**PH ro·ect** 'Ga a

4 april 2018 Anjo Eijeriks



Pagina **1** van **12**

**Inhoudsopgave**

[Inleiding 3](#_TOC_250024)

[Opdracht 1, tabellen 3](#_TOC_250023)

[Opdracht 2, het menu 4](#_TOC_250022)

[Opdracht 3, maak verbinding met de database 4](#_TOC_250021)

[Opdracht 4, klanten toevoegen (Create) aan de tabel *klant* 5](#_TOC_250020)

[Opdracht 5, laat alle klanten zien (Read) 6](#_TOC_250019)

[Opdracht 6, zoeken op klantid (Search) 6](#_TOC_250018)

[Opdracht 7, een klant wijzigen (update) 7](#_TOC_250017)

[Opdracht 8, een klant verwijderen (delete) 8](#_TOC_250016)

[Opdracht 9, CRUD en zoeken voor de tabel auto 9](#_TOC_250015)

[Hoe haal ik een voldoende voor dit werkstuk? 10](#_TOC_250014)

[Opdracht 10, Welke auto's horen bij de klanten?(½ punt) 10](#_TOC_250013)

[Opdracht 11, Maak je site mooi (½ punt) 10](#_TOC_250012)

[Opdracht 12, Gebruik SCSS in plaats van CSS (½ punt) 10](#_TOC_250011)

[Opdracht 13, Controle bij wijzigen (Update) (½ punt) 10](#_TOC_250010)

[Opdracht 14, Type auto met eigenaar(½ punt) 10](#_TOC_250009)

[Opdracht 15, verwijderen klant met controles (½ punt) 10](#_TOC_250008)

[Opdracht 16, inlogscherm (½ punt) 10](#_TOC_250007)

[Opdracht 17, verschillende rechten (1 punt) 10](#_TOC_250006)

[Opdracht 18, CRUD voor gebruikers (1 punt) 11](#_TOC_250005)

[Opdracht 19, je eigen profiel (½ punt) 11](#_TOC_250004)

[Opdracht 20, GIT repository (½ punt) 11](#_TOC_250003)

[Opdracht 21, Uploaden auto's (1 punt) 11](#_TOC_250002)

[Afsluiting 11](#_TOC_250001)

[Bijlage: De broncode 11](#_TOC_250000)

Pagina **2** van **12**

## Inleiding

Je gaat een programma maken voor de garagebedrijf. De database bestaat uit twee tabellen: klant en auto. Zoals je weet kan je met elke tabel 4 dingen doen, dat wordt aangegeven met de afkorting CRUD. Die geeft de vier handelingen aan Create, Read, Update en Delete die je met een tabel kunt uitvoeren.

Bij elke database kan je dus voor elke tabel deze vier keuzen uitwerken en dan kan je alle gewenste handelingen met de database doen.

#### Opdracht 1, tabellen

Maak eerst een database *garage,* met daarin de tabellen *klant* en *auto.* Je hoeft niet alle gegevens in te voeren, drie regels per tabel is voorlopig voldoende. Als je straks je programma's gaat testen worden het er vanzelf meer. Let er wel op dat de klantid's die je bij *auto* invult ook echt bestaan in de tabel *klant.*

De tabel klant

**klantid klantnaam klantadres klantpostcode**

Eijeriks, Anjo Dijkwal 145b 3015CD

1. Poenai, V Schiphollaan 32136V
2. Stoop, Sam Stoopplein 45 2023Kl
3. Efes,Mo Noorderweg 321 2012 B

5

6

8

**10**

13

Vliet, Sandra van Hoofdweg 7

3483MN

Kelvin,G.

El Ouafa, Saida Antonic,H. Nikola, **N**

Celsiuslaan 273 4532BV

---•

Binnenweg 229 3045HG

Hoofdstraat 54 2932B8

nlkoslraat 24 3028NN

**klantplaats** Rotterdam Rotterdam Lekkerkerk Schiedam Capelle Capelte Rotterdam Schiedam

Rotterdam

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| De tabel auto |  | | | |
| **autokenteken** | **automerk** | **autotype** | **autokmstand** | **klantid** |
| 04-JOK-9 | Volkswagen | Polo | 4087 | 3 |
| 37-TVR-1 | Opel | Corsa | 80544 | 2 |
| 67-YB-44 | Talra | 603 | 309872 | 1 |
| 95-RP-LR | Flat | Ooblo | 143989 | 1 |
| HV-GB-23 | Jaguar | XJ6 | 2043439 | 1 |

Pagina **3** van 12

###### Opdracht 2, het menu

Maak een map *garage* in de map *htdocs* van XAMP. Type het programma *gar-menu.php* over en sla het op. Als het goed is krijg je dan het volgende scherm te zien. Je kunt de broncode van dit programma vinden in de bijlage.

**Garage menu**

**klant**

* create
* read
* zoeken op klantid
* 11pdate
* **delete**

**auto**

* cmam
* read
* zoeken op ken eken
* 11pdate
* **delete**

Zoals je ziet heb je niet alleen de keuzen Create, Read, Update en Delete, maar ook de keuze *zoeken op klantid* en *zoeken op kenteken.* Eigenlijk is Zoeken ook een vorm van Read, je leest gegevens uit de tabel. Het is gebruikelijk dat je bij elke tabel in ieder geval kunt zoeken op de sleutel. Klantid en kenteken zijn de primaire sleutels van de tabellen *klant* en *auto.*

###### Opdracht 3, maak verbinding met de database

Voor elke keuze uit het menu moeten we een php-bestand maken. Soms horen er zelfs drie bestanden bij één enkele keuze. Deze bestanden maken steeds gebruik van de database, dat betekent datje steeds contact moet maken (connecten) met de database. Het is handig om één programma te maken voor het connecten, dan kunnen alle andere bestanden daar gebruik van maken.

Type daarvoor het programma *gar-connect.php* over en probeer het uit op je database. Op regel 16 zie je een remark (//) staan met een echo opdracht er achter. Tijdens het testen heb je die opdracht nodig. Na het testen kan je de slashes er voor zetten. Dan geeft het programma geen uitvoer meer, maar dat is niet erg. Het drukt alleen iets af op het scherm als het fout gaat.

Pagina **4** van **12**

###### Opdracht 4, klanten toevoegen (Create) aan de tabel *klant.*

Deze menukeuze bestaat uit twee onderdelen: Eerst maak je een formulier waarin je alle klantgegevens kunt toevoegen. Alleen het *klantid* hoef je niet in te vullen, deze kolom is als het goed is auto increment gemaakt. Het formulier kan je vinden in het bestand *gar-create-klant1.php.*

Hieronder zie je wat hoe dit programma er in je browser uit moet zien.

|  |  |  |  |
| --- | --- | --- | --- |
| **garage create klant 1**  **Dit formulier wordt gebruikt om klantgegevens**  in **te voeren.**  **klnntnaam:**1 1 | | | |
| **klantadres:**1 | 1 |  | 1 |
| **klantpostcode:**1  **klantplaats:**1  1 **Verzenden** 1 |  | 1 |

Dan komt het tweede onderdeel: Als je op de *verzenden-knop* drukt dan roept het formulier *gar­ create-klant2.php* op. Dat zorgt er voor dat de gegevens in de tabel *klant* komen te staan. Test je programma een paar keer, je krijgt dan gelijk een aantal klanten in je tabel. De uitvoer van het tweede deel ziet er als volgt uit:

**garage create klant 2**

**Een klant toevoegen aan de tabel klant** in de

**database garage.**

**De klant** *is* **toegevoegd**

ll'.!llg naar het meuu

Er zijn twee mogelijke manieren om de variabelen die door het formulier *gar-create-klant1.php* zijn doorgegeven aan *gar-create-klant2.php* te koppelen aan de query. Je kunt er *bindParam()* voor gebruiken of je koppelt ze aan de query bij de *execute().* Bekijk zelf wat je de makkelijkste methode vindt en gebruik die.

Pagina **5** van **12**

#### Opdracht 5, laat alle klanten zien (Read}

Dit is eigenlijk een heel gemakkelijk programmaatje, je hebt maar één bestand nodig: *gar-read­ klant.php.* Omdat de query de inhoud van een tabel laat zien, zetten we hem ook in een html-tabel.

Dit moet de uitvoer worden (waarschijnlijk heb je andere klantgegevens dan in dit voorbeeld).

**garage read klant**

Dit zijn alle gegevens uit de tabel klanten van de database garage.

4 Efes,Mo

Noorderweg 3212012 B Schiedam

*5* Vliet, Sandra van Hoofdweg 7 3483MN Capelle

6 Kelvin, G.

8 El Ouafa, Saida 10.Antonic, H. 13Nikola,N

14 Watersluis, P

temg: naar het menu

Celsiuslaan 273 4532BV Capelle Binnenweg 229 3045HG Rotterdam

Hoofdstraat 54 2932BB Schiedam

nikostraat 24

**Kipstraat 23**

3028NN Rotterdam

30l **SKJ Rotterdam**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| l | Eijeriks, Anjo | Dijkwal 145b | 3015CD | Rotterdam |
| 2 | Poenai, V | Schiphollaan | 3213BV | Rotterdam |
| 3 | Stoop, Sam | Stoopplein 45 | 2023KL | Lekkerkerk |

#### Opdracht 6, zoeken op klantid (Search}

Dit is eigenlijk dezelfde read-opdracht als bij opdracht 5. Alleen laat je nu de gegevens van één klant zien. Omdat je een klantid moet kunnen invoeren heb je nu eerst een formulier nodig en daarna een bestand met een query om het overzicht te kunnen maken.

Type daarvoor eerst het programma *gar-search-klantl.php.* Je hoeft maar één ding in te lezen en dat is het klantid. Het wordt dus een lekker simpel formulier.

**garage zoek op klantid 1**

***Dit* formulier zoekt een ldant op uit de tabel klanten vandatabase garage.**

**Welk klantid zoekt u?**

1 **Verzenden** f

Dat formulier geeft het klantid door aan gar-search-klant2.php, dat programma voert de query uit en geeft het volgende overzicht:

Pagina **6** van **12**

**garage zoek op klantid 2**

Op klantid gegevens zoeken uit de tabel klanten van de database garage.

*5* Vliet, Sandra van Hoofdweg 7 3483MN Capelle

remg rnrnr het menu

Ook al levert de query nu maar één (of helemaal geen) gegevens op, toch lezen we het resultaat uit met behulp van een lus. Dat is nodig omdat het resultaat van de query altijd een array is, zelfs als er geen gegevens gevonden worden. (Dat kan bijvoorbeeld als je een niet bestaand klantnummer intypt.)

###### Opdracht 7, een klant wijzigen (update)

Dit wordt het lastigste deel van deze garage-applicatie. We hebben er drie bestanden voor nodig. Eerst een formulier dat vraagt welk klantid je wilt veranderen, dan een tweede bestand dat de klantgegevens laat zien. Tot zo ver gaat alles hetzelfde als bij opdracht 6. Maar nu laat je de gegevens niet in een tabel zien, maar op een tweede formulier. Je moet ze immers kunnen veranderen.

Dat tweede bestand (en formulier) geeft de gewijzigde gegevens door aan het derde bestand dat uiteindelijk de query bevat om de klantgegevens in de tabel klant te zetten.

Het eerste deel dat je uitwerkt wordt *gar-update-klant1.php,* het is het eerste formulier waarin je aangeeft welk klantid je wilt wijzigen.

**garage update klant 1**

**Dit formulier wordt gebmil't.om klantgegevens te wijzigen.**

Welk klaotid wilt u wijzigen1?1\_3

j Verzenden j

Dit formulier opent het tweede formulier met de naam *gar-update-klant2.php,* daarmee kan je alle gegevens wijzigen, behalve het klantid. Het klantid mag nooit veranderen omdat het de primaire sleutel is. Voer ook dit programma in. De uitvoer ziet er zo uit:

|  |  |
| --- | --- |
| **garage update klant 2**  **Dit formulier wordt gebruiktom klantgegevens te wijzigen** in **de tabel klant van de database garage.**  klantid: 13 ,  Jdantnaam: jNikola, **N** 1  1 1  klantadres: !nikostraat 24 1  klantpostcode: J3028NN · 1  ,...... ,-  klantplaats:!Rotterdam J | |
| Verzenden 1 |  |

Pagina 7 van **12**

Je ziet dat de gegevens in een formulier worden weergegeven in plaats van in een tabel. Dit tweede formulier geeft de gewijzigde gegevens door aan *gar-update-klant3.php.* Het klantid moet ook worden doorgegeven, dit gebeurt met behulp van een hidden input.

Nadat je dit *gar-update-klant3.php* hebt getypt geeft het de volgende uitvoer.

**garage update klant 3**

**Klantgegevens wijzigen** in **de tabel klant van de database garage.**

**De klant** is **gewijzigd.**

kl:!Jg naar ht-t menu

##### Opdracht 8, een klant verwijderen (delete)

Dat is minder werk dan een klant updaten. Eerst maken we weer een formulier om een klantid in te kunnen voeren. Daarna worden de gegevens doorgegeven aan een tweede formulier dat de klantgegevens zoekt en op het scherm zet. Tot zo ver is het gewoon opdracht 6.

Maar nu is het tweede programma ook weer een formulier omdat we willen vragen of die gegevens echt verwijderd moeten worden. Dat doen we met een checkbox.

Uiteindelijk wordt het klantid en de inhoud van de checkbox doorgegeven aan het derde bestand dat een selectie heeft. Als de checkbox aangevinkt was worden de klantgegevens verwijderd, anders niet.

Type de code voor het eerste formulier *gar-delete-k/ant1.php,* de uitvoer ziet er zo uit:

**garage delete klant 1**

Dit **fomiulier zoekt een klant op** uit **de tabel klanten** ,'all **database garage om hem te kunnen**

**\'envijderen.**

Welk klantid wilt ,u·erwijderen?

15 1

j **Verzenden** 1

Type daarna de code voor het tweede formulier met de *naam gar-delete-klant2.php.*

**garage delete klant 2**

Op klantid gegevens zoeken uit de tabel klanten van de database garage zodat ze verwijderd kunnen worden.

15Brood. Semira Oude Haven 147 310SKL Schiedam

l J Verwijder deze klant.

[ Verzend n [

Pagina **8** van **12**

En tenslotte voer je de code voor gar-delete-klant3.php in. Controleer het programma minimaal twee maal, een keer verwijder je de klant niet, de andere keer verwijder je de klant wél.

**garage delete klant 3**

Op klantid gegevens zoeken uit de tabel klanten van de database garage zodat ze verwijderd kunnen worden.

**De gegevens** zijn **venvijderd.**

Te111g niH1r het menu.

Nu heb je alle basishandelingen gedaan die je met een tabel kunt doen: Create, Read, Update en Delete (CRUD). Zoeken is een eigenlijk een vorm van Read. Er zijn geen controles ingebouwd. Dat gaat soms fout. Probeer maar eens om een niet bestaand klantid te wijzigen, je zult zien dat je dan een hele serie foutmeldingen krijgt!

We hebben nog een tweede tabel, de tabel *auto.* Daarvoor kunnen we dezelfde programma's maken. De namen van die programma's staan voor een deel in de broncode van *gar-menu.php.* Voor deze bestanden gebruik je dezelfde manier van naamgeving als je bij de klanten gebruikt hebt.

#### Opdracht 9, CRUD en zoeken voor de tabel *auto*

Maak nu alle programma's af zodat het hele menu functioneel geworden is.

Pagina **9** van **12**

## Hoe haal ik een voldoende voor dit werkstuk?

Als je alle voorgaande opdrachten goed hebt gedaan dan heb je een 5 verdiend voor je werkstuk. Natuurlijk ga je voor de 10. Die kan je dan ook halen door extra punten te verdienen door de opdrachten hieronder ook uit te voeren.

###### Opdracht 10, Welke auto's horen bij de klanten?(½ punt)

Maak een extra menukeuze, als daarop geklikt wordt moeten alle auto's met hun eigenaren afgedrukt worden.

###### Opdracht 11, Maak je site mooi (½ punt)

Gebruik nu een CSS met de naam *garage.css* om je site mooier te maken dan die van je klasgenoten. Je mag het uiterlijk van je site zo aanpassen als je wilt, het moet er wel duidelijk en professioneel uit blijven zien.

###### Opdracht 12, Gebruik SCSS in plaats van CSS (½ punt)

Eigenlijk is het nog mooier om SCSS te gebruiken in plaats van CSS. Maak een stylesheet met de naam *garage.scss.* Je maakt het stylesheet óf in CSS óf in SCSS.

###### Opdracht 13, Controle bij wijzigen (Update)(½ punt)

We hebben het er al eerder over gehad, als je een niet bestaande klant of auto probeert te updaten komen er foutmeldingen. Bouw een controle in zodat je die meldingen niet meer te zien krijgt.

###### Opdracht 14, Type auto met eigenaar(½ punt)

Het komt in de garagewereld geregeld voor dat een bepaald type auto teruggeroepen wordt naar de garage omdat er een ernstige fout moet worden opgelost. Het is dan ook handig om een keuze in het menu te hebben waarbij je alles auto's van een bepaald type te zien krijgt, met daarbij de klantgegevens zodat je contact kunt opnemen met de eigenaar.

###### Opdracht 15, verwijderen klant met controles (½ punt)

Het is de bedoeling dat een klant alleen verwijderd kan worden als hij geen auto's meer heeft staan in de tabel *auto.*

###### Opdracht 16, inlogscherm (½ punt)

Maak een inlogscherm dat met behulp van een tabel *gebruikers* de toegang tot de site blokkeert of toestaat. Er wordt onderscheid gemaakt tussen monteurs, planners, directie en systeembeheerders.

###### Opdracht 17, verschillende rechten (1 punt)

De monteurs en de directie hebben alleen maar de rechten om gegevens te kunnen zoeken en om ze in te kunnen zien (Read en Search). De planners en systeembeheerders krijgen ook de rechten om gegevens te kunnen invoeren, wijzigen en verwijderen (Create, Read, Searcg, Update en Delete).

Pagina **10** van **12**

###### Opdracht 18, CRUD voor gebruikers (1 punt)

Als je dan toch een tabel voor gebruikers hebt, dan kan je het menu ook aanpassen met CRUD +S voor de gebruikers. Hou rekening met de al eerder genoemde rechten van de gebruikers.

###### Opdracht 19, je eigen profiel (½ punt)

Het is toch wel handig als de monteurs en directieleden hun eigen gegevens in de tabel *gebruikers* mogen aanpassen. Daar hebben ze alleen maar leesrechten. Zorg er voor dat ze hun eigen gegevens wél kunnen updaten, maar niet die van de anderen!

###### Opdracht 20, GIT repository (½ punt)

Zet je hele project in een GIT repository.

###### Opdracht 21, Uploaden auto's {1 punt)

Zorg er voor dat je bij elke auto een foto van die auto kunt uploaden.

# Afsluiting

Je hebt nu gezien dat je bij elke tabel die je maakt altijd deze handelingen kunt verrichten: Create, Read, Search, Update en Delete. Die programma's kan je standaard altijd maken.

Daarbij kan je vaak nog een aantal controles toevoegen, die zullen per bedrijf en per applicatie verschillen. Om er achter te komen welke controles er nodig zijn zal je goed met de opdrachtgever moeten overleggen. Dat geldt ook voor de overzichten, meestal moeten er speciale overzichten gemaakt worden waar bij je op andere gegevens kunt selecteren dan op de primaire sleutels.

Je hebt nu voldoende basis gekregen om voor elke database de benodigde programma's te maken. Hoop ik©!

# Bijlage: De broncode

Op de volgende pagina's volgt de broncode van de vijf programma's die je nodig hebt om CRUD + S te maken voor de tabel klanten.

Pagina **11** van **12**

Pagina **12** van **12**

1. !doctype html
2. <html lang="nl.">
3. <head>
4. <meta name="author" content="Anj.Q Eijeriks">
5. <meta charset="UTF-8">
6. <title> gar-menu.php </title>
7. </head>
8. <body>
9. <hl>Garage menu</hl>
10. <h2>.k.J.ant.</h2>
11. <ul>
12. <li><a href="gar-create-klantl.php">
13. <1-i><a href="gar-read-k.la.nt..php">
14. <li><a href="gar-search-klantl.php">
15. <li><a href="gar-update-klantl.php">
16. <li><a href="gar-delete-klantl.php">

17 </ul>

##### 18

19 <h2>auto</h2>

1. <ul>

create read

##### zoeken op klaotid

update delete

</a></li>

</a></li>

</a></li>

</a></li>

</a></li>

1. <li><a href="gar-create-autol.php">
2. <li><a href="gar-read-auto.php">
3. <li><a href="gar-search-autol.php">
4. <li><a href="gar-update-autol.php">
5. <li><a href="gar-delete-autol.php">
6. *<lul>*
7. </body>
8. </html>

29

create read

##### zoeken op kenteken

update delete

</a></li>

</a></li>

</a></li>

</a></li>

</a></li>

1. <?php
2. *Il* gar-connect.php
3. *Il* maakt. .een connectie met .de database 'garage'
4. *Il* .Aaj..u Eijeriks

5

1. $servernarne = "localhost";
2. $dbname = "garage";
3. $username "root";
4. $password "";

10

1. try
2. {
3. $conn = **new** PDO( "ID.à[.S.ql: host=$servername;dbname-$dbname",
4. $username, $password);
5. $conn->setAttribute(PDO::ATTR ERRMODE, PDO::ERRMODE\_EXCEPTION);
6. *11* echo "Connectie gel11kt <b*-*r.*1>";*

17 }

1. catch(PDOException $e)
2. {
3. **echo** "Connectie misJ11kt: ". $e->getMessage();

21

22 ?>

23

* 1. !doctype html
  2. <html lang="nl">
  3. <head>
  4. <meta name="author" content="Anj\_o\_ Eijeri ks">
  5. <meta charset="UTF-8">
  6. <title>gar-create-klantl.php</title>
  7. </head>
  8. <body>
  9. <hl>garage create .kl.an:t 1</hl>
  10. <p>
  11. Dil formulier HOrd:t gebrni kt om klantgegevens in .t.e voeren.

12 </p>

1. <;form action="gar-create-klant2.php" method="post">
2. klantnaam: <input type="text" name="klantnaamvak"> <br />
3. klantadres: <input type="text" name="klantaciresvak"> <br />
4. klantpostcode: <input type="text" name="klantpostcocievak"><br />
5. klantplaats: <input type="text" name="kl antplaatsvak"> <br />
6. <input type="submit">
7. </form>
8. </body>
9. </html>

22

1. <!doctype html
2. <html lang="nl">
3. <head>
4. <meta name="author" content="Anj.o Eijeriks">
5. <meta charset="UTF-8">
6. <title>gar-create-klant2.php</title>
7. </head>
8. <body>
9. <hl>garage create .kJ.an:t 2</hl>
10. <p>
11. Een .kJ.an:t toevoegen .aan .de ..t..abel
12. .kJ.an:t in .da database garage.

13 </p>

1. <?php
2. // kJ antgegevens uil he..t farm)]] i er hal.en --------------------------
3. $klantid NULL; / / .komt niet. llil he.t formnlier (aJltoi ncrement)
4. $klantnaarn $\_POST["klantnaarovak"J;
5. $klantadres $\_POST[ "klantadresva *k"];*
6. $klantpostcode $\_POST["klantpostcodevak"];
7. $klantplaats $\_POST["kJantpl aatsva k"];

21

1. *Il* klantgegevens invoeren in .de .t.ab..e.l------------------------------
2. **equire**\_**once** "gar-connect ..php";

24

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 25 | $sql = $conn->prepare(" | | |  |
| 26 |  | |  | insert into .klant. values |
| 27 |  | |  | ( |
| 28 |  | |  | :klantid, :k)antnaaro, :k)aotadres, |
| 29 |  | |  | :kJaotpostcade, :klantplaats |
| 30 |  | |  |  |
| 31 |  | | ") ; |  |
| 32 |  | |  |  |
| 33 |  | //manier 1 (of manier 2 gebruiken)-------------------------------- | | |
| 34 |  | // $sql->bindParam(":klantid", $klantid); | | |
| 35 |  | // $sql->bindParam(":klantnaam", $klantnaam); | | |
| 36 |  | // $sql->bindParam(":klantadres", $klantadres); | | |
| 37 |  | // $sql->bindParam(":klantpostcode", $klantpostcode); | | |
| 38 |  | // $sql->bindParam(":klantplaats", $klantplaats); | | |
| 39 |  |  | | |
| 40 |  | //$sql-> execute(); | | |
| 41 |  |  | | |
| 42 |  | // manier 2 | | |
| 43 |  | $sql->execute([ | | |
| 44 |  | "klantid" => $klantid, | | |
| 45 |  | "klantnaam" => $klantnaam, | | |
| 46 |  | "klantadres" => $klantadres, | | |
| 47 |  | "klantpostcode" => $klantpostcode, | | |
| 48 |  | "klantplaats" => $klantplaats | | |
| 49 |  | ]) ; | | |
| 50 |  |  | | |
| 51 |  | **echo** "De klant is toegevoegd <br />"; | | |
| 52 |  | **echo** "<a href='gar-menu.php'> terug naar het menu </a>"; | | |
| 53 | ?> |  | | |
| 54 | </body> |  | | |
| 55 | </html> |  | | |
| 56 |  |  | | |

1. !doctype html
2. <html lang="nl.">
3. <head>
4. <meta name="author" content="Anj..oE.ijeriks">
5. <meta charset="UTF-8">
6. <title>gar-read-llan.t.pbp</title>
7. </head>
8. <body>
9. <hl>garage read llan.t</hl>
10. <p>
11. *Dit* .z.ij,n .a.ll.e gegevens llil .de
12. .:t.ahe.l klanten van .de database garage.

13 </p>

1. <?php
2. // .tabel nitlezen en netjes afdr1Jkken ­
3. require\_once "gar-connect.php";

17

1. $klanten= $conn->prepare("
2. select kJanti d,
3. klantnaam,
4. kJ antadres,
5. k)aotpastcode,
6. klantplaats

24

25

26 $klanten->execute();

27

1. **echo** "<table>";

**Tf)** ;

from

.klant

1. foreach($klanten as $klant)
2. {
3. **echo** "<tr>";
4. **echo** "<td>" **$klant** ["klantid"]
5. **echo** "<td>" **$klant** ["klantnaam"]
6. **echo** "<td>" $klant["klantadres")
7. **echo** "<td>" $klant["klantpostcode"]
8. **echo** "<td>" $klant["klantplaats"J
9. **echo** "</tr>";

38

"</td>";

"</td>";

"</td>";

11</td>";

11</td>";

1. **echo** "</table>";
2. **echo** "<a href='gar-menu.php'> terug naar het menu </a>";

41 ?>

1. **</body>**
2. **</html>**
   1. !doctype html
   2. <html lang="n.l">
   3. <head>
   4. <meta name="author" content="Anj.oEijeriks">
   5. <meta charset="UTF-811>
   6. <title>gar-search-klant1.php</title>
   7. </head>
   8. <body>
   9. <hl>garage .z.aak. op klantid l</hl>

##### <p>

* 1. D.i.t formulier z.oekt .een .klant op .uil
  2. .de .:t.abel klanten van database garage.

##### 13 </p>

1. <form action="gar-search-klant2.php" method="post">..
2. Hel.k klantid .zoekt u?
3. <input type="text" name="klantidvak"> <br />
4. <input type="submit">
5. </form>
6. </body>
7. </html>
8. !doctype html
9. <html lang="nl.">
10. <head>
11. <meta name="author" content="Anj.o**Eijeriks">**
12. <meta charset="UTF-8">
13. <title>gar-search-klant2.php</title>
14. </head>
15. <body>
16. <hl>garage .z..aek op klantid 2</hl>
17. <p>
18. Op klantid gegevens zoeken uil de
19. **..tabel klanten van de** database garage.

13 </p>

##### <?php

1. // klaotid .u.i.t he..t farmnJier.hal.an ­
2. $klantid = $\_POST["klantidvak");

|  |  |  |
| --- | --- | --- |
| 17  18 |  | //klantgegevens.uil.de. ..t.a.b.e.l .hal.an ­ |
| 19 |  | **require\_once** "gar-connect..pb.p"; |
| 20 |  |  |
| 21 |  | $klanten= $conn->prepare(" |
| 22 |  | select klantid, |
| 23 |  | klaotoaaro, |
| 24 |  | klaotadres, |
| 25 |  | klantpastcade, |
| 26 |  | klantplaats |
| 27 |  | from .kl.an.t. |
| 28 |  | where klaotict = ;k)aotid |
| 29 |  | ") ; |
| 30 |  | $klanten->execute(["klantid" => $klantid]); |
| 31 |  |  |
| 32 |  | // klantgegevens laten zien |
| 33 |  | **echo** "<table>"; |
| 34 |  | foreach($klanten as $klant) |
| 35 |  | { |
| 36 |  | **echo** "<tr>"; |
| 37 |  | **echo** "<td>" **$klant** ["klantid"] "</td>"; |
| 38 |  | **echo** "<td>" **$klant** ["klantnaam"] "</td>"; |
| 39 |  | **echo** "<td>" $klant["klantadres"] "</td>"; |
| *40* |  | **echo** "<td>" $klant["klantpostcode"]. "</td>"; |
| 41 |  | **echo** "<td>" **$klant** ["klantplaats"] "</td>"; |
| 42 |  | **echo** "</tr>"; |
| *43* |  |  |
| 44 |  | **echo** "</table><br />"; |
| 45 |  | **echo** "<a href='gar-menu.php'> terug naar het menu </a>"; |
| 46 |  |  |
| 4 7 | ?> |  |
| 48 | **</body>** |  |
| 49 | **</html>** |  |

* 1. <!doctype html
  2. <html lang="nl.">
  3. <head>
  4. <meta name="author" content="Anj.cEijeriks">
  5. <meta charset="U F-8">
  6. <title>gar-update-klantl.php</title>
  7. </head>
  8. <body>
  9. <hl>garage update .kl.ant 1</hl>

##### <p>

* 1. Dil formulier .HQrd:t gebruikt om klantgegevens .t.e wijzigen.

12 </p>

1. <form action="gar-update-klant2.php" method="post">
2. He.lk ·klantid wilt u wijzigen?
3. <input type="text" name="klantidvak"> <br />
4. <input type="submit">
5. </form>
6. </body>
7. </html>
8. ï<!doctype html
9. <html lang="n1">
10. <head>
11. <meta name="author" content="An\_j.c**Eijeriks">**
12. <meta charset="UTF-8">
13. <title>gar-update-klant2.pbp</title>
14. </head>
15. <body>
16. <hl>garage update .kJ.ant 2</hl>

##### <p>

1. Dil formulier .w:o.rdt gebruikt om klantgegevens .t.e wijzigen
2. **in .de .:tab.el .k.1.ant van .de database garage.**

##### 13 </p>

1. <?php
2. // klantid uil heJ:: formulier hal.en ­
3. $klantid = $\_POST("klantidvak"];

17

1. //klantgegevens.uil.de .t.ahe.l bal.e.n
2. require\_once "gar-connect..pb.p";

20

##### $klanten= $conn->prepare("

1. select klantid,
2. klantnaam,
3. klantadres,
4. kl aotpastcacie,
5. klantplaats
6. from .klant

28

29 ") ;

##### where klantid = :klantid

30

##### 31

32

33

34

35

##### 36

37

38

39

*40*

##### 41

42

43

44

45

##### 46

47

48

49

50

##### 51

52

53

54

55

##### 56

57

5B

##### 59

60

61

6?

63

### 64

6:i

##### 66

67

##### 68

?>

</body>

</html>

$klanten->execute(["klaohd" => $klantid]);

// klantgegevens in een nieuw formulier laten zien ­

**echo** "<form action='gar-update-klant3.php' method='post'>"; foreaoh($klanten as $klant)

{

##### // klantid mag niet gewijzigd worden

**echo"** klantid:" . $klant["klantid"];

**echo"** <input type='hidden' name='klantidvak1 ";

**echo** "value=' " . $klant["klantid"] . " '> <br /> ";

**echo"** klantnaam: <input type='text' ";

**echo"** name = 'klantnaamvak' ";

**echo"** value = 111 .$klant["klantnaam"]. "' ";

**echo** " > <br />";

**echo"** klantadres: <input type='text' ";

**echo"** name = 'klantadresvak' ";

**echo** " value = "' .$klant["klantadres"J. "' ";

**echo">** <br />";

##### **echo"** klantpostcode: <input type='text' ";

**echo"** name = 'klantpostcodevak' ";

**echo** " value = '" **.$klant**["klantpostcode"]. "' ";

**echo">** <br />";

##### **echo"** klantplaats: <input type='text' ";

**echo"** name = 'klantplaatsvak' ";

**echo"** value = 111 .$klant["klantplaats"]. "' ";

**echo">** <br />";

}

##### **echo** "<input type='submit'>";

**echo** "</form>";

// er moet eigenlijk nog gecontroleerd worden op een bestaand klantid

* 1. !doctype html
  2. <html lang="nl">
  3. <head>
  4. <meta name="author" content="Anj.aEijeriks">
  5. <meta charset="UTF-8">
  6. <title>gar-update-klant3.php</title>
  7. </head>
  8. <body>
  9. <hl>garage update k.1aD± 3</hl>
  10. <p>
  11. Klantgegevens wijzigen in de .tab.el.
  12. **klam van de database garage.**

13 </p>

1. <?php
2. //klantgegevens.uil he..t fQrmnlier .hal.e.n. --------------------------
3. $klantid $\_POST["klantidvak"J;
4. $klantnaam $\_POST["klantnaarovak"];
5. $klantadres $\_POST("kJantadresvak"J;
6. $klantpostcode $\_POST["klantpasteadevak"];
7. $klantplaats $\_POST["klantplaatsvak"J;

21

1. *Il* updaten klantgegevens
2. require\_once "gar-connect ..p.hp";

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 24 |  |  |  | |
| 25 | **$sql** = **$conn->prepare** | |
| 26 | (" | |
| 27 | update .klant set | | klantnaam | :klantnaam, |
| 28 |  | | klantadres | :klantadres, |
| 29 |  | | kJ antpasti ade | :klantpastcade, |
| 30 |  | | klantplaats | :klantplaats |
| 31 |  | | where klant id | :klantid |
| 32 | ") ; | |  |  |
| 33 |  | |  |  |
| 34 | **$sql->execute** | |  |  |
| 35 | ( [ | |  |  |
| 36 | "klantid" => | | **$klantid,** |  |
| 37 | "klantnaam" => | | **$klantnaam,** |  |
| 38 | "klantadres" => | | **$klantadres,** |  |
| 39 | "klantpostcode" => | | **$klantpostcode,** |  |
| 40 "klantplaats" => **$klantplaats** | | | | |
| 41 |  | J) ; | | |
| 42 |  |  | | |
| 43 |  | **echo** "De klant is gewijzigd. <br />"; | | |
| 44 |  | **echo** "<a href='gar-menu.php'> terug naar het menu </a>"; | | |
| 45 | ?> |  | | |
| 46 | </body> |  | | |
| 47 | </html> |  | | |

1. !doctype html
2. <html lang="nJ.">
3. <head>
4. <meta name="author" content="Anj.QEijeriks">
5. <meta charset="UTF-8">
6. <title>gar-delete-klantl.php</title>
7. </head>
8. <body>
9. <hl>garage delete kJ.aru; 1</hl>

##### <p>

1. Dil formulier .z.o.ek.t .een kJ..an.t op uil
   1. de .tabel. klanten van database garage
   2. om hem .te kunnen verwijderen.

14 </p>

1. <form action="gar-delete-klant2.php" method="post">
2. He.lk klantid wilt u verwijderen?
3. <input type="text" name="klantidvak"> <br />
4. <input type="submit">
5. </form>
6. </body>
7. </html>
   1. !doctype html
   2. <html lang="n1">
   3. <head>
   4. <meta nam.e="author" content="Anj.oEijeriks">
   5. <meta charset="UTF-8">
   6. <title>gar-delete-klant2.php</title>
   7. </head>
   8. <body>
   9. <hl>garage delete kl.ant; 2</hl>
   10. <p>
   11. Op klantid gegevens zoeken .uil .de
   12. .tahe.l klanten van .de database garage
   13. .zodat .ze verwijderd kunnen worden.

14 </p>

1. <?php
2. // klaotid lli.t he.t. farmnJier hal.en ­
3. $klantid = $\_POST["klantidvak"];

18

1. // klantgegevens .ui..t:. .de .t..ab.e.l halen ­
2. **require\_once** "gar-connect.php";

21

1. **$klanten=** $conn->prepare("
2. select kJantid,
3. kJaotoaaro,
4. kJ aotadres,
5. kJantpnstcacie,
6. kJaotpJaats
7. from .kl.an.t
8. where kJaotid = :kJantid
9. Il) **Ï**
10. $klanten->execute(["klantid" => $klantid]);

32

1. // klantgegevens laten zien ­
2. **echo** "<table>";
3. **foreach($klanten as** $klant)

36 {

37 **echo** "<tr>";

38 **echo** "<td>"

39 **echo** "<td>"

40 **echo** "<td>"

41 **echo** "<td>"

42 **echo** "<td>"

43 **echo** "</tr>";

44

45 **echo** "</table><br />";

46

**$klant**["klantid"]

**$klant**["klantnaam"]

$klant["klantadres"]

**$klant**["klantpostcode"]

**$klant**["klantplaats"]

"</td>";

"</td>";

"</td>";

"</td>";

. "</td>";

47

48

49

##### 50

51

5:2

5:3

54

55

515

57

513

?>

</body>

</html>

**echo** "<form action='gar-delete-klant3.php' method='post'>";

// klantid mag niet meer gewijzigd worden

echo "<input type='hidden' nam.e='klantidvak' value=$klantid>";

// Waarde 0 doorgegeven als er niet gecheckt wordt

**echo** "<input type='hidden' name='verwijdervak' value='0'>"; **echo** "<input type='checkbox' name='verwijdervak' value='l'>"; **echo** "Verwijder deze klant. <br />";

echo "<input type='submit'>";

**echo** "</form>";

1. !doctype html
2. <html lang=11nl.">
3. <head>
4. <meta name="author" content="Anj.c.Eijeri **ks">**
5. <meta charset="UTF-8">
6. <title>gar-delete-klant3.php</title>
7. </head>
8. <body>
9. <hl>garage delete lla.n.t 3</hl>
10. <p>
11. Op klantid gegevens zoeken .uil .de
12. .tabel klanten van .de database garage
13. .z.o.dat .ze verwijderd kunnen worden.

14 </p>

1. <?php
2. // gegevens .ui..t hel:. farmul i er halen -------------------------------
3. $klantid $\_POST["klantidvak");
4. $verwijderen = $\_POST["verwijdervak"J;

##### 19

1. *Il* klantgegevens verwijderen--------------------------------------
2. if($verwijderen)
3. {
4. **require\_once** "gar-connect..p.hp";

24

25 $sql = $conn->prepare("

26 delete from ..k.l.an.t.

27 where

28 n) ;

klantid

:klaotid

29 $sql->execute(["klantid" => $klantid]);

30

31 **echo** "De gegevens .z.i..j.n verwijderd. <br />"; 32

33 **else**

34

35 **echo** "De gegevens zijn niet verwijderd. <br />"; 36

37

1. **echo** "<a href='gar-rnenu.php'>Terug naar het menu. </a>";

39 ?>

1. </body>
2. </html>