

PLIKI COOKIES

Nietyksza Filip i Jakub Sienkiewicz

Co to są Ciasteczka i po co się ich używa?

Ciasteczka (ang. cookies) to niewielkie pliki, zapisywane i przechowywane na Twoim komputerze lokalnym lub innym urządzeniu, podczas gdy odwiedzasz różne strony WWW w sieci Internet. Plik „cookie” zazwyczaj zawiera:

- podstawowe informacje na temat nazwy strony WWW, z której pochodzi,
- „długość życia” ciasteczka — czyli jaki ustawiony jest czas jego istnienia na Twoim komputerze,
- przypadkowo wygenerowany unikalny numer służący do identyfikacji przeglądarki, z jakiej następuje połączenie ze stroną WWW.

Pliki „cookies” (tzw. ciasteczka) stanowią dane informatyczne, w szczególności pliki tekstowe, które przechowywane są w urządzeniu końcowym użytkownika strony WWW. Ciasteczka używane są najczęściej w celu optymalizacji procesu korzystania ze stron WWW. Stosowane są również w celu gromadzenia danych statystycznych, które pozwalają identyfikować sposób korzystania użytkowników ze stron internetowych. Daje to możliwość późniejszego ulepszania struktury i zawartości strony WWW.

Tworzenie plików cookie w PHP

```
setcookie($nazwa, $wartosc, $koniec, $sciezka, $domena, $bezpieczne);
```

Wymagany jest tylko parametr nazwy. Wszystko inne jest opcjonalne. Ta funkcja musi pojawić się przed znacznikiem <html>.

```
Przykład: setcookie('oddano_glos', '1', time()+3600*24);
```

Sprawdzanie obecności ciasteczka

Wiesz, jak ustawić ciasteczko, jego wartość oraz datę wygaśnięcia. Czas teraz na sprawdzenie, czy ciastko istnieje. Każde ciasto jest przechowywane w tablicy globalnej \$_COOKIE. Sprawdźmy, czy istnieje cookie o nazwie „aktywacja”:

```
<?php
```

```
if(isset($_COOKIE['aktywacja']))
```

```
    echo "Ciasteczko istnieje";

else

    echo "Brak ciasteczka o nazwie aktywacja";

?>
```

Przykład:

```
1  <?php
2  $nazwa_ciasteczka = "imie";
3  $wartosc_ciasteczka = "Janusz";
4  $waznosc_ciasteczka = time() + (60 * 60 * 24 * 30); // 30 dni
5  $zasieg_ciasteczka = "/";
6  setcookie($nazwa_ciasteczka, $wartosc_ciasteczka, $waznosc_ciasteczka, $zasieg_ciasteczka);
7
8  if(!isset($_COOKIE[$nazwa_ciasteczka])){
9      echo "Ciasteczko o nazwie '" . $nazwa_ciasteczka . "' nie zostało ustawione!";
10 }else{
11     echo "Ciasteczko '" . $nazwa_ciasteczka . "' zostało ustawione!<br>";
12     echo "Wartość ciasteczka: " . $_COOKIE[$nazwa_ciasteczka];
13 }
14 ?>
15
```

Usuwanie pliku cookie

Aby usunąć ciasteczko wystarczy użyć funkcji `setcookie()` i należy ustawić dla niego ujemną wartość czasową.

```
np. setcookie("user", "", time() - 3600);
```

SESJA

Co to jest sesja w PHP?

Mechanizm sesji wykorzystywany jest do przechowywania informacji potrzebnych w trakcie przeglądania strony. Wykorzystywany jest chociażby do zapamiętywania informacji o zalogowaniu do serwisu. Dzięki temu przemieszczając się po poszczególnych podstronach nie musimy za każdym razem się logować. Sesja utrzymywana jest przez przeglądarkę do momentu jej zamknięcia. Chcąc zapamiętać informację po

stronie użytkownika dłużej niż tylko do momentu zamknięcia przeglądarki wykorzystujemy chociażby mechanizm ciasteczek.

Rozpoczęcie sesji

Aby poinformować przeglądarkę, że dana podstrona wykorzystuje mechanizm sesji musimy na samym początku pliku umieścić następującą linię kodu:

```
session_start();
```

Linia ta informuje o rozpoczęciu (kontynuowaniu) sesji na danej podstronie.

Tworzenie zmiennych sesyjnych

Utworzyliśmy sesję, możemy zatem przejść do tworzenia zmiennych sesyjnych, w których zapiszemy potrzebne nam dane. I tak przykładowo:

```
$_SESSION["kolor"] = "niebieski";
```

Utworzyliśmy zmienną kolor o wartości niebieski.

Odczyt wartości zmiennych sesyjnych

Przejdźmy zatem do odczytania wartości wcześniej utworzonej zmiennej. Wartości te przechowywane są w tablicy `$_SESSION`.

Wypiszmy najpierw całą zawartość tablicy `$_SESSION`:

```
print_r($_SESSION);
```

Aby wykorzystać wartość konkretnej zmiennej, podobnie jak w pracy ze zwykłymi tablicami odwołujemy się do komórki o konkretnym adresie. W naszym przypadku:

```
echo $_SESSION['kolor'];
```

Czyszczenie / kończenie sesji

Aby uwolnić wszystkie zarejestrowane w sesji zmienne wykorzystamy:

```
session_unset();
```

Aby usunąć wszystkie dane związane z sesją użyjemy:

session_destroy();

Przykład:

```
1  <?php
2  session_start();//Rozpoczęcie sesji
3  $_SESSION["kolor"] = "niebieski";//Tworzenie zmiennych sesyjnych
4  echo' <br>';
5  print_r($_SESSION);//Odczyt wartości zmiennych sesyjnych
6  echo' <br>';
7  echo' <br>';
8  echo $_SESSION['kolor'];
9
10 session_unset();//Czyszczenie
11 session_destroy();//kończenie sesji
12
13 ?>
```