

Introlektion

PHP är ett av de enklare språken att lära sig just pga. dess dynamiska struktur. Det används för att bygga upp både stora och mindre system. Några vanliga system som använder sig av PHP är exempelvis wordpress, jomla och drupal. De flesta stora populära sidorna använder sig av PHP exempelvis skapades facebook i PHP och mängder av andra vanliga system/hemsidor.

Observera att alla "citationstecken" i uppgiften måste skrivas om när du skriver din kod, det går därmed inte att kopiera exempelkoden rakt av från introduktionen.

Ett första kommando

PHP används oftast i samband med HTML, och används oftast till att utveckla hemsidor. Det går att använda PHP till andra saker än hemsidor, men det var ursprungligen skapat för detta syfte. Och det är där som PHP presterar bäst.

När man skriver i PHP måste man först se till att man har en fungerande kompilator, det går alldeles utmärkt att använda WAMP eller skolans server för att programmera i PHP.

Alla filer som bearbetar PHP script ska avslutas med ändelsen .php, exempelvis ska en indexsida till en hemsida heta index.php

En fil som heter .php kan även innehålla HTML kod och taggar, bara för att filen heter .php så kommer inte datorn att behandla allt som står där som PHP script. Du måste som programmerare först säga till datorn att den ska behandla det du skriver som PHP script.

För att göra detta så ska du alltid i början och slutet av dina kodstycken skriva på följande sätt för att säga åt webbserverna att den ska behandla det som står mellan dessa taggar som PHP script.

```
<?PHP
```

```
?>
```

Du kan börja med det enklaste kommandot som man oftast använder i alla programmeringsspråk, vi börjar helt enkelt med att skriva "Hej världen". Kommandot i PHP för att skriva ut text till användaren heter echo och åtföljs av citationstecken. Varje programmeringsrad ska i likhet med många andra språk avslutas med semicolon. Se detta exempel.

```
echo ("hej världen");
```

Lägg nu ihop dessa två och skriv ditt första program.

Citationstecknen använder vi när vi säger åt scriptet att den ska skriva ut den text som vi har skrivit in. Om vi struntar i citationstecken kommer programmet inte att fungera som det borde. Men vi kan lägga in tal utan citationstecken. Om vi skriver saker utan citationstecken då tror datorn att det är en beräkning som den måste göra. Prova skriva 1+1 med och utan citationstecken bakom echo, men kom ihåg semikolon för att avsluta kodraden.

Variabler i PHP

Variabler kommer vi att gå in på lite mer senare men kort förenklat kan vi i PHP betrakta dom som ett namn som har ett värde kopplat till sig. Vi kan ha namnet tal och koppla ett siffervärde till det. Och vi kan ha namnet text och koppla en text till det.

Variabler i PHP script är mycket enklare än i många andra språk, alla variabler börjar med ett dollartecken och sedan variabelns namn. Du behöver inte definiera om variabeln ska innehålla ett tal eller en text som man behöver göra i andra språk.

```
$text = "hej världen";
```

Prova nu att skriva variabelns namn efter echo istället för den text som stod där tidigare och se vad som händer. Prova även att skriva 1+2, eller liknande talberäkningar och låt scriptet skriva ut det.

När man programmerar så gäller en sak i alla språk oavsett om du sitter i excel, C# eller php. När då säger åt att programmet ska göra en beräkning så måste du använda parenteser på rätt ställe, det är bara säga åt datorn vilka beräkningar den ska göra först. Det fungerar precis som vanlig matematiska regler att det inom parenteserna ska räknas ut först. Ibland kan det gå bra utan att använda parenteser, men det kan ibland ge konstiga problem. I denna kursen gäller det att använda dom eller göra om uppgiften. Exempelvis

```
$summa = (tal1/tal2)*(tal3/tal4);
```

Använda sig av `_GET`

Ni har säkert sett många hemsidor där ni ser konstiga tecken bakom raderna och kanske undrat hur detta fungerar. Exempelvis www.facebook.com?ref=mokki det är ett system som kallas för GET, och som används av hemsidan för att skicka meddelanden till sig själv för att den ska kunna bestämma vad användaren vill se. Det finns även ett annat system som heter POST där hemsidan skickar hemliga meddelanden som användaren inte kan se för att bestämma vad användaren ska se.

Vi ska börja med GET då det är en bra struktur för att skapa en hemsida. En get består av en variabel en namn som har ett värde kopplat till sig. I <http://it.guldstadsgymnasiet.se/?meny=php> har vi namnet meny som har fått värdet php.

Du kan börja med att säga till att hämta in GET informationen i en variabel och det gör vi på med kommandot `$_GET`, se detta exempel.

```
$text = $_GET["text"];
```

Prova nu att skriva in detta i ditt script säg åt en echo att skriva ut variabeln text som du laddar in och ladda sedan sidan. Gå in till navigationsfältet och skriv `?text=hejsan` efter adressen till det php sida som du laddar. Nu borde du kunna skriva in saker i navigationsfältet på din webbläsare som din hemsida sedan återberättar till dig.

Detta är det system vi ska använda oss av för att bygga upp menyssystemet på din hemsida.

Kombinera PHP och HTML

Nu när du kan ta in information från din hemsida så kan du prova att lägga till lite länkar på din sida, utanför din scripttag som du skrivit kan du använda htmltaggar som vanligt. Prova att göra en a href och sedan hänvisa den till texten i förra övningen. Exempelvis

```
<a href="?text=hejsan">Hemsidan skriver hejsan</a>
```

Ladda sidan och testa klicka på länken, och din hemsida borde skriva ut text till användaren av hemsidan. Nu kanske du börjar förstå att det går att använda detta som ett navigationssystem på hemsidan där vi lätt kan säga till php scriptet vad den ska göra.

IF satser

Prova nu att skapa en href där värdet på text är tal. Så att datorn skriver ut tal på skärmen när du trycker på länken. Nu ska vi kombinera kraften mellan html och PHP i ett enkelt exempel. Nu ska vi använda ifsatsen och säga åt scriptet att den ska göra något annat om namnet tal står.

En ifsats är en villkorssats som säger åt scriptet att den ska göra något ifall något speciellt inträffar, den kollar oftast olika variabler mot varandra eller mot förinställda värden. En ifsats börjar alltid med IF och har sedan villkoren som ska inträffa i en parentes efter ifsatsen. När man i programmeringsspråk använder ifsatsen ska man alltid använda sig av två likamedstecken annars tror ifsatsen att man vill ändra värdet på variabeln som man använder.

Skriver vi \$text = tal i programmeringsspråk så kommer programmeringsspråket tro att vi vill ändra variabeln text till värdet tal, skriver vi däremot \$text == tal så säger vi åt scriptet att jämföra dessa för att se om dom är likadana, vi måste också som vanligt använda situationstecken alltid när det handlar om text, annars kommer ert program inte fungera som det ska. Se detta exempel.

```
if ($text == "tal")
```

För att säga åt vad ett script ska göra efter ifsatsen gör man på detta sätt

```
if ($text == "tal") echo ("Vill du att jag ska räkna ut ett tal?");
```

Nu borde erat script skriva ut något till när du skrivet tal i din webbläsare eller klickar på din länk. Men den kommer fortfarande att skriva ut tal efteråt, tänk på att du också kan använda html taggar i echo. Exempelvis på detta sätt

```
if ($text == "tal") echo ("Vill du att jag ska räkna ut ett tal?<br />");
```

Men vi vill få bort att den skriver tal och bara skriver ut det vi vill istället för tal, då kan vi använda os av elseif eller else. Elseif och else fungerar på liknande sätt.

```
if ($text == "tal") echo ("Vill du att jag ska räkna ut ett tal?<br />");  
elseif ($text == "hej") echo ("Om du skriver hej så skriver jag hej<br />");  
else echo ("Det här skriver jag oavsett vad du skriver som inte innehåller tal eller hej<br />");
```

Du kan nu ta bort den gamla echo som du har i koden som ligger utanför ifsatsen. Du kan även lägga till flera elseif satser som reagerar på olika texter och som då skriver ut olika saker, prova att lägga till några texter.

Komma igång med includes och kommentarer

Det är vanligt att man bygger upp hemsidor i moduler och ofta inkluderar andra filer för att enklare kunna bestämma vad som ska visas för användaren. Detta är väldigt viktigt när man börjar bygga större system.

Moduluppbyggnaden fungerar lite förenklat som att i en sida så hänvisar man till en annan sida, man säger helt enkelt åt PHP scriptet att den ska ladda ett annat skript när man säger åt den att göra detta. Kommandot för detta är `include("")`; inom situationstecknen skriver man sökvägen till den fil man vill inkludera. Det går även bra att använda variabler, men mer om det senare.

Om vi nu exempelvis har en fil som heter `index.php` kan den se ut på detta sätt

```
<?php
```

```
include("meny.php");
```

```
?>
```

Includen pekar (hänvisar) till en fil som heter `meny.php`, och med namnet att döma så kan vi anta att det handlar om någon slags meny. Döp alltid filerna till saker så att vem som helt kan lista ut vad den går till. Använd även kommentarer när ni skriver er kod för att ytterliggare förtydliga. Kommentarer kan man skriva på olika sätt, men vi ska börja med att använda oss av `//` före kommentaren.

Allt som står bakom `//` kommer PHP scriptet inte att kompilera den kommer ignorera det som står där, och dom enda som kan läsa detta är de programmerare som jobbar med scriptet. Det kan se ut såhär.

```
<?php
```

```
// Ladda in olika sidor som behövs
```

```
include("meny.php"); // Laddar in filen som innehåller huvudmenyn
```

```
...
```

```
?>
```

Kommentaren kan man sätta på olika ställen antingen direkt över efter som i detta fall, behöver man skriva flera rader så skriver man bara `//` framför texten. Kommentarer är viktigt att ni skriver, alla uppgifter som lämnas in utan kommentarer blir automatiskt ej godkända och ni behöver komplettera eller göra om uppgifterna.

Skapa en fil och prova på att inkludera den i din PHP kod, prova sedan att låta PHP scriptet inkludera filer beroende på olika GET variabler som skickas till sidan. Det är viktigt att du förstår det inför nästa moment i denna kurs.

Är du färdig med allt och känner dig trygg dessa enkla php kommandon kan du börja bygga upp din hemsida i uppgift 1.