

Ćw.1. Obsługa formularzy w PHP

1. Do projektu z poprzedniego laboratorium dodaj nowy plik *formularz.html* z formularzem jak na rys.1. (*HTML File*) oraz nowy plik *odbierz.php* (*PHP Web Page*).
2. W skrypcie *odbierz.php* wyświetl dane przesłane z *formularz.html* (Rys.2). Pamiętaj o ustawieniu odpowiednich atrybutów *action='odbierz.php'* i *method* w znaczniku *form* dokumentu *formularz.html*. (Wykład 1).

UWAGA!

Skrypt, który odczytuje parametry przesłane w żądaniu HTTP, na początek powinien sprawdzać ich istnienie. Do tego celu skorzystaj z funkcji *isset(\$parametr)*, która sprawdza, czy parametr istnieje i czy jego wartość jest różna od *NULL*. Przy pobieraniu danych z pól tekstowych pamiętaj o odpowiednim wykorzystaniu funkcji *trim()* i *htmlspecialchars()*. Fragment kodu z Listingu 1 przedstawia prawidłowe wykorzystanie niezbędnych funkcji.

Listing 1. Przykładowy kod skryptu odbierającego dane z żądania HTTP

```
<div>
<?php
    echo '<h2>Dane odebrane z formularza:</h2>';
    if (isset($_REQUEST['nazw'])&&($_REQUEST['nazw']!="")) {
        $nazwisko = htmlspecialchars(trim($_REQUEST['nazw']));
        echo 'Nazwisko: '.$nazwisko.'<br />';
    }
    else echo 'Nie wpisano nazwiska <br />';

    //pozostałe instrukcje zbierające dane wysłane
    //z formularza w postaci parametrów żądania
    //...
?>
</div>
```

Rys.1. Wygląd formularza z pliku *formularz.html*



Rys.2. Przykładowy efekt działania skryptu *odbierz.php*

Ćw.2. Typy zmiennych w PHP

W kolejnym skrypcie (zadanie2.php) utwórz zmienne o następujących wartościach:

- 134,
- 67.67,
- 1,
- 0,
- true (zmienna logiczna),
- „0” (ciąg znaków),
- „Ala ma kota, a kot ma Alę”,
- [1, 2, 3, 4] (tablica wartości).
- pusta tablica,
- [„zielony”, „czerwony”, „niebieski”],
- obiekt *DateTime* z dzisiejszą datą (http://www.w3schools.com/php/php_date.asp),

- a) Następnie sprawdź na nich działanie następujących funkcji: *is_bool()*, *is_int()*, *is_numeric()*, *is_string()*, *is_array()*, *is_object()*.
- b) Wyświetl poszczególne zmienne (dla tablic użyj pętli *for* oraz funkcji *count()* do liczenia elementów tablicy).
- c) Dodatkowo na tablicach sprawdź funkcje *var_dump()* i *print_r()*.

Ćw.3. Operatory porównania

W skrypcie (zadanie3.php) utwórz zmienną logiczną (*false*) oraz zmienną o wartości 0.

Następnie porównaj te zmienne za pomocą instrukcji warunkowej **if** oraz operatorów „==” oraz „===”.

Ćw.4. Tablice asocjacyjne

W skrypcie (zadanie4.php) utwórz tablicę asocjacyjną, która będzie przechowywała nazwisko i imię studenta dla numerów indeksów użytych jako identyfikatorów (tablica ma mieć długość co najmniej 5 elementów).

Następnie wykonaj następujące zadania:

- a) Wyświetl wszystkie elementy tablicy korzystając z pętli *foreach* w następującym formacie: <nr indeksu> - <nazwisko i imię>.
- b) Wyświetl wszystkie elementy tablicy posortowane wg klucza <nr indeksu>
- c) Wyświetl wszystkie elementy tablicy posortowane wg wartości

- d) Dodaj nowy element do tablicy korzystając z funkcji *array_push()* oraz dodaj nowy element do tabeli korzystając z funkcji *array_merge()*
http://www.w3schools.com/php/php_ref_array.asp.
- e) Wyświetl wszystkie elementy (klucze oraz wartości) tabeli w losowej kolejności (*shuffle()*) umieszczając je w tabeli HTML.

Ćw.5. Wielowymiarowe tablice asocjacyjne

W skrypcie (zadanie5.php) stwórz tabelę asocjacyjną, która jako klucze będzie przechowywała nr indeksu studenta, a jako wartość tablicę z jego ocenami.

Oblicz średnią ocen dla każdego nr indeksu.