

*Web-dasturlashga kirish fani
bo'yicha elektron resurs uchun
taqdimot materiallari*

16-17-Ma'ruza. PHPda global
o'zgaruvchilar.

Reja:

1. PHPda global o'zgaruvchilar: POST, GET, COOKIE, SESSION, SERVER, REQUEST, ENV, FILES.
2. O'zgaruvchilarning ko'rinish sohalari.
3. Ma'lumotlarni filtirlash.

1.1. PUT va POST usullari

HTTP PUT va HTTP POST usullari fayllarni serverga yuklash uchun mo'ljallangan.

Web-serverdagi ma'lumot bilan ishlash uchun HTTP Protokolda uchta usul mavjud: GET, PUT va POST. GET usuli Web-sahifalarni olish uchun qo'llanadi, bunda forma hamma o'zgaruvchilari URL ga uzatiladi. Ko'pgina Web-serverlarda URL maksimal uzunligi cheklanganligi uchun (odatda 1024 dan oshmaydi), katta xajmdagi ma'lumotlarni uzatishda GET usulini qo'llab bo'lmaydi.

PUT usuli serverdagi ma'lumotni Yangilash uchun qo'llanadi va HTTP PUT so'rov natijasi serverda saqlanishini talab qiladi. So'rov quyidagi ko'rinishga ega:
PUT /path/filename.html HTTP/1.1

Bu holda Web-server bu so'rov natijasini Web-serverning URL nomlar fazosida /path/filename.html ko'rinishda saqlashi lozim. Ko'zda tutilgan bo'yicha Web-server o'zi bunday so'rovlarni bajarmaydi, balkim ularni qayta ishlash uchun CGI-stsenariy beradi. Apache da PUT so'rovlarni qayta ishlovchi stsenariy tayinlash uchun httpd.conf fayldagi Script direktivasini o'zgartirish mumkin, masalan quyidagicha:

Script PUT /cgi-bin/put.cgi

Bu shuni bildiradiki PUT-so'rovlarni CGI-script put.cgi qayta ishlaydi.

Odatda fayllarni serverga yuklash uchun HTTP POST usulidan foydalaniladi. Bu usul formadan ma'lumotlar katta xajmini uzatishga imkon beradi va so'rov tanasidagi hamma forma o'zgaruvchilarni saqlaydi.

PHPда суперглобал ўзгарувчилар

PHPда суперглобал ўзгарувчиларга сайт тузилмасидаги файлларнинг ихтиёрий жойидан мурожаат қилиш имконияти мавжуд бўлади:

- `$GLOBALS`
- `$_SERVER`
- `$_REQUEST`
- `$_POST`
- `$_GET`
- `$_FILES`
- `$_ENV`
- `$_COOKIE`
- `$_SESSION`

PHPда суперглобал ўзгарувчилар

- ✓ `$_SERVER['PHP_SELF']` – скрипт бажарилаётган файл номини аниқлайди;
- ✓ `$_SERVER['SERVER_ADDR']` – сервер IP манзилини аниқлайди;
- ✓ `$_SERVER['SERVER_NAME']` – сервер домен номини аниқлайди;
- ✓ `$_SERVER['SERVER_SOFTWARE']` – сервер идентификациясини (Apache, PHP версиялари) аниқлайди;
- ✓ `$_SERVER['HTTP_USER_AGENT']` – саҳифани очишда фойдаланилган веб броузер номини аниқлайди;
- ✓ `$_SERVER['REQUEST_METHOD']` – саҳифага кириш сўров усулини аниқлайди;

PHPда суперглобал ўзгарувчилар

- ✓ `$_REQUEST` – HTML формада юборилган маълумотларни жамлаш учун ишлатилади;
- ✓ `$_POST` – HTML формада `method="post"` усулида юборилган маълумотларни жамлаш учун ишлатилади;
- ✓ `$_GET` – HTML формада `method="get"` усулида юборилган маълумотларни жамлаш учун ишлатилади.



- PHP \$_POST - bu PHP super global o'zgaruvchisi bo'lib, u HTML formasini yuborganidan so'ng forma ma'lumotlarini to'plash uchun ishlatiladi = "post". \$_POST o'zgaruvchini o'tkazish uchun ham keng qo'llaniladi.:

```
<html>
<body>
```

```
<form method="post" action="<?php echo $_SERVER['PHP_SELF'];?>">
  Name: <input type="text" name="fname">
  <input type="submit">
</form>
```

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  // collect value of input field
  $name = $_POST['fname'];
  if (empty($name)) {
    echo "Name is empty";
  } else {
    echo $name;
  }
}
?>
```

```
</body>
</html>
```



- PHP \$_GET - bu PHP super global o'zgaruvchisi, u HTML shaklini yuborganidan keyin form ma'lumotlarini yig'ish uchun ishlatiladi = "get":

```
<html>
```

```
<body>
```

```
<?php
```

```
echo "Study " . $_GET['subject'] . " at " . $_GET['web'];
```

```
?>
```

```
</body>
```

```
</html>
```


PHPда суперглобал ўзгарувчилар

- `$ _SERVER` - bu PHP super o'zgaruvchisi, unda sarlavhalar, yo'llar va skriptlarning joylashuvi haqida ma'lumot mavjud:

```
<?php
echo $_SERVER['PHP_SELF'];// joriy fayl nomini qaytaradi
echo "<br>";
echo $_SERVER['SERVER_NAME'];// xost-server nomini qaytaradi
echo "<br>";
echo $_SERVER['HTTP_HOST'];//Xost nomini joriy so'rovdan qaytaradi
echo "<br>";
echo $_SERVER['HTTP_REFERER']; Joriy sahifaning to'liq URL
manzilini qaytaradi
echo "<br>";
echo $_SERVER['SCRIPT_NAME'];//Amaldagi skript yo'lini qaytaradi
?>
```



PHPда суперглобал ўзгарувчилар

- `$_SESSION` – PHP да маълумотни бир нечта саҳифа орасида сақлаш учун ишлатилади, `$_SESSION` ўзгарувчисидаги маълумотлар серверда сақланади:

```
<?php
// Start the session
session_start();
?>
<html>
<body>
<?php
$_SESSION["user_id"] = 157;
$_SESSION["role"] = "editor";
echo $_SESSION["user_id"]; //157
?>
</body>
</html>
```



- PHP `$_REQUEST` - bu PHP super global o'zgaruvchisi, u HTML formasini yuborganidan keyin ma'lumotlarni yig'ish uchun ishlatiladi:

```
<html>
<body>
<form method="post" action="<?php echo $_SERVER['PHP_SELF'];?>">
  Name: <input type="text" name="fname">
  <input type="submit">
</form>
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  // collect value of input field
  $name = $_REQUEST['fname'];
  if (empty($name)) {
    echo "Name is empty";
  } else {
    echo $name; } }
?>
</body>
</html>
```

1.3. Cookie texnologiyasini o'rnatish

Cookie — klient kompyuterida saqlanuvchi va u har safar serverga murojaat qilayotganda web-serverga yuboriladigan matn satridir. Shu tariqa ma'lumotlar turli skriptlar aro formalar yoki URL adreslarsiz uzatilishi mumkin. Cookie qiymatini o'rnatish uchun quyidagi sintaksisdan iborat bo'lgan setcookie funksiyasidan foydalaniladi:

```
bool setcookie (string name [, string value [, int  
expire [, string  
path [, string domain [, bool securej]]]])
```

Funksiya klient kompyuterida saqlanuvchi cookie larni tasnif etadi. Quyida ushbu funksiya parametrlarining tasnifi ketirilgan:

- name. Cookie ning nomi.
- value. Cookie ning qiymati.
- expire. Cookie ning amal qilish muddati. Agar uberilsa, u tugagach cookie o'chirib tashlanadi. Agar ko'rsatilmasa, cookie brauzer oynasi yopilgandan so'ng o'chirib tashlanadi
- path. Serverdagi cookie ga ruxsat etilgan adres.
- domain. cookie ga ruxsat etilgan domen.
- secure. HTTPS protokol orhali bog'lanishda cookie ning havfsizlik belgisi. Standart holatda cookie HTTPS uchun ham HTTP dagi kabi ishlaydi.

Cookie texnologiyasini o'qish

Cookie qiymati o'rnatilganidan so'ng, sahifa qayta yuklanmagunicha darhol undagi skript uchun cookie ga ruxsat mavjud bo'lmaydi. Chunki cookie foydalanuvchi kompyuterida saqlanadi va brauzer orhali web-serverga jo'natiladi. Bundan tashqari, cookie qachonki uning domeni server domeni bilan mos tushgandagina jo'natiladi.

Cookie ga ruxsatni olish uchun maxsus `$_COOKIE` superglobal massividan foydalaniladi. Massivning qiymati sifatida oldin ishlatilgan cookie nomi olinadi. Massiv odatda skript yuklanayotgan vaqtda `$_GET`, `$_POST` va `$_REQUEST` massivlari bilan avtomatik tarzda to'ldiriladi

Cookie dan foydalanishdan oldin uning qiymati o'rnatilganligiga ishonch hosil qiling. Buning uchun isset funksiyasidan foydalanish juda qulay. Quyidagi misolda message nomli cookie ning qiymati tekshirilishi va ko'rsatilishi keltirilgan.

Misol:

```
<HTML>
  <HEAD> <TITLE> Cookie ning qiymatini o'qish <\TTITLE> <\HEAD>
  <BODY>
    <CENTER> a
    <N1> Cookie ning qiymatini o'qish <\H1>
      Cookie ning qiymati:
    <?php
      if (isset ($_COOKIE ['message']))
      {
        echo Cookie ning qiymati:' . $_COOKIE ['message' ] ;
      }
    else
    {
      echo 'Cookie o'rnatilmagan' ;
    }
    ?>
    <\CENTER;
  <BODY>
<\HTML>
```

Cookie massivlarda ham tashkil etilgan bo'lishi mumkin. Masalan quyida uchta cookie o'rnatilgan:

- `setcookie ("cookie[one]" , "Bugun");`
- `setcookie ("cookie[two] " , "Hayot");`
- `setcookie ("cookie[three] " , "go'zal!") ;`

Natijada `$_COOKIE['cookie']` massiv qiymatlari quyidagi tarzda bosmaga chiqarilishi mumkin:

```
if (isset ($_COOKIE ['cookie'])) {  
    foreach ($_COOKIE['cookie'] as $data)  
    {  
        echo "$data <BR>";  
    }  
    ...  
}
```


Sessiyalarni o'rnatish

- Sessiya - bu bir nechta sahifalarda ishlatiladigan ma'lumotni (o'zgaruvchida) saqlash usuli.
- Cookie-dan farqli o'laroq, ma'lumotlar foydalanuvchilar kompyuterida saqlanmaydi.

PHP sessiyasi nima?

- Ilova bilan ishlashda siz uni ochasiz, ba'zi o'zgarishlar kiritasiz va keyin yopasiz. Bu sessiyaga o'xshaydi. Kompyuter sizning kimligingizni biladi. Ilovani qachon boshlaganingizni va qachon tugaganingizni biladi. Ammo Internetda bitta muammo bor: veb-server sizning kimligingizni va nima qilayotganingizni bilmaydi, chunki HTTP manzili holatini saqlab qolmaydi.
- Sessiya o'zgaruvchilari ushbu muammoni bir nechta sahifalarda foydalanuvchi ma'lumotlarini saqlash orqali hal qilishadi. Odatiy bo'lib, sessiya o'zgaruvchilari foydalanuvchi brauzerni yopguncha davom etadi.
- Shunday qilib; Sessiya o'zgaruvchilari bitta foydalanuvchi haqida ma'lumotga ega va bitta dasturdagi barcha sahifalar uchun mavjud.

PHP sessiyasini boshlash

- session_start() Funktsiya bilan sessiya boshlanadi .
- Sessiya o'zgaruvchilari PHP global o'zgaruvchisi bilan o'rnatiladi: \$_SESSION.
- ```
<?php
// Start the session
session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// Set session variables
$_SESSION["favcolor"] = "green";
$_SESSION["favanimal"] = "cat";
echo "Session variables are set.";
?>
</body>
</html>
```

## **PHP sessiyasining o'zgaruvchan qiymatlarini olish**

- Keyinchalik, biz "demo\_session2.php" deb nomlangan boshqa sahifani yaratamiz. Ushbu sahifadan biz birinchi sahifada o'rnatgan sessiya ma'lumotlariga kiramiz ("demo\_session1.php").
- E'tibor bering, sessiya o'zgaruvchilari har bir yangi sahifaga alohida o'tkazilmaydi, aksincha ular har bir sahifaning boshida biz ochgan sessiyadan olinadi ( session\_start()).
- Shuni ham unutmangki, barcha sessiya o'zgaruvchilari qiymatlari global \$ \_SESSION o'zgaruvchisida saqlanadi:

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// Echo session variables that were set on previous page
echo "Favorite color is " . $_SESSION["favcolor"]
. "
";
echo "Favorite animal is " . $_SESSION["favanimal"]
. ".";
?>
</body>
</html>
```

- PHP sessiyasining o'zgaruvchisini o'zgartirish

- `<?php`

```
session_start();
```

```
?>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<?php
```

```
// to change a session variable, just overwrite it
```

```
$_SESSION["favcolor"] = "yellow";
```

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

```
?>
```

```
</body>
```

```
</html>
```

## PHP sessiyasini yo'q qilish

- Barcha global majlisi parametrlarga olib tashlash va sessiya, foydalanish yo'q qilish `session_unset()` va `session_destroy()`:

- ```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// remove all session variables
session_unset();
// destroy the session
session_destroy();
?>
</body>
</html>
```

PHP sessiyasini yo'q qilish

- Barcha global majlisi parametrlarga olib tashlash va sessiya, foydalanish yo'q qilish `session_unset()` va `session_destroy()`:

- ```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// remove all session variables
session_unset();
// destroy the session
session_destroy();
?>
</body>
</html>
```



## O'zgaruvchilarning ko'rinish sohalari.

- O'zgaruvchilar funksiyalarda lokal ko'rinish sohasiga ega. Bu shuni bildiradiki hatto lokal va tashqi o'zgaruvchilar bir xil nomga ega bo'lsa ham, lokal uzgaruvchi o'zgarisha tashqi o'zgaruvchiga ta'sir qilmaydi.

- `<? function get_sum() {`
- `$var = 5; // lokal o'zgaruvchi`
- `echo $var; }`
- `$var = 10; // global o'zgaruvchi`
- `get_sum(); // chiqaradi 5 (lokal o'zgaruvchi)`
- `echo("<br>$var"); // chiqaradi 10 (global o'zgaruvchi)`
- `?>`

- Lokal o'zgaruvchini global qilish mumkin, agar uning nomi oldidan global kalit so'zi ko'rsatilsa. Agar tashqi o'zgaruvchi global sifatida e'lon qilingan bo'lsa, unga ixtiyoriy funksiyadan murojaat qilish mumkin:

- <?
- function get\_sum() {
- global \$var;
- \$var = 5; // izmenyaem globalnuyu peremennuyu
- echo \$var; }
- \$var = 10;
- echo("\$var<br>"); // chiqaradi 10
- get\_sum(); // chiqaradi 5 (globalnaya o'zgaruvchi izmenena)
- ?>

Global o'zgaruvchilarga `$GLOBALS` assotsiativ massiv orqali murojaat qilish mumkin:

- <?
- `function get_sum() {`
- `$GLOBALS["var"] = 20; // global parametrlar o'zgaradi $var`
- `echo($GLOBALS["var"]);`
- `}`
- `$var = 10;`
- `echo("$var<br>"); // chiqaradi 10`
- `get_sum(); // chiqaradi 20 (global o'zgaruvchi o'zgardi)`
- ?>

`$GLOBALS` massiviga ixtiyoriy funktsiya ko'rinish sohasida murojaat qilish mumkin va u dasturda foydalaniluvchi hamma global o'zgaruvchilarni o'z ichiga oladi.

## Ma'lumotlarni fitrlash

Validating data = Ma'lumotlarning to'g'ri shaklda ekanligini aniqlash

Sanitizing data = Ma'lumotlardan noqonuniy belgilarni olib tashlash

### PHP kengaytmasi

- PHP filtrlari tashqi kirishni tekshirish va sanitarizatsiya qilish uchun ishlatiladi.
- PHP filtri kengaytmasi foydalanuvchi kirishini tekshirish uchun zarur bo'lgan ko'plab funktsiyalarga ega va ma'lumotlarni tekshirishni osonroq va tezroq bajarishga mo'ljallangan.
- Ushbu `filter_list()` funktsiyadan PHP filtri kengaytmasi nimani taklif qilishi uchun foydalanish mumkin:

```
<table>
 <tr>
 <td>Filter Name</td>
 <td>Filter ID</td>
 </tr>
 <?php
 foreach (filter_list() as $id => $filter) {
 echo '<tr><td>' . $filter . '</td><td>' .
filter_id($filter) . '</td></tr>';
 }
 ?>
</table>
```

Filter Name	Filter ID
int	257
boolean	258
float	259
validate_regexp	272
validate_domain	277
validate_url	273
validate_email	274
validate_ip	275
validate_mac	276
string	513
stripped	513
encoded	514
special_chars	515
full_special_chars	522
unsafe_raw	516
email	517
url	518
number_int	519
number_float	520
magic_quotes	521
callback	1024

## **Nima uchun filtrlardan foydalanish kerak?**

- Ko'pgina veb-ilovalar tashqi ma'lumot oladi. Tashqi kirish / ma'lumotlar quyidagilar bo'lishi mumkin:
- Shakldan foydalanuvchi kiritishi
- Cookies
- Veb-xizmatlar ma'lumotlari
- Server o'zgaruvchilari
- Ma'lumotlar bazasi so'rovlari natijalari

## *Filtirlash funksiyalari*

- `filter_var()` Funksiya belgilangan filter bitta o'zgaruvchini filtrlaydi. Buning uchun ikkita ma'lumot kerak:
- Siz tekshirmoqchi bo'lgan o'zgaruvchi
- Foydalaniladigan o'zgaruvchi turi

```
<?php
```

```
$str = "<h1>Hello World!</h1>";
```

```
$newstr = filter_var($str,
FILTER_SANITIZE_STRING);
```

```
echo $newstr;
```

```
?>
```



```
<?php
 $int = 100;

 if (!filter_var($int, FILTER_VALIDATE_INT) === false) {
 echo("Integer is valid");
 } else {
 echo("Integer is not valid");
 }
?>
```

```
/////////
```

```
<?php
 $int = 0;

 if (filter_var($int, FILTER_VALIDATE_INT) === 0 ||
 !filter_var($int, FILTER_VALIDATE_INT) === false) {
 echo("Integer is valid");
 } else {
 echo("Integer is not valid");
 }
?>
```

## Manzilni filtirlash

```
<?php
 $ip = "127.0.0.1";

 if (!filter_var($ip, FILTER_VALIDATE_IP) ===
 false) {
 echo("$ip is a valid IP address");
 } else {
 echo("$ip is not a valid IP address");
 }
?>
```

## Emailni filtirlash

```
<?php
 $email = "john.doe@example.com";

 // Remove all illegal characters from email
 $email = filter_var($email,
 FILTER_SANITIZE_EMAIL);

 // Validate e-mail
 if (!filter_var($email, FILTER_VALIDATE_EMAIL)
 === false) {
 echo("$email is a valid email address");
 } else {
 echo("$email is not a valid email address");
 }
?>
```