actieve thema zie je vaak een knop `Customizer` of `Thema aanpassen`. Hiermee kun je het thema verder instellen. Denk aan:

- Je eigen logo of site-icoon toevoegen
- Kleuren en lettertypes aanpassen
- Achtergrondafbeelding kiezen
- Eigen CSS toevoegen (voor gevorderden)

Als je op **Nieuwe toevoegen** klikt, kom je in een soort *themawinkel* terecht. Daar kun je zoeken naar gratis thema's. Dit lijkt een beetje op de App Store of de Play Store op je telefoon. De meeste thema's zijn gratis en direct te installeren.

☐☐ Opdracht – Thema installeren en aanpassen

1. Log in op jouw WordPress-site.
2. Ga naar `Weergave` → `Thema's`.
3. Kies een ander standaardthema (bijv. Twenty Twenty-Four) en activeer dit.
4. Klik op `Thema aanpassen` en probeer minimaal 3 instellingen te veranderen (kleur, titel, lettertype, etc).

5. Voeg een nieuw thema toe via de knop `Nieuwe toevoegen`.
6. Kies een gratis thema dat je mooi vindt en installeer en activeer dit.

☐☐ Controlevragen

- Wat is het verschil tussen een thema en een plugin?
- Wat is het voordeel van een customizer in WordPress?
- Hoe kun je jouw site persoonlijker maken met een thema?

☐☐ Inleveren

- Maak een screenshot van jouw aangepaste homepage met het nieuwe thema.
- Lever ook de controlevragen (.txt of .pdf) in met jouw antwoorden op de drie vragen.

☐☐ Opdracht 2 – Twee thema's installeren en testen

1. Open de backend van je WordPress-website (`http://localhost/wp-<jouw-naam>/wp-admin`).
2. Ga naar **Weergave → Thema's**.
3. Klik op **Nieuwe toevoegen** en kies twee gratis thema's die jij mooi vindt.
4. Installeer én activeer deze twee thema's, één voor één.
5. Bekijk steeds je homepage in je browser terwijl het thema actief is.

☐☐ Inleveren

- Maak **twee screenshots** van je homepage, telkens met een ander thema actief.
- Op het screenshot moet ook de **URL zichtbaar zijn**, bijvoorbeeld: `http://localhost/wp-jan/`.
- Sla de bestanden op met deze naamstructuur:
`homepage-<jouw-naam>-<thema-naam>.jpg`
`homepage-<jouw-naam>-<thema-naam>.jpg`

- Na het maken van de screenshots, activeer je weer het standaardthema **Twenty Twenty-One**.

Opdracht 2 – Twee thema's installeren en testen

1. Open de backend van je WordPress-website (`http://localhost/wp-<jouw-naam>/wp-admin`).
2. Ga naar **Weergave → Thema's**.
3. Klik op **Nieuwe toevoegen** en kies twee gratis thema's die jij mooi vindt.
4. Installeer én activeer deze twee thema's, één voor één.
5. Bekijk steeds je homepage in je browser terwijl het thema actief is.

Inleveren

- Maak **twee screenshots** van je homepage, telkens met een ander thema actief.
- Op het screenshot moet ook de **URL zichtbaar zijn**, bijvoorbeeld: `http://localhost/wp-jan/`.
- Sla de bestanden op met deze naamstructuur:
`homepage-<jouw-naam>-<thema-naam>.jpg`
`homepage-<jouw-naam>-<thema-naam>.jpg`
- Na het maken van de screenshots, activeer je weer het standaardthema **Twenty Twenty-One**.

4 Hoofdnavigatiemenu en extra menu

Leerdoelen

- Je kunt zelf een navigatiemenu samenstellen in WordPress.
- Je weet hoe je een hoofdmenu en een tweede menu maakt.

- Je kunt links groeperen en submenu's maken in WordPress.

Uitleg

Een **hoofdnavigatiemenu** is het menu dat bovenaan je website staat en bezoekers helpt om te navigeren tussen de belangrijkste onderdelen van je site.

In WordPress kun je meerdere menu's aanmaken, bijvoorbeeld één als hoofdmenu en één als apart menu voor andere links (zoals naar je favoriete games of muziek).

Je kunt ook submenu's (uitklapmenu's) maken door één of meer links onder een andere link te laten inspringen.

Opdracht – Maak een hoofdmenu en ext menu

1. Ga naar → .
2. Maak een nieuw menu aan en noem dit **'Mijn hoofdmenu'**.
3. Voeg de volgende links toe aan dit menu:
 - Home
 - Contact
 - Voorbeeldpagina (of hoe je die zelf genoemd hebt)
 - **MBO College Amstelland** – met een link naar:
<https://www.rocva.nl/MBO-onderwijs/MBO-Colleges/MBO-College-Amstelland>
4. Zorg ervoor dat één of meerdere links **inspringen** zodat er een uitklapmenu ontstaat. (Bijvoorbeeld: maak 'Contact' een submenu van 'Home')
5. Vink aan dat dit het **Hoofdmenu** (Primary Menu) is en klik op **Menu opslaan**.
6. Maak daarna een tweede menu aan met een zelfgekozen naam (bijvoorbeeld 'Favorieten').
7. Voeg hierin minstens 5 externe links toe, bijvoorbeeld naar je favoriete:
 - Muziekartiest of band
 - Gamesites
 - Sportclubs

8. Wijs dit tweede menu NIET toe als hoofdmenu.
9. Voeg het tweede menu toe op een plek waar jouw thema dat toelaat (bijv. in een widget, footer of zijbalk).

☐☐ Controlevragen

- Wat is het voordeel van een submenu?
- Wanneer is het handig om een tweede menu te gebruiken?
- Wat heb je geleerd over het gebruik van externe links?

☐☐ Inleveren

- Maak een **screenshot van de homepage** waarop beide menu's zichtbaar zijn.
- Geef het bestand deze naam: `menus-<jouw-naam>.jpg`
- Lever de afbeelding in via de gebruikelijke manier.

5 WordPress uitbreiden met plugins

☐☐ Leerdoelen

- Je weet wat een plugin is in WordPress.
- Je kunt zelf een plugin installeren via de zoekfunctie.
- Je begrijpt dat plugins extra functies toevoegen aan je website.

☐☐ Uitleg

In het vorige onderdeel heb je jouw website in WordPress een nieuw **uiterlijk** gegeven door een thema te installeren. Nu gaan we kijken naar hoe je je website **extra functionaliteit** kunt geven

met **plugins**.

Plugins kun je zien als extra *Lego-blokken* die je aan WordPress toevoegt. Waar een thema vooral gaat over het uiterlijk van je site, voegen plugins functies toe.

Voorbeelden van functies die je met een plugin kunt toevoegen:

- Een webshop (bijv. WooCommerce)
- Een contactformulier
- Een agenda of boekingssysteem
- Zoekmachineoptimalisatie (SEO)
- Beveiliging of back-ups

Net als bij thema's zijn er drie manieren om plugins toe te voegen aan je WordPress-site:

- Je maakt zelf een plugin en uploadt die
- Je koopt een plugin bij een ontwikkelaar en installeert deze handmatig
- Je zoekt en installeert een plugin via de ingebouwde **pluginzoekfunctie**

In deze les gebruiken we de derde methode: een plugin zoeken en installeren via de **plugin-bibliotheek** van WordPress. Dit werkt eigenlijk net als in de **App Store** of **Google Play Store**.

Heb je een plugin gevonden die je wilt gebruiken? Dan klik je op '**Nu installeren**' en daarna op '**Activeren**'. Vaak kun je daarna de instellingen van de plugin aanpassen in het WordPress-menu.

Opdracht – Plugin toevoegen

1. Log in op je WordPress-dashboard.
2. Ga naar **Plugins** → **Nieuwe plugin**.
3. Zoek in de bibliotheek naar een plugin met handige functies, bijvoorbeeld:
 - Een formulierplugin zoals **WPForms**
 - Een beveiligingsplugin zoals **Wordfence**
 - Een galerijplugin voor afbeeldingen zoals **NextGEN Gallery**
4. Klik op **Installeren** en daarna op **Activeren**.
5. Ga naar de instellingen van de plugin en bekijk wat je kunt aanpassen.

☐☐ Controlevragen

- Wat is het verschil tussen een thema en een plugin?
- Welke plugin heb jij gekozen en waarom?
- Wat heb je aangepast of ingesteld binnen deze plugin?

☐☐ Inleveren

- Maak een screenshot van jouw WordPress-dashboard waarop te zien is dat de plugin actief is.
- Lever ook een document in (.txt of .pdf) met jouw antwoorden op de drie reflectievragen.

6 *Plugin installeren in WordPress*

☐☐ Leerdoelen

- Je weet hoe je een plugin installeert en activeert in WordPress.
- Je kunt verschillende soorten plugins herkennen en benoemen.
- Je begrijpt dat je meerdere plugins tegelijk actief kunt hebben.

☐☐ Uitleg

In WordPress kun je via het menu **Plugins** plugins toevoegen, verwijderen of activeren. Plugins zijn uitbreidingen die extra functies aan je site toevoegen, zoals formulieren, een webshop, beveiliging of een fotoalbum.

Zo kom je bij de plugin-instellingen:

1. Log in op je WordPress-dashboard.
2. Klik in het linker menu op **Plugins**.

Je komt dan op een pagina waar je een overzicht ziet van alle **geïnstalleerde plugins**. Sommige plugins zijn actief (ingeschakeld), anderen niet. In tegenstelling tot thema's kun je meerdere plugins tegelijk actief hebben. Meestal gebruik je er zelfs behoorlijk wat.

Standaard zijn er al een of twee plugins geïnstalleerd (zoals Akismet of Hello Dolly), maar die kun je voorlopig negeren.

📁 Een nieuwe plugin zoeken

Klik op **'Nieuwe plugin'** om naar de plugin-bibliotheek te gaan. Dit is het zoekscherm waar je duizenden plugins kunt doorzoeken en installeren.

Je kunt zoeken op naam, functie of populair/gewaardeerd. Net als in de App Store kun je zien hoeveel sterren een plugin heeft en hoe vaak deze is geïnstalleerd.

📁 Voorbeelden van soorten plugins:

- 📁 Plugins voor SEO (zoekmachineoptimalisatie)
- 📁 Plugins voor beveiliging tegen hackers en spam
- 📁 Plugins om een webwinkel te maken (zoals WooCommerce)
- 📁 Plugins voor fotoalbums en mediagalleries
- 📁 Plugins voor contactformulieren
- 📁 Plugins om je site sneller te maken (bijv. caching, afbeeldingen optimaliseren)
- 📁 Plugins om bezoekersstatistieken bij te houden
- 📁 Plugins voor automatische back-ups van je website
- 📁 Plugins voor sociale media (Snapchat, Instagram, Facebook, enz.)

De meeste plugins die je vindt in deze bibliotheek zijn gratis te gebruiken. Voor sommige geavanceerde functies moet je betalen voor een 'Pro'-versie.

📁 Opdracht – Installeer en activeer een plugin

1. Log in op je WordPress-dashboard.
2. Ga naar **Plugins** → **Nieuwe plugin**.
3. Zoek naar een plugin die jou handig lijkt, bijvoorbeeld:

- **WPForms** – om formulieren toe te voegen
- **Wordfence** – voor extra beveiliging
- **Yoast SEO** – om beter gevonden te worden in Google

4. Klik op **Installeren** en daarna op **Activeren**.

5. Ga na het activeren naar de instellingen van de plugin en bekijk wat je kunt aanpassen.

☐☐ Controlevragen

- Welke plugin heb je gekozen en waarom?
- Wat voor functies voegt deze plugin toe aan jouw site?
- Heb je iets aangepast in de instellingen van de plugin?

☐☐ Inleveren

- Maak een screenshot van jouw pluginpagina waarop je plugin zichtbaar en geactiveerd is.
- Noem het bestand: `plugin-.jpg`
- Lever ook een kort document (.txt of .pdf) in met je antwoorden op de drie controlevragen.

7 Het installeren van een plugin – Classic Editor

☐☐ Leerdoelen

- Je weet hoe je de editor van WordPress kunt aanpassen met een plugin.
- Je kunt de Classic Editor installeren en activeren.
- Je kunt verschillen benoemen tussen de standaardeditor (Gutenberg) en de Classic Editor.

Uitleg

Als je pagina's maakt in WordPress, gebruik je standaard de **Gutenberg-editor**. Deze editor werkt met blokken: elk stukje tekst, afbeelding of knop is een apart blok dat je kunt verslepen of aanpassen.

Sommige gebruikers vinden deze editor handig, anderen vinden het juist onoverzichtelijk. Gelukkig kun je dit aanpassen! Je kunt namelijk via een plugin overstappen op de oude **Classic Editor**.

Classic Editor is een plugin die de oude WordPress-editor terugbrengt. Deze werkt meer zoals Microsoft Word: je hebt een eenvoudige werkbalk bovenaan en typt alles in één veld.

Opdracht – Classic Editor installeren en vergelijken

1. Ga naar de **voorkant van je site** (de homepage).
2. Klik bovenin op **'Pagina bewerken'**.
3. Kijk goed naar hoe de editor eruitziet. Voeg eventueel wat tekst toe.
4. Ga in de backend naar **Plugins** → **Nieuwe plugin**.
5. Zoek op **'Classic Editor'**, installeer en activeer de plugin.
6. Ga opnieuw naar je homepage en klik weer op **'Pagina bewerken'**.
7. Je ziet nu een andere editor. Pas iets aan en klik op **Bijwerken**.

Je kunt op elk moment terugschakelen door de plugin **uit** te zetten via de pluginpagina.

Controlevragen – Vergelijk de editors

- Beschrijf minimaal drie verschillen tussen de standaardeditor en de Classic Editor.
- Welke editor heeft jouw voorkeur? Waarom?

Inleveren

- Lever een korte tekst (.txt of .pdf) in waarin je de drie verschillen beschrijft en aangeeft welke editor jij fijner vindt.

- Maak een screenshot van de Classic Editor in je browser.

Bestandsnaam:

□ Bonusopdracht (2 extra punten)

1. Zoek in het plugin-overzicht naar een andere editor (bijvoorbeeld **Elementor** of **WPBakery Page Builder**).
2. Installeer en activeer deze nieuwe editor. Zet de 'Classic Editor' plugin eerst uit.
3. Probeer deze nieuwe editor uit door een pagina aan te passen.

□□ Bonus Reflectie

- Wat zijn de voor- en nadelen van deze nieuwe editor?
- Welke editor is nu je favoriet en waarom?

8 *Formulieren maken met WPForms*

□□ Leerdoelen

- Je weet hoe je WPForms installeert en activeert.
- Je kunt een formulier maken met verschillende soorten invoervelden.
- Je kunt een formulier invoegen op een pagina en opnemen in het menu.

□□ Uitleg

Een formulier op je website is handig om bezoekers contact met je op te laten nemen. Je kunt hiermee gegevens verzamelen, vragen laten stellen, of zelfs aanmeldingen verwerken.

Met de plugin **WPForms** kun je eenvoudig formulieren bouwen via een visuele interface. Je hoeft niet te programmeren: je sleept de velden naar de juiste plek.

Zo installeer je WPForms:

1. Ga in de backend naar **Plugins** → **Nieuwe plugin**.
2. Zoek naar **WPForms**.
3. Klik op **Installeren** en daarna op **Activeren**.

Na het activeren verschijnt er een nieuwe optie **WPForms** in het linker menu. Ook krijg je een introductiescherm met een uitlegvideo. Bekijk deze video voor je verder gaat.

Opdracht – Maak een contactformulier

1. Ga naar **WPForms** → **Nieuwe toevoegen**.
2. Kies **Leeg formulier** of begin met een **Contactformulier-template**.
3. Voeg de volgende velden toe aan je formulier:
 - **Voornaam** (verplicht)
 - **Achternaam** (verplicht)
 - **Straatnaam**
 - **Huisnummer**
 - **Emailadres** (verplicht)
 - **Telefoonnummer** (verplicht)
 - **Hoe mogen we contact met u opnemen?**
Meerkeuze: Per telefoon / Per email
 - **Wanneer mogen we contact opnemen?**
Dropdown: 's ochtends / 's middags / 's avonds
 - **Vraag die u wilt stellen**
Tekstvak (verplicht)
4. Sla het formulier op. Kopieer de **shortcode** die je ziet (bijv. `[wpforms id="123"]`).
5. Maak een nieuwe pagina aan met de titel **Contact**.
6. Plak de shortcode in deze pagina en sla op.
7. Test de pagina en controleer of je formulier correct wordt weergegeven.
8. Ga naar **Weergave** → **Menu's** en voeg je nieuwe **Contactpagina** toe aan het hoofdmenu.

Controlevragen

- Welke soorten invoervelden heb je gebruikt?
- Wat vond je van het werken met WPForms?
- Wanneer zou je dit soort formulieren in het echt kunnen gebruiken?

☐☐ Inleveren

- Maak een screenshot van jouw **Contactpagina** met het formulier zichtbaar en de URL van de pagina in beeld.
- Noem het bestand: `contactformulier-<jouw-naam>.jpg`
- Lever ook een document in (.txt of .pdf) waarin je de drie controlevragen hierboven beantwoordt.

9 Project: WordPress-site voor Vakantiepark 'De Eekhoorn'

☐☐ Leerdoelen

- Je kunt zelfstandig een WordPress-site installeren op een eigen serveromgeving.
- Je kunt een site ontwerpen met een passend thema en aangepaste huisstijl.
- Je kunt verschillende soorten plugins gebruiken voor formulieren, galerijen, SEO en veiligheid.
- Je kunt een complete website opleveren op basis van wensen van een opdrachtgever.

☐☐ Projectbeschrijving

Je bent net opgeleid tot WordPress-ontwikkelaar en krijgt meteen je eerste opdracht: het bouwen van een website voor het Drentse vakantiepark **De Eekhoorn**. De site wordt geïnstalleerd op jouw eigen PLESK-serveromgeving (gebruik hiervoor een aparte map: `vakantiewoning` en maak een nieuwe database aan).

☐☐ Te bouwen pagina's:

- **Homepage** – algemene info over het vakantiepark
- **Reserveringsaanvraagpagina** – met een formulier voor boekingsverzoeken
- **Informatiepagina** – tekst en afbeeldingen over het park
- **Type woningen** – drie woningtypes met beschrijving en fotogalerij of -slider

☐☐ Projecteisen

☐☐ Design

- Gebruik een gratis **thema uit de WordPress-theme store** dat responsive is.
- Pas dit thema aan via de **Customizer**.
- Gebruik de kleuren van de Drentse vlag: **rood, wit, zwart**.
- Zorg dat het **logo op alle schermgroottes zichtbaar blijft**.

☐☐ Reserveringsformulier (op aparte pagina)

- Naam (verplicht)
- Adresvelden
- Emailadres (verplicht)
- Telefoonnummer (verplicht)
- Gewenste woning (selectbox)
- Aankomstdag (tekstveld, verplicht)
- Aantal weken (min 1 – max 4)
- Aanvullende opmerkingen

Kies een gratis **formulierenplugin** uit de WordPress-plugin store (bijv. WPForms). Zorg dat het formulier werkt en verzonden wordt naar jouw studenten-email.

☐☐ Foto's en woningtypes

- Gebruik een **gallery- of sliderplugin** om de drie woningtypes aantrekkelijk te presenteren.
- Zorg voor een goede opmaak en duidelijke beschrijvingen.

☐ Veiligheid

- Installeer een gratis **beveiligingsplugin** (bijv. Wordfence of All In One WP Security).

☐ SEO

- Installeer **Yoast SEO (gratis versie)** voor zoekmachineoptimalisatie.

☐ Navigatie en menu's

- Zorg voor een **duidelijk navigatiemenu** dat goed werkt op mobiel en desktop.
- Alle pagina's moeten via het menu bereikbaar zijn.

☐ Gebruik klantmateriaal

- Gebruik de **teksten en afbeeldingen** die door de klant zijn aangeleverd.
- [Download hier het materiaal](#)

☐ Inleveren

- Controleer of alle pagina's, plugins en vormgevingseisen zijn uitgevoerd.
- Lever de **url** van jouw site in.
- Vraag een **eindgesprek** aan waarin je jouw site presenteert.
- In het gesprek vertel je:
 - Welke plugins je hebt geïnstalleerd
 - Hoe je de site hebt gestyled
 - Hoe je met het klantmateriaal hebt gewerkt

☐ Tip: Lees de opdracht nog één keer goed door voordat je de site inlevert.

Yii Introduction

1 Intro - Wat is Yii?

In deze les gaan we leren wat een framework is en waarom we Yii gaan leren.

We leren verder wat MVC en CRUD is en wat heeft dat te maken heeft met een framework.

Voor het examen moet je een webapplicatie bouwen.

Je moet zelf weten hoe je dat doet. De meeste studenten kiezen voor PHP. Waarom? Het is relatief eenvoudig en de meeste web applicaties zijn ontwikkeld met PHP ([bron](#)).

Veel websites lijken op elkaar en doen vaak dezelfde dingen. Ze maken een verbinding naar de database, laten gegevens zien, je kunt aanloggen, je kunt soms gegevens veranderen en ga zo maar door.

Om niet telkens al deze 'standaard' zaken van 'scratch' af aan te moeten bouwen zijn er zogenaamde frameworks ontwikkeld. Elk framework is eigenlijk een verzameling bouwblokken die je telkens op nieuw kunt gebruiken voor jouw website. Alle frameworks zijn net een beetje anders, maar er zijn ook veel overeenkomsten.

Wat zijn belangrijke zaken als je voor een framework kiest:

1. Wordt het framework actief onderhouden; bestaat het al wat langer? Je wilt niet in iets gaan investeren dat volgend jaar misschien niet meer bestaat.
2. Hoe moeilijk is het om een framework te leren; wat is de complexiteit?
3. Hoe snel kun je iets ontwikkelen met het framework?
4. Hoe flexibel is een framework; kun je er echt alles mee maken?

Bij punt 3 en 4 is het vaak het een of het ander. Of je kunt heel snel iets bouwen en je bent niet zo flexibel, of je kunt er eigenlijk alles mee ontwikkelen maar het kost veel tijd om dat te doen.

Bijvoorbeeld: PHP is super flexibel je kunt er echt alles mee ontwikkelen, het Laravel framework is ook redelijk flexibel, het Yii framework is iets minder flexibel en als je naar bijvoorbeeld WordPress kijkt (wat je ook als een soort framework zou kunnen zien), dan is dat het minst flexibel. Aan de andere kant kun je met WordPress heel snel een site maken, je bent alleen een beetje beperkt in de functionaliteiten.

❑ Waarom Yii?

Laravel is het meest gebruikte PHP-framework. Yii staat in de meeste lijstjes ongeveer op plaats 7. Dat komt, omdat je weliswaar "snel even wat kan bouwen", maar je kunt het vrij lastig alles precies zo maken als jij het wilt. Het is minder flexibel en om het precies zo te krijgen wilt dan moet je meer moeite doen. Als je straks een project maakt dan kun je waarschijnlijk ongeveer 70%-80% van je code maken met Yii, daarvoor hoef je heel weinig te programmeren en heb je tijd om je te storten op de overige 20%-30%. De projecten en examens die jullie gaan doen lenen zich goed om met Yii te maken. Je bent namelijk redelijk vrij in het ontwerp en kunt dus de 'Yii-manier' aanhouden.

Maar wat dan als ik later verder wil met (bijvoorbeeld) Laravel? Dat kan en Yii en Laravel hebben dezelfde opzet. Ze werken beiden volgens het [MVC principe](#). Je zult in Laravel dus veel dingen terugvinden die je in Yii hebt geleerd.

MVC

MVC, wat is dat? MVC staat voor **Model**, **View** en **Controller**. Deze drie-deling wordt gemaakt in je code, zodat je eenvoudiger code kunt terugvinden.

Model	Hierin staat waar de verbinding tussen jouw code en de database beschreven
View	Dit is de front-end en is vooral HTML/CSS (bootstrap) met hier een daar een 'vleugje' PHP
Control	Dit is waar je bepaald wat er gebeurt, als je op een knop drukt ergens dan wordt de code uit de bijbehorende control uitgevoerd.

In een plaatje ziet dat er als volgt uit. De gebruiker doet wat en stuurt daarmee een verzoek (request) naar de juiste controller. De controller 'overlegt' met het model als er iets met data moet gebeuren en het resultaat wordt via de view aan de gebruiker getoond.

image-1594413597727.png

In de introductie van deze module (workshop) komen deze begrippen terug. Als het nu allemaal nog wat onduidelijk is, geen probleem we gaan er mee aan de slag en zullen snel een gevoel krijgen wat MVC precies is en hoe het werkt.

<https://www.youtube.com/watch?v=FCkDEHWDATI>

<https://www.youtube.com/embed/FCkDEHWDATI>

CRUD?

Nog even....waar staat CRUD ook alweer voor? CRUD staat voor:

Create	 Nieuwe toevoegen
Read	 Bekijken / lezen
Update	 Bewerken
Delete	 Verwijderen

In een CRUD-applicatie kun je dus gegevens aanmaken (create), lezen (read), bewerken (update) en verwijderen (delete).

Voor jouw examen moet je minimaal een CRUD kunnen maken.

Yii heeft een CRUD-generator en je kan zonder te coderen een standaard CRUD-applicatie maken. Super handig dus voor het examen. In de volgende les gaan we hier gelijk mee aan de slag.

Quizvragen

1. **Wat is een framework en waarom zou je het gebruiken bij het maken van een webapplicatie?**

Leg in je eigen woorden uit wat een framework is en wat het voordeel is ten opzichte van alles zelf programmeren.

2. **Wat zijn volgens de tekst belangrijke vragen die je moet stellen voordat je kiest voor een bepaald framework?**

Noem minstens twee overwegingen en leg kort uit waarom die belangrijk zijn.

3. **Wat is volgens de tekst het verschil tussen Laravel en Yii?**

Leg uit in welk opzicht Yii sneller is dan Laravel, maar ook waar je tegenaan zou kunnen lopen.

4. **Wat betekenen de letters M, V en C in MVC**

Beschrijf per onderdeel kort de rol die het speelt binnen een webapplicatie.

5. **Hoe werken Model, View en Controller samen?**

Beschrijf het proces dat plaatsvindt als een gebruiker op een knop drukt.

6. **Wat betekent CRUD en wat moet je hierover kunnen voor het examen?**

Noem alle vier onderdelen van CRUD en leg uit waarom dit belangrijk is voor je examenproject.

7. **Waarom is Yii een goede keuze voor het examenproject?**

Noem minstens twee redenen waarom Yii goed past bij de opdrachten die je moet maken.

☐ Inleveren

- Beantwoord de vragen en lever dit in als .txt of .pdf-bestand.
(Het op dit is een goede voorbereiding op de kennis-check)

2 Installation Yii

In de komende paar opdrachten gaan we alles klaarmaken en installeren voor het werken met Yii. Aan het eind van de opdracht 4 heb je volledig werkende Yii CRUD-applicatie.

Een CRUD-applicatie is een applicatie waarmee je records kunt aanmaken (Create), records kunt lezen (Read), records kunt aanpassen (Update) en records kunt verwijderen (Delete).

We gaan drie stappen doen om een werkende web app te maken:

1. installeren van alle software
2. maken van een eenvoudige database
3. generen van code

In drie stappen hebben we straks een *werkende* web app in PHP gemaakt.

Voordat je met deze les begint weet je hoe je met [phpMyAdmin](#) (of een andere tool) een database kunt maken en hoe je een .sql file kan importeren. Ook heb je XAMPP geïnstalleerd.

☐ Installatie

De installatie kent een aantal stappen. Hieronder zijn de 5 stappen beschreven die je moet uitvoeren om tot een succesvolle Yii installatie te komen.

☐ stap 1 - XAMPP

Je hebt [XAMPP](#) geïnstalleerd. Voor Yii hebben dit nodig, omdat we een database gaan gebruiken.

Heb je nog geen XAMPP of werkt die niet meer dan kun je [hier](#) lezen hoe je XAMPP kan installeren.

☐☐ stap 2 -Composer

Zorg er voor dat je [composer](#) hebt geïnstalleerd. Composer is de installer van de WEB tools.

Uitleg, wat is Composer?

Als we een framework als Yii of Laravel gaan gebruiken dan installeren we eigenlijk een hele grote doos met allemaal bouwblokken. Deze blokken moeten allemaal samen werken en zijn vaak afhankelijk van elkaar. De blokken worden door verschillende mensen ontwikkeld en op n of andere manier moet er voor worden gezorgd dat de juiste blokken zijn geïnstalleerd. Hiervoor dient Composer. Composer is als het ware de installer van PHP tools, deze worden vaak libraries of packages genoemd. Libraries zijn dan de blokken functionaliteit die je kunt gebruiken om een programma te maken.

<https://getcomposer.org/>

Let op: check goed of de installatie errors/warnings geeft. Zijn er errors, lees deze dan goed en los de issues met behulp van ChatGPT op!

Tip: Zorg dat XAMPP al is geïnstalleerd voordat je composer installeert; composer heeft namelijk de locatie van php.exe nodig.

In deze korte video wordt getoond hoe je Composer kunt installeren:

<https://www.youtube.com/embed/Rvb6DQ5PYHM>

☐☐ stap 3 - Create new yii Project

We maken het Yii project **world** aan.


Je hoeft niet zelf een map/folder aan te maken. Dit doet composer voor jou. Zorg ervoor dat je in de terminal in de parent-folder (in de folder/map waar je nu staat) bent.

Vul de volgende code in de terminal:

```
composer create-project --prefer-dist yiisoft/yii2-app-basic world
```

Dit commando maakt een folder aan met de naam world. Dit is je nieuwe Yii project folder (er staan honderden bestanden in).

⚠️ Error, warning ?

 image.png Image not found or type unknown	<p>Let op, check of je errors en of warnings krijgt! Zie je die, probeer die dan met behulp van ChatGPT op te lossen!</p> <p>Extra uitleg: bij sommige installaties van XAMPP staan bepaalde extensies niet aan. Zo staat bijvoorbeeld de ZIP extensie soms uit in de PHP config (XAMPP, php.ini). Vraag chatgpt om dit op te lossen.</p> <p>Als dit zo is dan kan de Yii installatie fout gaan. Check daarom de output en pas de configuratie met behulp van ChatGPT aan.</p> <p>Als je opnieuw wilt beginnen moet je eerst je oude foutieve installatie verwijderen (in dit geval de map "world"). Doe dit vanuit de file explorer.</p>
---	---

Let op dat het nieuwe Yii project wordt gemaakt in de folder waarin je de cmd box opent. Stel je wilt je jouw nieuwe project op je Desktop maken dan open je een cmd box en je tikt het commando `cd Desktop` in. Voer dan het composer commando uit (zoals hierboven aangegeven) en het nieuwe Yii project zal op je Desktop worden gemaakt.

📌 stap 4 - Start Development Server

yii heeft een ingebouwde php server als je die opstart dan draait jouw website op

<http://localhost:8080/>

Let wel dat je XAMPP ook blijft draaien voor je database.

In VSC open je jouw Yii project en daarna open je een nieuwe terminal.


image.1618344180761.png
Image not found or type unknown

In de terminal type je dan het commando in:

```
php yii serve
```

De Yii webserver draait nu. Ga naar je browser en controleer of je via localhost:8080 jouw nieuwe Yii project ziet.

Wil je cmd box gebruiken of lukt dit niet. Kijk dan naar de [instructiefilm \(Koppelingen naar een externe site.\)](#)

<https://www.youtube.com/embed/5idkjTWIL7g>

☐☐ stap 5 - Configure web.php

We moeten nu nog het een en ander configureren in Yii. We willen dat Yii zoveel mogelijk voor ons regelt. Om te beginnen mag Yii de routing regelen (dit wordt uitgelegd in de [volgende les](#)).

Open **config/web.php** - en zet het gedeelte dat urlmanager heet, uit commentaar (uncomment prettyURL). Het moet er als volgt uitzien (ongeveer op regel 50 van config/web.php).

```
'urlManager' => [  
    'enablePrettyUrl' => true,  
    'showScriptName' => false,  
    'rules' => [  
    ],  
],
```

Dit is een instelling die de [routing](#) (waarover later meer) eenvoudiger maakt.

☐☐ stap 6 - Database

Let er op dat je **database goed is gedefinieerd** en dat je ook de relaties heb vastgelegd. Dat kun je doen met [phpMyAdmin](#) . De ingebouwde [Yii CRUD-generator](#) werkt alleen goed als de database en de relaties goed zijn vastgelegd.

Maak de database world en importeer deze file: <https://www.roc.ovh/attachments/22>

Lukt niet of weet je niet precies hoe je een database moet importeren? In [deze les](#) staat beschreven hoe je deze database kunt installeren. In deze les staat de [SQL-file world.sql](#) waarmee je de database World kan maken.

We gaan Yii vertellen welke database er moet worden gebruikt.

We openen de file **config/db.php** en zetten daar het volgende in:

```
return [  
    'class' => 'yii\db\Connection',  
    'dsn' => 'mysql:host=127.0.0.1;dbname=world',  
    'username' => 'root',  
    'password' => '',  
    'charset' => 'utf8',  
];
```

Let op dat je geen localhost gebruikt om naar je database te verwijzen. Gebruik 127.0.0.1.

☐ stap 7- Klaar?

Klaar? Dan moeten we nog even testen of alles werkt.

Je had al gezien dat je een leeg Yii project hebt, ga nu naar: <http://localhost:8080/gii>

Jouw scherm ziet er ongeveer zo uit:

Screenshot 2022-05-20 183633-1.png

Maak een screen shot van jouw *hele* scherm met browser en lever deze in.

☐ Inleveren

1. Screenshot van Yii waarin je <http://localhost:8080/gii> laat zien. Dit is de Model en Crud generator die we in de volgende stap gaan gebruiken.

3 *CRUD aanmaken*

☐ Leerdoelen

- Je kunt met de Gii CRUD Generator een complete CRUD maken voor een databasetabel.
- Je begrijpt dat je eerst een model moet genereren voordat je een CRUD kunt maken.

☐ CRUD

We gaan een CRUD maken en met Yii is dat heel makkelijk.

We moeten eerst een verbinding met de database maken. Weet je nog welke letter van MVC daar ook alweer voor stond?

Yep, de M van Model. Het model beschrijft hoe jouw Yii applicatie een verbinding met de database kan maken. We gaan dus een Model maken.

☐ Stap 1 - models maken

We gaan opnieuw naar de component builder, open <http://localhost:8080/gii/model>

Klik links op 'Model Generator' en type de naam van de databasetabel **country** in:

image-1594285274135.png

De naam van het model wordt automatisch aangemaakt **Country** . Let op dat de 'Model Class Name' met een **hoofdletter** begint; kijk goed naar het voorbeeld. De naam van een class (oop) wordt altijd met een hoofdletter geschreven.

Druk onderaan op de pagina op *preview* , Yii laat nu zien welke file hij gaan aanmaken voor je. Druk op **generate** om de file te maken.

De file **model/country** is nu aangemaakt, we gaan later meer in op de inhoud van deze file. Wat voor nu belangrijk is om te onthouden is dat de model/country.php file de verbinding is tussen de tabel country en Yii. Elke keer als Yii informatie uit de country tabel nodig heeft dan gebruikt Yii de model/country.php file

Omdat er vanuit de country tabel wordt verwezen naar de andere twee tabellen, **City** en **Countrylanguage** maak je nog twee modellen van deze twee tabellen.

☐ **Controleer de model folder en als het klopt (let op de hoofdletters!) ga dan niet door.**

In je **model** directory (folder) heb je nu dus de volgende files staan

```
City.php
ContactForm.php
Country.php
Countrylanguage.php
LoginForm.php
User.php
```

☐ Stap 2 - CRUD maken

We gaan de complete CRUD maken.

Ga terug naar de component [builder](#).

Klik links op 'CRUD Generator' en type het volgende in:

image-1618227641051.png

Let goed op de **hoofdletters**, deze zijn belangrijk.

Het View path wordt leeg gelaten, daar kan Yii zelf iets voor kiezen.

Druk op preview, je ziet 8 files die klaar staan om te worden aangemaakt.

image-1618227971123.png

Druk op **generate** om de files aan te maken.

Controller

In de directory controllers staat een file CountryController.php. Dit is de controller van de country CRUD. Al het denkwerk dat nodig is om de country CRUD aan te sturen zit in de CountryController.php file.

View

In de views/country directory staan de views. De index.php is de standaard view en deze file bevat de (voornamelijk HTML code) om de pagina weer te geven.

Weten we nog waar de controller en view ook alweer vandaan komen? Het Framework is gebouwd volgens de MVC-architectuur. In stap 1 hebben we het model gemaakt en in deze 2de stap hebben we de view en controller gemaakt.

image-1612557541726.png

De controller is waar alle verzoeken van de gebruiker naar toe gaan. De view is de presentatielaag met voornamelijk HTML, CSS, en JavaScript. En in het model wordt de verbinding met de database gemaakt.

Klaar ?

Ga naar <http://localhost:8080/country>

image-1594288457454.png

Je hebt een CRUD van Country gemaakt, compleet met search en sort-functionaliteiten. Je hebt ook een mooie pagina selector waarmee je daar de verschillende pagina's kan navigeren. Je kunt country's lezen/tonen, aanpassen, verwijderen en aanpassen. Probeer het maar! Wees niet bang om de database 'kapot' te maken, want je kunt in een paar tellen de database opnieuw importeren.

In de volgende lessen gaan we de automatisch gebouwde applicatie stap-voor-stap aanpassen.

Voor nu kan je trots zijn op je eerste (?) web CRUD applicatie.

Extra hulp met Filmpje

Niet gelukt? Kijk in dit filmpje nog een keer alle stappen door: [Instructie Yii - les1](#)

<https://www.youtube.com/embed/6NRMaJgSmLw>

□□ Uitleg

Je hebt al een CRUD gemaakt voor de tabel `country`. Nu ga je hetzelfde doen voor de tabellen `city` en `countrylanguage`.

Denk eraan: je moet eerst de modellen genereren voor deze twee tabellen met de Model Generator, voordat je de CRUD maakt.

□□ Opdracht

1. Open de **Model Generator** op <http://localhost:8080/gii/model>
2. Maak het model voor de tabel `city`.
3. Maak het model voor de tabel `countrylanguage`.
4. Open daarna de **CRUD Generator** op <http://localhost:8080/gii/crud>
5. Genereer een CRUD voor `City`.
6. Genereer een CRUD voor `Countrylanguage`.

Let goed op de hoofdletters bij het invullen van de modelnamen!

Controleer op

1. <http://localhost:8080/city>
2. <http://localhost:8080/countrylanguage>

of de CRUD werkt.

GIT

"Meester, gisteren werkje het nog en ik heb niets gedaan?"

Bij Yii loop je soms vast. Je maakt wijzigen in een door het framework gemaakte files. Soms werkt er opeens *niets* meer (echt dat gebeurt iedereen!).

Met GIT kun je ervoor zorgen dat je steeds een werkende versie opslaat. Werkt het dan niet meer, dan kun je terug naar de vorige werkende versie.

Wert GIT nog niet in jouw VCS? [Hier](#) staat hoe je GIT in VCS kunt 'aanzetten'.

☐☐ Reflectie

- Waarom moet je eerst een model aanmaken voordat je een CRUD kunt genereren?
- In welke folder(s) worden er bestanden aangemaakt door de model-generator?
- In welke folder(s) worden er bestanden aangemaakt door de CRUD-generator?

☐☐ Inleveren

- Maak een screenshot van `http://localhost:8080/city` waarop de CRUD werkt.
- Maak een screenshot van `http://localhost:8080/countrylanguage` waarop de CRUD werkt.

4 View aanpassen

☐☐ Opdracht - deel 1

Zorg eerst dat, nu alles nog werkt, je met GIT een commit uitvoert. Je kan dan altijd terug als je wat 'kappot' maakt.

Ga in de code-editor (VSC) naar de map **views** en probeer te vinden waar het overzicht **Countries** is gemaakt.

Haal code weg zodat de kolom *Surface Area* niet meer wordt getoond.

Probeer maar wat, als je de boel 'kappot' maakt aan ga je met behulp van GIT naar de vorige versie.

image-1612558210976.png

Opdracht - deel 2

Kun je de in dezelfde view de groene knop 'Create Country' verplaatsen en onderaan de pagina plaatsen?

Dus het moet er zo uit te komen zien:

image-1612558884647.png

Resultaat

Jouw *country* overzicht ziet er dus zo uit:

image.png

Inleveren

1. Lever een compleet screenshot in van de **country** read page (Grid View; zoals plaatsje hierboven met het rode kruis).
2. Lever een compleet screenshot in van de read page (Grid View) van **city**
3. Lever een compleet screenshot in van de read page (Grid View) van **countrylanguage**
4. Lever het (aangepaste) bestand *views/country/index.php* in - noem dit bestand jouw-naam-index.php

5 Routing in Yii

Leerdoelen

- Je weet wat routing is in het algemeen.
- Je begrijpt hoe routing werkt binnen Yii.
- Je kunt URL's analyseren en begrijpen hoe ze naar controllers en acties verwijzen.

Uitleg

Wat is routing?

Routing is de manier waarop een webserver en je programma bepalen welke pagina moet worden. Kijk maar eens goed naar de URL in je browser: deze bestaat uit verschillende onderdelen.

image-1594292626981.png

Onderdeel	Wat is het?	Uitleg
Scheme	Protocol	Bepaalt hoe de informatie wordt verzonden (zoals http, https, ftp...)
Domeinnaam	Domein	Verwijst naar een server of groep servers.
Path	Bestandspad	Verwijst naar het bestand dat de webserver moet uitvoeren.
Parameters	Variabelen	Sturen informatie mee naar de server (zoals bij formulieren).
Anchor	Paginaanker	Springt naar een specifiek onderdeel binnen een pagina.

Voorbeeld-URL: `https://www.roc.ovh/link/302#bkmrk-nog-een-keer-in-sche`

Bij een simpele PHP-site zonder framework komt het pad overeen met de bestandslocatie op je computer.

`https://localhost/ opgave1/uitslag/index.php`

Dan wordt de file

`c:\xampp\htdocs\opgave1\uitslag\index.ph`

Wat is de document-root van de webserver?

De **document root** van de webserver is waar de webserver al zijn bestanden kan vinden en waar die heen gaat als je alleen de hostname gebruikt.

c:\xampp\htdocs wordt ook wel de document root van de webserver genoemd omdat:

https://localhost verwijst naar **c:\xampp\htdocs** (in een standaard geïnstalleerde XAMPP omgeving).

Routing bij frameworks zoals Yii

Frameworks zoals Yii gebruiken een ander soort routing. De URL wordt niet direct vertaald naar een bestand, maar naar een **controller** en een **actie** (methode).

Voorbeeld:

http://localhost:8080/country/index

De URL hierboven verwijst naar:

De controller	country in → CountryController in controllers/
De actie	index in → methode <code>actionIndex()</code> binnen die controller

De standaardregels in Yii:

- Het **eerste deel** van het pad wordt omgezet naar een **controller**. Elk woord begint met een hoofdletter. Een streepje (-) betekent een nieuw woord.
- Het **tweede deel** van het pad wordt een **methode (functie in de controller)**, voorafgegaan door `action`, en met dezelfde schrijfwijze als hierboven.

Voorbeelden

Pad	Controller	Actie
/kaart/ overzicht	KaartController	actionOverzicht()
/klas-lokaal/ overzicht	KlasLokaalController	actionOverzicht()
/klas-lokaal/ stoel-maat	KlasLokaalController	actionStoelMaat()
/klas-lokaal/ index	KlasLokaalController	actionIndex()
/klas-lokaal	KlasLokaalController	actionIndex()

Als het tweede deel ontbreekt, wordt automatisch `actionIndex()` gebruikt.

Opdracht – Hello World!

We gaan in onze World-applicatie oefenen met routing.

1. Open `CountryController.php` in de map `controllers`.
2. Voeg daarin de volgende methode toe:

```
public function actionHello() {  
    echo "Hello World!";  
    exit;  
}
```

3. Sla het bestand op.
4. Bedenk: welke URL moet je nu gebruiken om deze functie aan te roepen?
5. Open je browser en test de URL. Als het goed is zie je:

image-1594296881002.png

Controle

- Is de tekst `Hello World!` zichtbaar in je browser?

Inleveren

- Maak een screenshot van je browser waarop **Hello World!** zichtbaar is en waarin tevens de URL zichtbaar is.

Routing met een parameter

We gaan een nieuwe controller maken en oefenen met routing waarbij een **parameter** wordt doorgegeven via de URL.

1. Maak een nieuwe file aan in de map `controllers` met de naam `ExampleController.php`.
2. Plaats hierin de volgende code, maar **vervang de naam Ayoub op beide plekken door jouw eigen naam**:

```
<?php

namespace app\controllers;

use Yii;
use yii\web\Controller;

/**
 * ExampleController - voorbeeldcode voor routing.
 * Code door [Jouw Naam]
 */
class ExampleController extends Controller
{
    public function actionSay($message = '')
    {
        echo "Hello $message";
        exit;
    }
}
```

3. Sla het bestand op.
4. Open je browser en ga naar de juiste URL om `actionSay()` uit te voeren. Je gebruikt hiervoor:

```
http://localhost:8080/example/say
```

Je ziet nu als het goed is: **Hello**

Kun je via de URL ook een **parameter** meegeven, zodat in plaats van "Hello" er bijvoorbeeld "Hello Max" staat?

Hint: voeg je naam toe aan de URL als volgt:

```
http://localhost:8080/example/say?message=Max
```

□ Controle

- Wordt de pagina `Hello [jouw naam]` getoond in de browser?

- Werkt de parameter in de URL zonder dat je iets hoeft te veranderen aan de code?

☐☐ Reflectie

- Als je jouw naam in de URL zet wordt deze in de variabele `$message` gezet, er staat immers `echo "Hell $message";`
Hoekom komt het dat jouw naam in de variable met de naam `$message` staat?

☐☐ Inleveren

- Het bestand: `controllers/CountryController.php`
- Een screenshot van je volledige scherm met browser waarop de URL en **Hello [jouw naam]** te zien is.
- Een antwoord op de reflectievraag (in txt of pdf).

06 Menu

☐☐ Leerdoelen

- Je begrijpt hoe het menu (navigatie) in Yii is opgebouwd.
- Je kunt menu-items koppelen aan routes (routing).
- Je kunt submenu's maken en parameters gebruiken in je menu-items.

☐☐ Uitleg

In deze les ga je routing zoals we dat in de vorige les hebben geleerd toepassen. Je maakt een menu en koppelt de menu-items aan routes in je applicatie.

☐☐ Views folder

In de map `views` vind je drie mappen:

- **country:** bevat de views die je met de CRUD-generator hebt aangemaakt.

- **layouts:** bevat de standaard lay-out (zoals `main.php`) van elke pagina.
- **site:** bevat standaardpagina's zoals "About" en "Contact".

Elke pagina in Yii gebruikt de layout `main.php`

In de `main.php` staat `<?= $content ?>` dit is kort voor `<?php echo $content ?>` . Het toont jouw pagina die in de variabele `$content` staat:

☐ Experimenteer

Verwijder *tijdelijk* de regel uit `main.php` en bekijk wat er gebeurt. Wat zie je (of juist niet)?

☐ Menu via Widget

In de layout-bestand `main.php` staat een stuk code dat het menu opbouwt met een zogenaamde **Widget**. Een widget is een soort herbruikbaar codeblok, vergelijkbaar met [WordPress](#) plugins.

Vervang de standaard **menucode** in `main.php` door deze aangepaste versie:

```
<?php
use yii\bootstrap5\Nav;
use yii\bootstrap5\NavBar;

NavBar::begin([
    'brandLabel' => Yii::$app->name,
    'brandUrl' => Yii::$app->homeUrl,
    'options' => [
        'class' => 'navbar-expand-md navbar-dark bg-dark fixed-top',
    ],
]);

echo Nav::widget([
    'options' => ['class' => 'navbar-nav navbar-right'],
    'items' => [
        [
            'label' => 'Country',
            'items' => [
                ['label' => 'Overzicht', 'url' => ['/country/index']],
                ['label' => 'Voeg toe', 'url' => ['/country/create']],
                ['label' => 'Europa', 'url' => ['/country/index', 'CountrySearch[Continent]' => 'Europe']]
            ],
        ],
    ],
]);
```

```

],
[
  'label' => 'City',
  'items' => [
    ['label' => 'Overzicht', 'url' => ['/city/index']],
    ['label' => 'Voeg toe', 'url' => ['/city/create']],
  ],
],
],
]);

NavBar::end();
?>

```

Zoals je ziet gebruiken we [associatieve arrays](#) zoals:

```
['label' => 'Europa', 'url' => ['/country/index', 'CountrySearch[Continent]' => 'Europe']]
```

Opdrachten

1. Vervang de standaard menucode in `main.php` door het bovenstaande voorbeeld.

2. Om een country toe te voegen moet in de controller `CountryController` de method/function `actionCreate()` worden uitgevoerd.
Bedenk hoe je de **routing** naar deze pagina opzet en pas het menu aan 'voeg toe' zodat deze is gekoppeld aan `actionCreate()` van de `CountryController`.
3. Voeg het menu-item **Asia** toe, dat alleen landen van Azië toont.
4. Voeg vervolgens voor elk werelddeel een apart menu-item toe: Asia, Africa, North America, South America, Oceania en Antarctica.
5. Voeg een nieuw hoofdmenu toe: **City** met twee items: **Overzicht** en **Voeg toe**.

Inleveren

- Lever het aangepaste bestand `views/layouts/main.php` in.

07 Gridview

In deze les leer je aanpassingen in de Gridview maken.

*Je leert hoe je kolommen wel of niet **zichtbaar** kan maken, hoe je de kolom **headers** kan aanpassen, hoe je de kolom **breedte** kan aanpassen, hoe je de **inhoud** van de kolom zelf kan aanpassen en als laatste leren hoe we een **link** kunnen maken van een kolom.*

Country overzicht aanpassen

Onze CRUD-generator (Gii) heeft voor ons een country overzicht gemaakt. Als je de CountryController.php opent dan zie je dat function (method) actionIndex de view index aanroept. Open deze index file. Deze staat in views/country.

Gridview Widget

Je ziet dat er op regel 23 een zogenaamde widget wordt aangeroepen.

Een Widget is een plug-in waarmee je functionaliteit aan Yii toevoegt. De Gridview Widget is misschien wel de meest uitgebreide widget en het zorgt ervoor dat we een mooi overzicht van onze data krijgen.

Opdracht

De opdracht bestaat uit 8 stappen. In deze stappen gaan we de country view aanpassen (views/country/index.php).

Test na elke stap of de aanpassingen werken!

Stap 1 - Kolommen aanpassen

In de parameters van deze widget zie je dat de kolommen *Code* , *Name* , *Continent* , *Region* en *SurfaceArea* worden afgedrukt. Je ziet ook dat bepaalde kolommen niet worden afgedrukt, deze staan in commentaar.

- Verander de lay-out in het country overzicht en laat de volgende kolommen in deze volgorde zien:
Code , *Name* , *Population* , *Surface Area* en *Capital* .
Population laat inwoneraantal zien en *SurfaceArea* laat oppervlakte zien.

Test of het werkt!

Stap 2 - Kopjes aanpassen

Stel we willen het kopje boven de kolom veranderen, dan kan dan als volgt:
Vervang bijvoorbeeld 'SurfaceArea' in de GridView door het volgende array.

```
[ 'label' => 'Oppervlakte',  
  'attribute' => 'SurfaceArea',  
],
```

Je ziet dat we de surface area nu hebben vervangen door een associative array. Dus waar in de code het woord 'SurfaceArea' staat plaats je het associative array zoals hierboven is aangegeven.

Dit verteld de GridView snippet dat het label Oppervlakte moet zijn (dit wordt boven aan de pagina gezet) en dat de waarde in de kolom moet worden gevuld met 'SurfaceArea'.

- Pas nu de kolom *Population* aan; en zet in de kolom header 'Inwoners'.

Test of het werkt!

☐☐ Stap 3 - Kolom breedte aanpassen

We gaan de kolombreedte aanpassen. Vervang het blok van net door het volgende:

```
[ 'label' => 'Oppervlakte',  
  'attribute' => 'SurfaceArea',  
  'contentOptions' => ['style' => 'width:30px; white-space: normal;'],  
],
```

- Pas de breedte van de kolom *Inwoners* (*Population*) ook aan; zet de breedte op 30 pixels.

Test of het werkt!

☐☐ Stap 4 - Waarde van kolom aanpassen

Stel we willen km2 achter de oppervlakte plaatsen, dan kunnen we de waarde van de kolom als volgt aanpassen:

```
[  
  'label' => 'Oppervlakte',  
  'attribute' => 'SurfaceArea',  
  'format' => 'raw',  
  'value' => function($data) {  
    return sprintf("%8d k&#13217", $data->SurfaceArea);  
  }  
],
```

```
}  
]
```

Regel 5 t/m 7 in deze code verandert de waarde van de kolom. Dit is een functie en de return waarde van de functie wordt in de kolom afgedrukt.

Test of het werkt!

Als je achter return bijvoorbeeld "Hallo" zet dan wordt er in de kolom Oppervlakte overal Hallo afgedrukt. Probeer maar!

📌 Stap 5 - Link maken van een kolom

In de Capital kolom zien we geen plaatsnaam, maar een nummer. Dit is het ID van een plaats in de city tabel.

Dus de *Capital* kolom uit *country* is de foreign key die wijst naar de primary key *ID* in *city*.

In de city view kun je een plaatsnaam zoeken door bijvoorbeeld de volgende URL te gebruiken:

[http://localhost:8080/city/index?CitySearch\[ID\]=179](http://localhost:8080/city/index?CitySearch[ID]=179)

179 is het ID van Brussels, de hoofdstad van België.

We gaan nu de value van de kolom Capital vervangen door de link.

Laten we om te beginnen de kolom *Capital* een beetje gaan aanpassen:

1. Vervang de kolomnaam van de kolom Capital door 'Hoofdstad'.
2. Verander de breedte van deze kolom naar 30px.
3. Vervang de value van de kolom in

`'format' => 'raw', 'value' => function($data) { return "/city/index?CitySearch[ID]=179"; }` (alle kolommen krijgen dus dezelfde waarde).

Je ziet nu (in de browser):

image1614025244735.png

Test of dit werkt!

Dat is nog geen link. Dat komt omdat we met `...` een link moeten maken. Dat kan maar dat is lastig omdat we met " in " quotes zitten. Dus dubbele quotes in dubbele quotes. We kunnen een 'gewone' `<a href>` maken, maar het is een gepriegel.

Met de Yii module Html kunnen we dit eenvoudiger doen.

We vervangen de regel die de waarde returned in:

image1614026085274.png

Je ziet nu in de browser een nummer (FK naar `city`) en als je daar op klikt, ga je naar de city *Bruxelles [Brussel]*

Test of dit werkt!

We zijn er bijna, we moeten nu alleen de 179 nog variabel maken, anders linkt elke regel naar Brussel.

image1614026443516.png

□ Stap 6 - Testen en laatste aanpassing

1. Maak je code werkend met bovenstaande code.
2. Als je nu op het nummer klikt ga je naar de hoofdstad van het betreffende land.
Test of dit werkt!
3. Verander nu de tekst van de link zodat je in de kolom Hoofdstad, in 'naar hoofdstad'.
4. Als je er op klikt dan ga je dus naar de index view van de hoofdstad.
5. Verander ook de breedte van de kolom naar 200px zodat de kolom netjes past.

De kolom ziet er dus als volgt uit:

image1614026822965.png

□ Stap 7, kolommen aanpassen

1. Zet de kolommen in de volgende volgorde:
Code - Name - Hoofdstad - Inwoners- Oppervlakte
2. Verander de kolomnamen als volgt:
Code - Naam - Hoofdstad - Inwoners - Oppervlakte
3. Zorg dat de naam van het land vet gedrukt wordt (bold).
4. Haal vervolgens de volgnummers (1,2,3,4....) in de eerste kolom weg
5. Verplaats de edit kolom (de laatste) naar voren.

Het hele overzicht ziet er uit zoals te zien is op het volgende plaatje:

image1617127616569.png

☐ Stap 8 - bevolkingsdichtheid

De bevolkingsdichtheid zegt iets over de hoeveelheid inwoners in een land. Natuurlijk heeft Amerika meer inwoners dan Nederland, maar het land is ook vele groter. Om te vergelijken hoe dichtbevolkt een land is, kun je uitrekenen hoeveel mensen er gemiddeld per km² (dit een gebied van 1000x1000 meter) wonen.

bevolkingsdichtheid = aantal inwoners / oppervlakte.

Stel een land heeft een oppervlakte van 100 km² en je er wonen 200 mensen. Dan heb je in dit land dus gemiddeld 2 inwoners per km², want:

$$2 = 200 / 100$$









- Maak nu een nieuwe kolom in het *country* overzicht en bereken daarin de bevolkingsdichtheid.
- Gebruik de php functie *round()* om de bevolkingsdichtheid met 0 decimalen af te drukken.

☐ Aan het einde heb je jouw *gridview* aangepast en ziet het er zo uit:

Home / Countries

Countries

Showing 1-20 of 239 items.

	Code	Naam	Hoofdstad	Inwoners	Oppervlakte	Bevolkingsdichtheid
	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
 	ABW	Aruba	naar hoofdstad	103000	193 km ²	534
 	AFG	Afghanistan	naar hoofdstad	22720000	652090 km ²	35
 	AGO	Angola	naar hoofdstad	12878000	1246700 km ²	10
 	AIA	Anguilla	naar hoofdstad	8000	96 km ²	83

Het Eindresultaat

☐ Inleveren

1. Maak een **schermafdruk** van jouw Country's aangepaste *Gridview* en lever deze in.
2. Stuur ook jouw (aangepaste) **country view** in.

