

În cele ce urmează va fi realizată o aplicație simplă care păstrează istoricul căutărilor efectuate de utilizatori. Ideea poate fi extinsă la alte aplicații similare, de exemplu pentru a păstra istoricul ultimelor pagini vizitate, al comentariilor recente efectuate, etc. În toate aceste situații mecanismul de funcționare este același, diferă doar ce anume este persistat.

Aplicația este una simplă și constă într-un singur fișier PHP. Are la bază formularele, întrucât necesită transmiterea unor date de la utilizatori către serverul web.

Mecanismul de funcționare

Pentru a persista datele pe parcursul navigării (de la o afișare a paginii la alta) această aplicație folosește mecanismul de sesiuni.

În PHP, variabilele sunt ne-persistente (stateless). Asta înseamnă că ele există și au o valoare doar cât timp se procesează scriptul în care sunt definite. La terminarea execuției, valorile se pierd.

Astfel, la o nouă procesare a aceluiași script (spre exemplu la un refresh al paginii, când codul-sursă se execută din nou) toate variabilele se reinitializează și nu "țin minte" valorile de dinainte. Asta înseamnă că o navigare printr-un site cu mai multe pagini PHP constă, de fapt, într-o serie de execuții independente ale scripturilor și, în mod normal, o pagină "nu știe" ce variabile s-au definit la execuțiile anterioare sau pe alte pagini accesate.

Acesta este comportamentul implicit. Există totuși și un mecanism de a persista date de-a lungul navigării de la o pagină la alta, iar acest mecanism poartă numele de sesiune.

te

Formularul de căutare

```
<form method="post" action="">
```

```
    Cuvant cheie <input type="text" name="keyword" value="" />
```

```
    <input type="submit" name="trimite" value="Cauta" />
```

```
    <input type="reset" value="Curata" />
```

```
</form>
```

Codul PHP de persistare a unei căutări

```
# pornesc o sesiune
```

```

session_start();

// ...

# preiau datele existente din sesiune

if( isset( $_SESSION['cautari'] ) ) {

    $istoric = $_SESSION['cautari']; // istoric exista

} else {

    $istoric = array(); // initializez un vector gol

}

# verific daca a fost facut submit la formular

if( isset($_POST) && !empty($_POST['keyword']) ) {

    // adaug termenul cautat

    array_push($istoric, $_POST['keyword']);

    // stochez totul inapoi pe sesiune

    $_SESSION['cautari'] = $istoric;

    // e posibil sa fac alte procesari aici

    // de exemplu sa afisez lista de rezultate

}

```

Codul PHP de afisare a istoricului cautarilor

```

# verific daca exista istoric salvat

if( empty( $istoric ) ) {

    echo 'Nu ati cautat nimic pana acum';

} else {

```

```
# cum exista deja un istoric, il afisez  
  
echo '<ul>';  
  
# folosesc o structura repetitiva care parcurge vectorul si afiseaza datele lui  
  
foreach($istoric as $termen) {  
    echo "<li>$termen</li>";  
}  
  
# afisez cate elemente exista  
  
echo '</ul><p>', 'Cautari efectuate: ', count( $istoric ), '</p>';  
  
}
```