PHP5 PDO

A - Maak in PHP een databaseverbinding

Maak een nieuw .php bestand met de volgende naam:

connect-[jouwnaam].php - dus bijvoorbeeld *connect-hans.php*

Maak in dit php-bestand een verbinding met een database, genaamd ‘games’. Gebruik hiervoor de standaard gebruiker in xampp. Geef het verbindingsobject de naam ‘$conn1’.

Maak daarna een tweede verbinding met een tweede database, genaamd ‘artiesten’. Gebruik hiervoor een ander gebruikersaccount. Deze heeft als gebruikersnaam ‘roc-student’ en als wachtwoord ‘welkom123’. Geef het verbindingsobject de naam ‘$conn2’.

Inleveren:

connect-[jouwnaam].php

Tip: Je gaat vaker een verbindingen met databases maken, sla dit soort bestanden op een voor jou logische plek op zodat je ze later snel terug kunt vinden als naslagwerk.

B1 - insert game

Open PhpMyAdmin in je browser en maak een nieuwe database aan met de naam ‘winkel’.

Importeer het volgende bestand games.sql

Maak een nieuw .php bestand met de volgende naam:
nieuw-spel1-[jouwnaam].php
(vervang [jouwnaam] met je eigen naam)

Maak in dit bestand een database verbindingsobject met de naam ‘$conn’.

Maak in dit bestand een variabele $sql, waarin je als string een query maakt waarmee je de volgende game toevoegt aan de tabel ‘games’ in de database ‘winkels’:

Titel: F1 2022
Uitgever: EA Sports
Platform: Playstation 5
Voorraad: 28
Prijs: 69,-

Zorg met een exec() er voor dat de query wordt uitgevoerd en de gegevens in de database worden toegevoegd.

Let op de tabel 'games' heet in jouw versie van de database niet meer games. Bestudeer de database en bedenk zelf wat de nieuwe naam is van deze tabel.

Tip1 : Het id van de tabel is auto-increment. Deze hoef je dus geen waarde mee te geven.

Tip2 : Test de query eerst in PhpMyAdmin om te kijken of deze werkt. Dit kan in PhpMyAdmin door links op de database ‘winkel’ te klikken en daarna de tab ‘SQL’ te selecteren.

Inleveren

Het geteste en werkende PHP-bestand dat je hebt gemaakt.

B2 - insert game deel 2

Maak een nieuw .php bestand met de volgende naam:
nieuw-spel2-[jouwnaam].php
(vervang [jouwnaam] met je eigen naam)

Voeg net als in opdracht 2a een nieuw spel toe in de tabel ‘games’. De gegevens van dit spel stop je eerst in variabelen. Deze variabelen gebruik je vervolgens om de query mee te maken.

Bijvoorbeeld:

image.png

Voeg op deze manier een zelf verzonnen game toe met titel, uitgever, platform, voorraad en prijs.

Inleveren:

- nieuw-spel2-[jouwnaam].php
  (vervang [jouwnaam] met je eigen naam)

- Een screenshot van je browser met daarin PHPMyAdmin en de weergave van je databasetabel met het nieuwe spel

B3 - Insert met form

We gaan nog een stapje verder. In plaats van de spellen die je in de database wilt toevoegen in het PHP-bestand te zetten, ga je nu een formulier maken waarmee een bezoeker van de pagina een game kan invullen om toe te voeren.

Maak twee nieuwe .php bestanden met de volgende namen:
nieuw-spel3-[jouwnaam].php
voeg-toe3-[jouwnaam].php
(vervang [jouwnaam] met je eigen naam)

image.png

Inleveren:

- De twee in deze opdracht gemaakte .php-bestanden
- Een screendump van je browser met het formulier

C1 - Ophalen uit database deel 1

- In deze opdracht halen we alle gametitels uit de database en tonen deze in de browser in een list (<ul>).
- Gebruik hiervoor de tabel producten van de database webshop die je eerder hebt gebruikt.
- Maak nieuwe .php bestanden met de volgende namen:
  opdracht-c1-[jouwnaam].php
  database-connectie-[jouwnaam].php
  (vervang [jouwnaam] met je eigen naam)
- Zorg voor volledige en nette html in het opdracht-c1-[jouwnaam].php bestand (<html>,<head>,<body>, etc.)
- het bestand met de database-connectie gebruik je door middel van een ‘require’
- Het eindresultaat ziet er ongeveer uit als onderstaande plaatje:

  image.png

Voor een bonuspunt:

Zet achter idere game-titel, het platform tussen haakjes. De output op het scherm ziet er dan uit als onderstaand plaatje:

image.png

Inleveren:

- opdracht-c1-[jouwnaam].php
- database-connectie-[jouwnaam].php
- Een screenshot van je gehele browser met daarin het resultaat op je scherm en de url in de adresbalk.
  (vervang [jouwnaam] met je eigen naam)

C2 - Ophalen uit database deel 2

> Deze opdracht is een vervolg op de vorige opdracht.

- Maak nieuw .php bestand met de volgende naam:
  opdracht-c1-[jouwnaam].php
  (vervang [jouwnaam] met je eigen naam)
- Include hierin het bestand database-connectie-[jouwnaam].php uit de vorige opdracht (met een 'require').
- Haal alle gegevens van de tabel 'klanten' op vanuit de database
- Maak hier een nette html-tabel van met tabelheaders en toon deze op het browserscherm met de kolommen:
  - voornaam
  - achternaam
  - woonplaats
- De output op je scherm moet er ongeveer uitzien als onderstaand plaatje:

Tip: Doe het stap voor stap:

- Maak eerst een html-tabel met één rij met vaste testnamen (dus nog niet uit de database).
- Bepaal daarna welk gedeelte van de tabel herhaald moet worden en dus in de foreach-loop moet komen te staan.
- Vervang daarna de vaste waarden met de gegevens uit de database

image.png
Inleveren:

- opdracht-c2-[jouwnaam].php
- database-connectie-[jouwnaam].php
- Een screenshot van je browser met daarin het resultaat op je scherm en de url in de adresbalk.
  (vervang [jouwnaam] met je eigen naam)

D1 - Voorbereiding

Opdracht a

In deze opdracht ga je vooral voorbereidend werk doen voor het kunnen maken van de laatste twee opdrachten. Je gaat hierin verder met de tabel, die je in opdracht C2 (SELECT) hebt gemaakt.

Voeg hieraan twee kolommen toe. In deze kolommen staan per rij twee links.
• Een link voor het verwijderen van het record (van de betreffende rij)
• Een link voor het updaten van het record (van de betreffende rij)

d1-opdracht.png

De bedoeling is dat in de link een argument wordt meegegeven die je kunt gebruiken voor het maken van een query. Als je een query wilt maken voor het verwijderen van een bepaald record, geef je meestal de primary key mee. Dus als je bijvoorbeeld het record van ‘Jan’ wilt verwijderen en Jan heeft als primary key: ‘id = 8’, dan zal de link er uitzien als iets van:

https://studentnummer.jouw.website/verwijder.php?id=8

De waarde van id kun je in je script (verwijder.php) dan weer gebruiken als een $_GET variabele.

! in deze opdracht maak je alleen de links in de tabel. Het verwijderen / updaten, hoeft dus nog niet te werken!

Inleveren:

- opdracht-D1-[jouwnaam].php
- Een screenshot van je browser met daarin het resultaat op je scherm en de url in de adresbalk.
  (vervang [jouwnaam] met je eigen naam)

D2 - Verwijderen van een record

Zorg ervoor dat als je in je browser op een 'verwijder'-link klikt, het betreffende record wordt verwijderd uit de database en dat de tabel (na het verwijderen) in zijn geheel weer wordt getoond. Het verwijderde record is uiteraard verdwenen op je scherm.

Bedenk zelf hoe je dit doet en of je alles verwerkt in één script, of dat je meerdere php-bestanden aanmaakt. Beide is mogelijk en goed.

Inleveren:

- opdracht-D2-[jouwnaam].php
- database-connectie-[jouwnaam].php
- eventueel andere .php bestanden die je gebruikt hebt. Zorg ervoor dat jouw naam in de bestandsnaam staat. Bijvoorbeeld: verwijder-[jouwnaam].php
- Een screenshot van je browser met daarin het resultaat op je scherm en de url in de adresbalk.
  (vervang [jouwnaam] met je eigen naam)

D3 - Updaten van een record

Zorg ervoor dat als je in je browser op een 'move'-link klikt, het betreffende record wordt geüpdate in de database en dat de tabel (na het updaten) in zijn geheel weer wordt getoond. Het geupdate record wordt uiteraard aangepast op je scherm getoond.

Wat moet er worden geüpdatet?

- Als de woonplaats van klant niet 'Amstelveen' is, update je de woonplaats van die klant naar 'Amstelveen'.
- Als de woonplaats van klant 'Amstelveen' is, update je de woonplaats van die klant naar een zelf verzonnen woonplaats in de provincie Gelderland (zoek er maar een op).

Bedenk zelf hoe je dit doet en of je alles verwerkt in één script, of dat je meerdere php-bestanden aanmaakt. Beide is mogelijk en goed.

Inleveren:

- opdracht-D3-[jouwnaam].php
- database-connectie-[jouwnaam].php
- eventueel andere .php bestanden die je gebruikt hebt. Zorg ervoor dat jouw naam in de bestandsnaam staat. Bijvoorbeeld: verwijder-[jouwnaam].php
- Een screenshot van je browser met daarin het resultaat op je scherm en de url in de adresbalk.
  (vervang [jouwnaam] met je eigen naam)

Schrijf een reflectie

Het is gelukt! Je Crudchallenge is ingeleverd en goed gekeurd. In deze challenge zijn veel onderdelen die je hebt geleerd tijdens de verschillende PHP-modules samengekomen. Veel technieken die je hebt gebruikt in de challenge waren een herhaling voor je. Maar bij elke opdracht kom je ook weer nieuwe uitdagingen tegen.

Om stil te staan bij wat je van deze challenge hebt geleerd, ga je (minimaal) twee reflecties hierop maken en inleveren.

Bij een reflectie kijk je terug op wat je hebt gedaan en bedenk je wat er goed is gegaan. Of wat beter had gekund. Dit om bij een volgende keer dat je iets vergelijkbaars tegenkomt te bedenken wat je hebt geleerd. Wat je wéér zo zou doen, of juist anders.

Voorbeeld van een geval waar een reflectie over gemaakt zou kunnen worden:
Je hebt voor het maken van de tabel een while-loop gebruikt. Maar dat vond je lastig en kwam er na het vragen van hulp aan een collega-student achter dat je beter een foreach-loop kon gebruiken.

Een reflectie (stuk tekst) hoeft niet heel uitgebreid te zijn. Beschrijf beknopt, maar wel in duidelijk leesbare zinnen waar het over gaat.

Download het reflectiesjabloon en vul jouw reflecties in. Maak van het bestand een pdf-bestand met als naam:

reflecties-<jouw-naam>.pdf

Revision #2
Created 2024-06-08 15:59:36 UTC by Max
Updated 2024-06-08 18:33:40 UTC by Max