

Ćwiczenie 6

PHP – dynamiczne strony WWW

Czas trwania: 90 min.

Zadania do wykonania:

1. **W oparciu o instrukcję przećwiczyć podstawowe instrukcje w zakresie PHP.**

Ćwiczenie wykorzystuje powstałe wcześniej strony z kodem html.

(Czas trwania: ok. 45 min.)

W instrukcji zamieszczono podstawowe informacje związane z językiem PHP.

Instrukcja ma na celu zapoznać studenta ze specyfiką pisania skryptów.

2. **Na podstawie przeprowadzonych ćwiczeń wykonać obsługę formularza rejestracji użytkownika, którego zadaniem jest pobrać dane z przesłanej formatki i zaprezentować na stronie w sformatowany (CSS) sposób.**

(Czas trwania: ok. 45 min.)

Samodzielnie wykonać skrypt obsługi posługując się plikiem formularza z ćwiczenia nr 2.

Wynikiem końcowym ćwiczeń jest utworzenie scenariusza obsługi rejestracji z prezentacją danych rejestracyjnych na stronie WWW.

Objaśnienia

PHP – jest językiem dynamicznych stron WWW. Tworzone skrypty pozwalają na współpracę zarówno z kodem HTML, CSS, JavaScript jak i SQL. Współpraca skryptów z bazą danych MySQL skutkuje bogatym zapleczem informacyjnym dostępnym w ramach budowanego serwisu internetowego.

Ćwiczenia praktyczne

1. Utworzyć w katalogu `/var/www/html` plik o nazwie `obsługa.php`. Wyedytować plik w edytorze gEdit a następnie uzupełnić zawartość pliku o podstawowe znaczniki informujące o treści PHP. Następnie w pasku adresu uruchomionej przeglądarki internetowej wpisać adres: `http://localhost/obsługa.php`

```
<?php
echo „<b>Witaj Świecie !!!</b> <p> To mój pierwszy skrypt
PHP“;
?>
```

efekt: W oknie przeglądarki powinien pojawić się tekst w dwóch akapitach. Proszę zwrócić uwagę, że w linii wyjścia są wykorzystane znaczniki HTML.

2. Komentarze w skrypcie pozwalają na opis treści i przez to ułatwiają poprawki oraz orientację w kodzie. Treść skryptu staje się bardziej przejrzysta. Wstawić poniższy kod

– linijka po linijce (sprawdzić za każdym razem jak działa) - kod do pliku *obsługa.php*:
<! - - to jest komentarz html - - >
// to jest komentarz jednowierszowy
to jest komentarz wielbicieli powłok (jednowierszowy)
/* tak można tworzyć komentarze wielowierszowe - takie na przykład jakie się spotyka w Javie */

efekt: W zasadzie nic nie powinno się zmienić podczas wyświetlania kodu w porównaniu z poprzednim efektem, ponieważ w skrypcie wszystkie wstawione linijki, które testujemy to komentarze. Jeśli natomiast któraś z linijek jest widoczna w przeglądarce – to należy szukać w niej błędu.

3. Wyprowadzanie treści na ekran przeglądarki. PHP daje kilka możliwości wyprowadzenia danych w okno przeglądarki. Mogą to być: opisy, treść, stałe, zmienne, wartości funkcji i efekty jej działania jak np. rekordy danych. Kolejno wprowadzić do kodu skryptu poszczególne linijki obserwując linijka po linijce jak działają:

```
echo "Witaj, świecie";  
print "Witaj, świecie";  
echo 123;  
$zmienna=10; echo $zmienna; //aby wyświetlić wartość  
zmiennej, trzeba wcześniej ją przypisać  
echo "Witaj, " , "świecie";  
//print i echo można umieszczać w nawiasach:  
print ("Witaj");  
echo "To działa";  
echo 'tak samo jak to';  
echo "Ten łańcuch zawiera ' : pojedynczy cudzysłów" ;  
echo 'Ten łańcuch zawiera " : podwójny cudzysłów' ;
```

efekt: w oknie przeglądarki powinny pojawiać się kolejno poszczególne linie kodu

4. Przypisywanie zmiennym wartości. W poniższym kodzie zawartych jest kilka przykładów przypisania wartości zmiennych. Po przypisaniu proszę **samodzielnie** wyprowadzić wartości na ekran tak by były one wyświetlone każda w nowej linijce.

```
$zmn = 15; // wyprowadzić na ekran a potem zamienić  
wartość zmiennej na niższą:  
$zmn = "i tekst1 gotowy";  
$pojazd = "autobus czerwony";  
$stan = 'mknie';  
$spiew = "$pojazd ulicami mego miasta $stan";  
$ile = 45;  
$info = "$pojazd zabiera $ile osób";
```

efekt: pierwsza linijka to wprowadzenie wartości zmiennej, druga to podmiana liczby na tekst, zmienna *spiew* zawiera połączenie opisu ze zmienną *stan*. Podobnie zmienna *info*

5. Zmienne skalarne. Podobnie jak w poprzednim ćwiczeniu linijka po linijce wypróbować działanie przypisania wartości zmiennym i samodzielnie wyprowadzić je na ekran przeglądarki:

```
$tak = true;  
$test = false;  
$intvar = 65;  
$floatvar = 6.5;  
$floatvar1 = 1.12e3;  
$floatvar2 = 2e-2;  
$suma = $intvar + $floatvar;  
$ciekawe = $tak + $intvar;
```

efekt: na ekranie zostaną zaprezentowane wartości poszczególnych zmiennych w sposób w jaki zostanie określony przez ćwiczącego – on decyduje o wyprowadzeniu na ekran.

6. Definiowanie i używanie stałych

```
define( "pi" , 3.14159);  
$alfa = pi * pi;  
echo $alfa;
```

efekt: wyświetlenie na ekranie zmiennej *alfa* która jest iloczynem dwóch stałych *pi*

7. Operatory arytmetyczne. W skrypcie utworzyć powołać dwie zmienne o wartościach $\$a=10$ oraz $\$b=3$ a następnie zmienną $\$c$ przedstawić jako wynik działań arytmetycznych zawartych w tabelce:

Operator	Przykład	Czynność
+	$\$b = \$a + 3;$	Dodaje wartości
-	$\$b = \$a - 3;$	Odejmuje wartości
*	$\$b = \$a * 3;$	Mnoży wartości
/	$\$b = \$a / 3;$	Dzieli wartości
%	$\$b = \$a \% 3;$	Zwraca resztę
.	$\$t = 'Napis' . \$a;$	Konkatenacja tekstów (łączenie)

przykład:

```
$a=10; $b=3;  
$c=$a+$b;  
echo "suma wartości a i b to: $c <p>";
```

efekt: na ekranie zostaną wyświetlone poszczególne działania zgodnie z formułą w tabeli

8. Operatory przypisania. Podobnie jak w ćwiczeniu poprzednim wypróbować działanie operatorów przypisania zgodnie z tabelą. Przyjąć, że zmienna $\$a$ na początku przyjmuje wartość 10.

Operator	Przykład	Czynność
=	$\$a = 8;$	Przypisuje zmiennej wartość po prawej
+=	$\$a += 3;$	Dodaje do zmiennej wartość po prawej
-=	$\$a -= 5;$	Odejmuje od zmiennej wartość po prawej
.=	$\$a .= "drzew";$	Łączy wartość po prawej z bieżącą

przykład:

```
$a=10;  
echo $a+=3;
```

efekt: wyprowadzone wartości będą zmodyfikowane zgodnie z przykładem;

9. Post i pre inkrementacja. Sprawdzić działanie poszczególnych linii samodzielnie określając typ wyświetlania

```
echo "Postinkrementacja";  
$a = 5;  
echo "Powinno być 5: " . $a++ . "<BR>";  
echo "Powinno być 6: " . $a . "<BR>";
```

```
echo "Preinkrementacja";  
$a = 5;  
echo "Powinno być 6: " . ++$a . "<BR>";  
echo "Powinno być 6: " . $a . "<BR>";
```

efekt: zaobserwować jak i w jakiej kolejności wyświetlana jest zmienna \$a

10. Instrukcja warunkowa. Sprawdzić działanie kodu odświeżając stronę klawiszem F5:

```
$warunek=rand(0,10);  
if ($warunek>5) echo "Zmienna losowana jest większa od 5";  
elseif ($warunek<5) echo "Zmienna losowana jest mniejsza od 5";  
else echo "Zmienna losowana jest dokładnie równa 5";
```

efekt: Serwis będzie za każdym razem generował jedną zmienną \$a z przedziału 0,10 i w zależności od jej wartości informował na stronie o tym jak wygląda wartość losowana.

11. Pętla while. Podobnie jak w poprzednim przykładzie sprawdzić działanie instrukcji poprzez kilkakrotne naciśnięcie klawisza F5 po wprowadzeniu kodu do pliku php:

```
$licznik=1;  
$petlawhile=rand(0,10);  
while ($petlawhile<5)  
{  
echo "Wylosowana liczba jest nadal mniejsza od 5 bo jest to liczba: $petlawhile <br>";  
$licznik++;  
$petlawhile=rand(0,10);  
}
```

```
echo "za $licznik razem wylosowano liczbę większą od 5";
```

efekt: losowanie na stronie liczby i wskazanie za którym razem została wylosowana liczba 5

12. Pętla for. Podobnie jak w poprzednim przykładzie wprowadzić kod do pliku php a następnie sprawdzić jego działanie:

```
$test=rand(0,10);  
for ($licznik=1;$licznik<=10;$licznik++)
```

```
{
$test=rand(0,10);
  if ($test==5)
  {
    echo "za $licznik razem wylosowano liczbę 5<br>";
    break;
  }
else echo "trwa losowanie<br>";
}
if ($licznik==11)
echo "Nie wylosowano liczby 5 we wszystkich 10 próbach";
efekt: kod wykonuje 10 prób losowania i poszukuje liczby 5 informując o stanie aplikacji.
```

13. Przełącznik Switch. Podobnie jak w poprzednim ćwiczeniu przeanalizować poniższy kod, wprowadzić go do pliku php a następnie wykonać. Klawiszem F5 kilkakrotnie wywołać ponowne uruchomienie skryptu aby sprawdzić efekt działania.

```
$test=rand(0,10);
switch($test)
{
case "1":
echo "Działanie jeśli wypadnie 1";
break;

case "5":
echo "Działanie jeśli wypadnie 5";
break;

case "10":
echo "działanie jeśli wupadnie 10";
}
if ($test<>1 && $test <>5 && $test !=10) echo "Nie wypadła żadna z liczb funkcyjnych";
efekt: kod wyświetli komentarze po losowaniu zmiennej $test i sprawdzeniu jej wartości.
```

14. Opracować prostą aplikację, która wykona (za pomocą jednej pętli) działanie losowania dla klienta 6 liczb spośród 49 i zaprezentuje je na stronie jedna pod drugą tak aby pierwsza liczba wyświetlona była jako największa a ostatnia najmniejsza Należy wykorzystać do prezentacji kod HTML.

☺ Powodzenia