

Odczytywanie wartości z kontrolek formularzy

W formularzach HTML często korzystamy z pól wyboru, takich jak pola radio, checkboxy i listy wyboru. PHP pozwala nam na odczytanie wartości tych pól, zarówno gdy użytkownik może wybrać jedną opcję (np. pola radio), jak i gdy formularz pozwala na wybór wielu wartości (np. checkboxy).

W tym artykule nauczymy się, jak odczytać dane wysłane przez użytkownika z popularnych kontrolek, a szczególnie z list wyboru

`<select>` oraz pól typu radio i checkboxów – zarówno jako pojedyncze zmienne, jak i jako tablice.

Odczyt z klasycznych inputów (text, number, password, email etc.) jest z natury prosty – patrz lekcja: [Obsługa formularzy – metody POST i GET](#), natomiast listy wyboru i pola “zaznaczalne” (radio, checkbox) czasem sprawiają nam trudności – niech ta lekcja pomoże nam komfortowo poradzić sobie z nimi na egzaminie!

Odczyt wartości z pól typu radio

W formularzach HTML bardzo często chcemy, aby użytkownik wybrał dokładnie jedną z dostępnych opcji. Do takiego celu doskonale nadają się kontrolki typu

`radio`. Każdy z takich przycisków jest oddzielnym elementem typu `<input type="radio">`, ale wszystkie mają wspólną nazwę – a więc ten sam atrybut `name`. Dzięki temu przeglądarka wie, że należy je traktować jako jedną grupę wyboru (czyli zaznaczenie jednej opcji automatycznie odznacza inne).

Przykład formularza z wyborem języka programowania:

```
<form method="post">
  <label><input type="radio" name="jezyk" value="PHP"> PHP</label><br>
  <label><input type="radio" name="jezyk" value="Python"> Python</label><br>
  <label><input type="radio" name="jezyk" value="JavaScript"> JavaScript</label><br>
  <input type="submit" value="Wyślij">
</form>
```

W powyższym kodzie użytkownik może wybrać dokładnie jedną z trzech opcji: PHP, Python lub JavaScript. Pora teraz zrealizować obsługę tego formularza w PHP:

```
<?php
if (isset($_POST["jezyk"])) {
    $wybrany_jezyk = $_POST["jezyk"];
    echo "Wybrany język programowania: " . $wybrany_jezyk;
} else {
    echo "Nie wybrano języka!";
}
?>
```

Analiza działania powyższego kodu PHP:

- Używamy funkcji `isset()`, aby upewnić się, że użytkownik faktycznie wybrał jakąś opcję i przesłał formularz.
- Jeśli tak – odczytujemy wartość z pola `$_POST["jezyk"]` i zapisujemy ją do zmiennej `$wybrany_jezyk`.
- Na końcu wypisujemy komunikat informujący, co zostało wybrane.
- Jeśli użytkownik nie zaznaczył żadnej opcji i kliknął “Wyślij” – zobaczymy komunikat: *Nie wybrano języka!*

Dzięki temu mechanizmowi możemy stworzyć np. formularze ankietowe, pytania jednokrotnego wyboru, czy konfiguratory z domyślnym wyborem. Jak ustawić domyślnie zaznaczoną opcję? Otóż wystarczy w odpowiednim polu

`<input>` dodać atrybut `checked` . Przykład:

```
<form method="post">
  <label><input type="radio" name="jezyk" value="PHP" checked> PHP</label><br>
  <label><input type="radio" name="jezyk" value="Python"> Python</label><br>
  <label><input type="radio" name="jezyk" value="JavaScript"> JavaScript</label><br>
  <input type="submit" value="Wyślij">
</form>
```

W powyższym przykładzie opcja "PHP" będzie zaznaczona automatycznie – chyba że użytkownik wskaże inną. To proste rozwiązanie działa zawsze – wystarczy tylko pamiętać, że:

- Atrybut `checked` dodajemy do jednego (i tylko jednego) z pól radio z daną nazwą w atrybucie `name` .
- W przypadku formularzy wysyłanych metodą `GET` również możemy użyć domyślnego wyboru, ale będzie on przesłany tylko po kliknięciu przycisku.

To rozwiązanie sprawdza się idealnie np. gdy chcemy wskazać najczęściej wybieraną odpowiedź jako domyślną – użytkownik może ją zaakceptować lub zmienić.

Odczytywanie wartości z listy rozwijanej

Listy rozwijane w formularzach HTML to również bardzo wygodny sposób, aby użytkownik wybrał jedną opcję spośród kilku dostępnych. Z punktu widzenia odczytu wybranej wartości w PHP wygląda to bardzo podobnie jak odczyt wartości z pola tekstowego czy radio – różni się to raczej tylko wyglądem kontrolki w formularzu.

Przykład poniżej pokazuje jak utworzyć formularz z listą rozwijaną zawierającą trzy miasta, a następnie odebrać dane w PHP:

```
<form method="post">
  <label>Wybierz miasto:</label>
  <select name="miasto">
    <option value="Warszawa">Warszawa</option>
    <option value="Kraków">Kraków</option>
    <option value="Gdańsk">Gdańsk</option>
  </select>
  <input type="submit" value="Wyślij">
</form>
```

Kod PHP obsługujący formularz wygląda następująco:

```
<?php
if (isset($_POST["miasto"])) {
    $wybrane_miasto = $_POST["miasto"];
    echo "Wybrane miasto: " . $wybrane_miasto;
} else {
    echo "Nie wybrano miasta!";
}
?>
```

Analiza kodu:

- W formularzu używamy kontrolki `<select>`, która pozwala wybrać tylko jedną opcję.
- Atrybut `name="miasto"` powoduje, że po przesłaniu formularza wartość zostanie umieszczona w tablicy `$_POST["miasto"]`.
- W PHP sprawdzamy, czy formularz został przesłany i czy istnieje indeks `"miasto"` – używamy w tym celu funkcji `isset()`.
- Jeśli tak – wypisujemy informację o wybranym mieście. W przeciwnym razie – pokazujemy komunikat, że nic nie wybrano.

Jak ustawić domyślnie zaznaczoną opcję?

Jeśli chcesz, aby po wejściu na stronę jedna z opcji była zaznaczona automatycznie – wystarczy dodać atrybut

`selected` do wybranej opcji:

```
<select name="miasto">
  <option value="Warszawa" selected>Warszawa</option>
  <option value="Kraków">Kraków</option>
  <option value="Gdańsk">Gdańsk</option>
</select>
```

Dzięki temu użytkownik zobaczy od razu wybraną opcję – ale może ją oczywiście zmienić przed wysłaniem formularza.

Odczyt zaznaczonych checkboxów do tablicy

Pola typu

`checkbox` pozwalają użytkownikowi zaznaczyć **wiele opcji** jednocześnie. To idealne rozwiązanie, gdy np. chcemy zapytać o zainteresowania, umiejętności czy ulubione dania. Z punktu widzenia PHP, wszystkie zaznaczone pola checkbox trafiają do tablicy – o ile odpowiednio oznaczymy nazwę (czyli dodamy nawias kwadratowy do `name`).

Przykład formularza z checkboxami:

```
<form method="post">
  <p>Wybierz swoje zainteresowania:</p>
  <label><input type="checkbox" name="zainteresowania[]" value="Muzyka">
Muzyka</label><br>
  <label><input type="checkbox" name="zainteresowania[]" value="Sport">
Sport</label><br>
  <label><input type="checkbox" name="zainteresowania[]" value="Programowanie">
Programowanie</label><br>
  <input type="submit" value="Wyślij">
</form>
```

Odbiór danych w PHP:

```
<?php
if (isset($_POST["zainteresowania"])) {
    $wybrane = $_POST["zainteresowania"];
    echo "Twoje zainteresowania:<br>";
    foreach ($wybrane as $zainteresowanie) {
        echo "- " . htmlspecialchars($zainteresowanie) . "<br>";
    }
}
```

```

    }
} else {
    echo "Nie wybrano żadnych zainteresowań!";
}
?>

```

Co się tutaj dzieje?

- Każdy checkbox ma taką samą nazwę `name="zainteresowania[]"` – nawiasy kwadratowe oznaczają, że wartości mają trafić do tablicy.
- Użytkownik może zaznaczyć dowolną liczbę opcji – 0, 1, 2 lub wszystkie.
- Po przesłaniu formularza sprawdzamy, czy tablica `$_POST["zainteresowania"]` istnieje – to robimy funkcją `isset()`.
- Jeśli tak – przetwarzamy ją pętlą `foreach` i wypisujemy każdą wybraną wartość.
- Dodatkowo użyto funkcji `htmlspecialchars()`, aby zabezpieczyć kod przed ewentualnym wstrzyknięciem HTML.

Domyślnie zaznaczone checkboxy

Jeśli chcemy, aby dana opcja była domyślnie zaznaczona przy pierwszym otwarciu strony, dodajemy atrybut

`checked` :

```
<input type="checkbox" name="zainteresowania[]" value="Muzyka" checked> Muzyka
```

Odczyt wielu wartości z listy select multiple

W HTML wybór wielokrotny zrealizujemy także np. za pomocą atrybutu

`multiple` w kontrolce `<select>`. Co ważne – jeśli przewidujemy wybór wielu opcji na liście, to w atrybucie `name` tej kontrolki również (analogicznie jak to było przy checkboxach) dodajemy nawias kwadratowy – np. `name="kolory[]"`. Dzięki temu PHP wie, że przesłane zostaną dane w formie tablicy (wybór wielokrotny).

Przykład formularza:

```

<form method="post">
    <label>Wybierz ulubione kolory:</label>
    <select name="kolory[]" multiple>
        <option value="Czerwony">Czerwony</option>
        <option value="Zielony">Zielony</option>
        <option value="Niebieski">Niebieski</option>
    </select>
    <input type="submit" value="Wyślij">
</form>

```

Po przesłaniu formularza dane są przetwarzane przez następujący kod PHP:

```

<?php
if (isset($_POST["kolory"])) {
    $wybrane_kolory = $_POST["kolory"];
}

```

```

    echo "Wybrane kolory:<br>";
    foreach ($wybrane_kolory as $kolor) {
        echo "- " . $kolor . "<br>";
    }
} else {
    echo "Nie wybrano kolorów!";
}
?>

```

Wyjaśnienie kodu krok po kroku:

- Użytkownik może zaznaczyć dowolną liczbę kolorów na liście (bo używamy atrybutu `multiple`).
- Atrybut `name="kolory[]"` sprawia, że dane trafiają do PHP jako tablica – każda wybrana wartość będzie jednym z elementów tej tablicy.
- W PHP sprawdzamy, czy indeks `"kolory"` istnieje – czyli czy użytkownik coś wybrał.
- Jeśli tak – przechowujemy wynik w zmiennej `$wybrane_kolory` i przechodzimy po jej elementach pętlą `foreach`, wypisując każdy kolor.
- Jeśli nic nie zostało wybrane – wypisujemy komunikat *Nie wybrano kolorów!*

Jak zaznaczyć domyślnie kilka opcji?

Jeśli chcemy, aby niektóre opcje były zaznaczone od razu – dodajemy do nich atrybut

`selected` :

```

<option value="Czerwony" selected>Czerwony</option>
<option value=

```

Podsumowanie wiedzy

W tej lekcji nauczyliśmy się, jak prawidłowo odbierać dane przesyłane przez użytkownika za pomocą różnych typów kontrolek formularza – takich jak pola radio, checkboxy oraz listy rozwijane (zarówno pojedynczego jak i wielokrotnego wyboru). Wszystkie przesyłane dane trafiają do specjalnych tablic globalnych PHP:

`$_POST` lub `$_GET` – w zależności od ustawionej metody w formularzu.

- Pola typu radio służą do wyboru jednej opcji – wszystkie muszą mieć wspólną nazwę (`name`), aby były traktowane jako grupa. Odczytujemy wartość jako pojedynczą zmienną, np. `$_POST["jezyk"]`.
- Listy rozwijane select bez atrybutu `multiple` działają tak samo – pozwalają użytkownikowi wybrać jedną opcję, którą odczytujemy jako jedną zmienną po stronie PHP.
- Checkboxy oraz listy select z atrybutem `multiple` pozwalają zaznaczyć wiele opcji – dane te przesyłane są do PHP jako tablica, dlatego musimy dodać nawiasy kwadratowe do `name` w polu formularza, np. `name="zainteresowania[]"`.
- Po stronie PHP sprawdzamy, czy użytkownik przesłał dane, np. funkcją `isset()`, a następnie możemy użyć pętli – np. `foreach` do przetwarzania tablicy z wieloma wartościami.
- Aby ustawić domyślnie wybrane opcje:
 - dla radio i checkboxów – dodajemy atrybut `checked`

- dla select – dodajemy atrybut `selected` do odpowiednich opcji

Rozumienie mechanizmu przesyłania i odbioru danych formularzowych to absolutna podstawa na egzaminie INF.03. Dzięki tej lekcji potrafimy nie tylko odczytywać dane z formularzy, ale też przetwarzać je bezpiecznie i w zależności od kontekstu – jako pojedyncze wartości lub całe tablice.