

Első PHP oldalunk

- Hogyan hozunk létre egy php oldalt? Hogyan futtasuk?
- Miként használjuk a PHP kódot?

Az első programunk

A kezdéshez vegyük elő az egyik PHP -t támogató IDE -t. Készítsünk egy file-t, „elso.php” néven. Ebben a file-ban 3 sorunk lesz. A cél, hogy megismerjük a PHP beállításait. A file-t amennyiben XAMPP -ot telepítettünk, akkor az alapértelmezett XAMPP könyvtárban belül található htdocs mappába kell tenni. A böngészőben pedig, localhost/elso.php -ként lehet meghívni. Ha a htdocs könyvtárban létrehoztunk egy könyvtárat a program file-nak, akkor localhost/könyvtárneve/elso.php formában kell futtatni a böngészőben a programot.

2.1 Példaprogram

```
<?php  
phpinfo();  
?>
```

A 2.1 ben található példaprogram egyetlen függvényt tartalmaz, amely a PHP beállításairól jelenít meg egy táblázatot a böngészőben.

A file neve **elso.php** programunkban fontos a file kiterjesztése, ez alapján tudja a webszerver (webkiszolgáló), hogy PHP kódot tartalmazó file-ról van szó, és átadja a file-t a PHP értelmezőnek.

PHP blokkok (tag-ek)

A PHP kódot alapértelmezetten úgy találták ki, hogy a HTML kód közé vegyítve kell beírni, ahhoz hogy a kódot jól el lehessen különíteni, valamilyen tag-et kell használni. Az így elkülönülő kódot értelmezi és futtatja a PHP értelmező. Ha ezt nem tennénk meg, akkor HTML kódnak értelmezi az egészet a webszerver és küldi a böngészőnek változtatás nélkül. (Ha a tag-et elhagyjuk, akkor a böngészőben látszik a PHP kód)

2.1 Táblázat PHP nyitó és záró tagek

Hagyományos <?php ?>

Rövid <? ?>

A 2.1 -es táblázatban lévő tag-ek közül az első, az amit a leggyakrabban használnak.

File

Php , html keverése

Első megközelítésben a php utasításokat a HTML oldalba ágyazzuk be, az oldal kiterjesztése php, de alapvetően egy HTML oldalba írjuk a tag-eket. Ha a php file-ban nincs html az sem probléma, a phpinfo(); utasításnak nem kell HTML fejléc és törzs.

Oldal futtatása

A legtöbb IDE-ben a jobb egérgombra kattintással, lőjön egy gyors menü, amelyben van futtatás menüpont. A NETBEANS fejlesztő eszköze el tudja indítani az XAMPP alatt vagy más lokális szerver alatti php file-okat a böngészőben. A Visual Studio -ban ehhez egy beépülő modul kell.

A PHP nyelv

Alkotóelemek

A nyelv amely arra szolgál, hogy az adott feladat megoldását a számítógép által érthető formában fogalmazzuk meg, több építő elemből áll. Ezek közül az első, amit meg kell ismernünk az a változó, amelyben az adatunkat tárolni tudjuk ideiglenesen. A változó az operatív memóriában jön létre, életciklusa legkésőbb a program futásának végén fejeződik be.

Változók

A változók olyan tárolók, amelyek egyszerre egy adatot tárolnak. A program során azért van szükségünk ezekre a tárolókra, mivel nem fix adatokkal dolgozunk, hanem általánosan megfogalmazható feladat megoldásokat írunk le, a változókkal megfogalmazzuk, hogy milyen feladatot szeretnénk elvégeztetni, majd különböző értékekkel megtöltve a program elvégzi a már általánosan megírt műveleteket.

Lássunk egy példát, ki szeretnénk írni, hogy 2+3 értéke mennyi.

```
<?php  
    print(2+3);  
?>
```

A program futásának eredménye 5, ez jelenik meg a böngészőben. Fontos! Ez a program csak erre képes, amennyiben később 5+10 -et szeretnénk kiszámoltatni, akkor újra meg kell írni a programot. Ezért, ha általánosan megfogalmazzuk két szám összeadását, akkor már bármilyen számot írhatunk be a program kiszámolja azt.

```
<?php
    // $a és $b -nek értéket kell adni, amely érték kívülről érkezik
    print($a+$b);
?>
```

Az utolsó példa önmagában nem működőképes, mivel az \$a tárolónak / változónak sehol nem adtunk értéket, konkrét űrlapból érkező értékbevitelt később magyarázzuk.

Változó neve

A php nyelvben a változó \$ jellel kezdődik, karakterekből áll, tartalmazhat számot, de a \$ jel után nem kezdődhet azzal. Továbbá az aláhúzás (_) jel is használható a névben. Egy két példa a névadásra:

Megengedett	Nem megengedett
\$osszeg	\$osszeg erteke
\$darab2	\$2darab
\$egesz_ertek	\$egesz-ertek

Létrehozásuk

A változók létrehozása eléggé egyszerű, meg kell adni a változó nevét és értéket kell adni a változónak.

```
<?php
    $osszeg = 0;
?>
```

Ebben az esetben a változó értéke 0 lesz és a típusa szám. A programsor tartalmaz egy változó nevet egy értékadást (=) és egy egész szám értéket (integer literált). Az egyenlőségjel itt értékadást jelent nem összehasonlítást! Fontos jellemzője a változónak, hogy milyen típusú lesz. A PHP nyelv gyengén típusos, azaz nem kell megadni a változó típusát, hanem az első értékadásból tudja meg a program, hogy milyen típusa lesz a változónak, ez menet közben egy másik értékadással változtatható. Óriási szabadságot ad ez a lehetőség, de rengeteg hiba lehetőséget is rejt magában.

Adattípusok

A php a következő adattípusokat támogatja

- null
- bool (logikai típus értéke lehet true, false)
- int (egész szám, maximum értéke 32 vagy 64 bites rendszerben eltérő)
- float (floating-point number) (lebegő pontos szám, tört szám)
- string (karakterlánc, szöveg tárolása)

- array (tömb, több érték egy szerkezetben tárolásához)
- object (objektum, OOP programozás során használatos)
- callable (meghívható típus)
- resource (erőforrás típus)

A PHP egy dinamikusan tipizált nyelv, ami azt jelenti, hogy nem kell megadni a változó típusát, mivel az futás időben kerül meghatározásra. A típus meghatározza a rajtuk végrehajtható műveletek körét.

Műveletek, operátorok

Állandók (konstansok)

Vezérlési szerkezetek (Flow Control)

If, switch, match

For, foreach,

While

Gyakorlat

Háromszög területe

Tömb nélküli feladatok

Táblázat rajzolás

Táblázat csíkozása (Stripe)

1-100 ig 10 x 10 es táblázat

Palindrom