

# Projektowanie Aplikacji Internetowych

Artur Niewiarowski

Wydział Fizyki, Matematyki i Informatyki

Politechnika Krakowska



# **PHP, CZ. 1**

## Kilka słów o PHP

- PHP – obiektowy, skryptowy język programowania, umożliwiający tworzenie programów po stronie serwera WWW.
- W przeciwieństwie do języka JS, PHP jest wykonywany po stronie serwera, nie po stronie klienta.

bo strona serwera, nie bo strona klienta.



## Kilka słów o PHP

Pomocne linki:

<http://php.net/manual/en/introduction.php>

<http://www.w3schools.com/php/default.asp>

<http://www.w3schools.com/php/default.asp>



## Kilka słów o PHP

- Skrypty PHP są umieszczane najczęściej w plikach tekstowych wraz z kodem HTML o rozszerzeniu *.php*.
- PHP pozwala na wykonywanie skryptów z linii poleceń.

```
[Artur@sirius test_php]$ php x.php -x 1 -y 2 -z 3
```

```
Wynik sumowania: 6
```

```
array(3) {  
  ["x"]=>  
  string(1) "1"  
  ["y"]=>  
  string(1) "2"  
  ["z"]=>  
  string(1) "3"  
}
```

```
[Artur@sirius test_php]$ more x.php
```

```
<?
```

```
$parametry = getopt("x:y:z:");
```

```
echo "Wynik sumowania: " . ($parametry["x"] + $parametry["y"] + $parametry["z"]);
```

```
echo "\n";
```

```
var_dump($parametry);
```

```
?>
```

## Kilka słów o PHP

- Z poziomu języka PHP możemy łączyć się z różnymi bazami danych, np. MySQL
- Implementacja PHP wraz z serwerem WWW Apache oraz serwerem baz danych MySQL określana jest jako platforma AMP – Apache, MySQL, PHP

```
<?php
```

```
$link = mysqli_connect("host","uzytkownik_bd_mysql","haslo","baza_danych") or  
die("Error " . mysqli_connect_error($link));
```

```
$wynik = mysqli_query($link, "select * from users order by login");
```

```
echo "Lista loginów: <br>";
```

```
while($wiersz = mysqli_fetch_array($wynik)) {  
    echo "{$wiersz['login']} <br>";  
}
```

```
?>
```

## Kilka słów o PHP

- Program piszemy pomiędzy znacznikami:

`<?php kod programu ?>` lub

`<? kod programu ?>`

Drugi sposób nie jest domyślny i należy go włączyć w pliku konfiguracyjnym *php.ini*

```
[root@sirius ~]# more /etc/php.ini | grep -i short_  
short_open_tag = On
```

## Prosty program

```
<!DOCTYPE html>
<html>
<body>

<?php
echo "Pewien tekst...";
?>

</body>
</html>
```



## Komunikaty o błędach w plikach php na torusie

Plik: **.htaccess**

```
php_flag display_errors on  
php_flag html_errors on
```

## Wyprowadzanie danych

```
<?php
$zmienna="tekst";
echo "<h2>Przykładowy \n $zmienna!</h2>";
echo "To ", "jest ", "przykładowy ", "$zmienna. \n";
print "Jeszcze jeden $zmienna\n";
echo print "Jeszcze jeden $zmienna\n";
?>
```

```
[root@sirius ~]# php w.php
<h2>Przykładowy
  tekst!</h2>To jest przykładowy tekst.
Jeszcze jeden tekst
Jeszcze jeden tekst
1
```

## Komentarze w PHP

```
<!DOCTYPE html>
<html>
<body>

<?php
// Komentarz

# Komentarz

/*
Komentarz
*/
?>

</body>
</html>
```

## Brak rozróżnialności wielkości liter dla poleceń wbudowanych w PHP

```
<!DOCTYPE html>
<html>
<body>

<?php
ECHO "Tekst HTML<br>";
echo "Tekst HTML<br>";
EcHo "Tekst HTML<br>";
?>

</body>
</html>
```

## Zmienne w PHP zaczynają się od znaku dolara \$

```
<!DOCTYPE html>
<html>
<body>

<?php
$kolor="red";
echo "Wybrany kolor: " . $kolor . "<br>";
echo "Kolor: " . $KOLOR . "<br>";
echo "Kolor: " . $koLOR . "<br>";
?>

</body>
</html>
```

- Wielkości znaków w nazwach zmiennych są istotne!
- Typów zmiennych nie deklarujemy. Rozpoznawane są na podstawie przypisanych im danych.

## Typy danych

- String
- Integer
- Floating point numbers
- Boolean
- Array
- Object
- NULL

## Typy danych: **String**

```
<?php
$x = "Hello world!";
echo $x;
echo "<br>";
$x = 'Hello world!';
echo $x;
?>
```

Hello world!  
Hello world!

## Typy danych: **String** – *funkcje operujące na ciągach*

```
<?
$s = "To JEST \"przykładowy\" <b>ciąg 'tekstowy'</b>";

echo "\n 1 " . $s ;
echo "\n 2 " . addslashes($s) ;
echo "\n 3 " . crc32($s) ;
echo "\n 4 " . explode(" ", $s) ;
echo "\n 5 " . htmlspecialchars($s) ;
echo "\n 6 " . htmlspecialchars_decode(htmlspecialchars($s)) ;
echo "\n 7 " . implode(" ", explode(" ", $s)) ;
echo "\n 8 " . md5($s) ;
echo "\n 9 " . sha1($s) ;

echo "\n";

?>
```



## Typy danych: **String** – *funkcje operujące na ciągach*

```
[root@sirius ~]# php fun_string.php

1 To JEST "przykładowy" <b>ciąg 'tekstowy'</b>
2 To JEST \"przykładowy\" <b>ciąg \'tekstowy\'</b>
3 4168044307
4 Array
5 To JEST &quot;przykładowy&quot; &lt;b&gt;ciąg
'tekstowy'&lt;/b&gt;
6 To JEST "przykładowy" <b>ciąg 'tekstowy'</b>
7 To JEST "przykładowy" <b>ciąg 'tekstowy'</b>
8 7361830fee48a377cebed797dce95f25
9 c1cb7ac00bd3318deac205875f292f2ad207d54a
```

## Typy danych: **Integer**

```
<?php
$x = 5985;
var_dump($x);
echo "<br>";
$x = -345;
var_dump($x);
echo "<br>";
$x = 0x8C; // wartość heksadecymalna
var_dump($x);
echo "<br>";
$x = 047; // wartość oktalna
var_dump($x);
?>
```

`int(5985)`

`int(-345)`

`int(140)`

`int(39)`

## Typy danych: **Floating Point**

```
<?php
$x = 10.365;
var_dump($x);
echo "<br>";
$x = 2.4e3;
var_dump($x);
echo "<br>";
$x = 8E-5;
var_dump($x);
?>
```

float(10.365)

float(2400)

float(8.0E-5)

## Typy danych: **Boolean**

```
<?  
  
$x=true;  
$y=false;  
  
var_dump($x);  
var_dump($y);  
  
?>
```

```
[root@sirius ~]# php z.php  
bool(true)  
bool(false)
```

## Typy danych: **Array**

```
<?php  
$cars=array("Volvo", "BMW", "Toyota");  
var_dump($cars);  
?>
```

```
array(3) { [0]=> string(5) "Volvo" [1]=> string(3) "BMW" [2]=> string(6) "Toyota" }
```

## Typy danych: **Array**

```
<?php

$tablica_imion[0]="Ala";
$tablica_imion[1]="Jan";
$tablica_imion[2]="Anna";
$tablica_imion[3]="Stefan";

echo "Wybrane imie: {$tablica_imion[0]}<br>";

$liczby = array(1,2,3,4,5);

echo "Wybrana liczba: {$liczby[1]}";
?>
```

## Typy danych: **Array**

```
<?php

$tablica_imion[0]="Ala";
$tablica_imion[1]="Jan";
$tablica_imion[2]="Anna";
$tablica_imion[3]="Stefan";
$tablica_imion["Kowalski"]="Jan";

echo "Wybrane imie: $tablica_imion[0]\n";
echo "Wybrane imie: { $tablica_imion['Kowalski'] }\n";
echo "Długość imienia: " . strlen($tablica_imion['Kowalski']) .
"\n";
?>
```

```
[root@sirius ~]# php u.php
Wybrane imie: Ala
Wybrane imie: Jan
Długość imienia: 3
```

## Typy danych: **Object**

```
<?
class Student {

var $nazwisko;
var $imie;
function __construct($nazwisko, $imie){

$this->nazwisko = $nazwisko;
$this->imie = $imie;

}
function pobierz_nazwisko(){return $this->nazwisko;}
function pobierz_imie(){return $this->imie;}
function pobierz_dane(){return $this->pobierz_nazwisko() . ", " . $this->pobierz_imie();}

}
?>

<?

$studenci = array (new Student('Kowalski', 'Jan'), new Student('Nowak', 'Anna'));

foreach ($studenci as $dane){
echo $dane->pobierz_dane() . "\n";

}

?>
```

```
[root@sirius ~]# php obj.php
```

```
Kowalski, Jan
```

```
Nowak, Anna
```



## Typy danych: **NULL**

```
<?php
$x="Hello world!";
$x=null;
var_dump($x);
?>
```

NULL

```
<?php
function suma($x,$y) {}

var_dump (suma(1,2));
?>
```

```
[root@sirius ~]# php t.php
NULL
```

## Zmienne globalne i lokalne

```
<?php
$x=5; // zasięg globalny

function Test()
{
$y=10; // zasięg lokalny
echo "<p>Wartości zmiennych wewnątrz funkcji :</p>";
echo "Variable x is: $x";
echo "<br>";
echo "Variable y is: $y";
}

Test();

echo "<p>Wartości zmiennych poza funkcją:</p>";
echo "Variable x is: $x";
echo "<br>";
echo "Variable y is: $y";
?>
```

## Zmienne globalne i lokalne

```
<?php
$x=5; $y=10;

function Test() { global $x,$y; $y=$x+$y; }
Test();
echo $y;
?>
```

```
<?php
$x=5; $y=10;

function Test2() { $GLOBALS['y']=$GLOBALS['x']+$GLOBALS['y']; }

Test2();
echo $y;
?>
```

## Zmienna **static**

```
<?php  
  
function Test()  
{  
  static $x=0;  
  echo $x;  
  $x++;  
}  
  
Test();  
Test();  
Test();  
  
?>
```

```
js>
```

```
Test():
```

```
Test():
```

```
Test():
```

## Peęta **for**

```
<?php
for ($x=0; $x<=10; $x++)
{
    echo "Numer: $x <br>";
}
?>
```

## Pętla **foreach**

```
<?php
$kolory = array("czerwony", "zielony", "niebieski", "żółty");

foreach ($kolory as $wartosc)
{
    echo "$wartosc <br>";
}
?>
```

```
<?php
$kolory = array("czerwony" =>
"red", "zielony"=>"green", "niebieski"=>"blue", "żółty"=>"yellow");

foreach ($kolory as $pol=>$ang)
{
    echo "$pol == $ang \n";
}
?>
```

## Warunki

```
<?php
$t=date("H");
if ($t<"10")
{
    echo "Have a good morning!";
}
elseif ($t<"20")
{
    echo "Have a good day!";
}
else
{
    echo "Have a good night!";
}
?>
```

## Operatory porównania

Operator	Przykład	Wynik
==	<code>\$x == \$y</code>	true jeżeli <code>\$x</code> jest równe <code>\$y</code>
===	<code>\$x === \$y</code>	true jeżeli <code>\$x</code> jest równe <code>\$y</code> i są tego samego typu
!=	<code>\$x != \$y</code>	true jeżeli <code>\$x</code> nie jest równe <code>\$y</code>
<>	<code>\$x &lt;&gt; \$y</code>	true jeżeli <code>\$x</code> nie jest równe <code>\$y</code>
!==	<code>\$x !== \$y</code>	true jeżeli <code>\$x</code> nie jest równe <code>\$y</code> lub nie są tego samego typu
>	<code>\$x &gt; \$y</code>	
<	<code>\$x &lt; \$y</code>	
>=	<code>\$x &gt;= \$y</code>	
<=	<code>\$x &lt;= \$y</code>	



## Operatory arytmetyczne

```
<?php
$x=10;
$y=5;
echo "\n1. " . ($x + $y);
echo "\n2. " . ($x - $y);
echo "\n3. " . ($x * $y);
echo "\n4. " . ($x / $y);
echo "\n5. " . ($x % $y);
echo "\n6. " . ($x % 2);
echo "\n7. " . ($y % 2);
echo "\n"
?>
```

```
1. 15
2. 5
3. 50
4. 2
5. 0
6. 0
7. 1
```

## Operatory inkrementacji i dekrementacji

```
<?php

$x=10;
echo ++$x; // wynik: 11
echo "\n";

$y=10;
echo $y++; // wynik: 10
echo "\n";

$z=5;
echo --$z; // wynik: 4
echo "\n";

$i=5;
echo $i--; // wynik: 5
echo "\n";

?>
```

## Operatory logiczne

### Operator

and

or

xor

&&

||

!

### przykład

`$x and $y`

`$x or $y`

`$x xor $y`

`$x && $y`

`$x || $y`

`!$x`

## Wybory

```
<?
$liczba = rand(0,6);
echo "Wylosowałem ";
switch ($liczba) {

case 0: echo "zero!"; break;
case 1: echo "jedynekę!"; break;
case 2: echo "dwójkę!"; break;
case 3: echo "trójkę!"; break;
default: echo "liczbę z przedziału 4 do 6 :)";
}
echo "\n";

?>
```

```
[Artur@sirius test_php]$ php case.php
Wylosowałem dwójkę!
[Artur@sirius test_php]$ php case.php
Wylosowałem liczbę z przedziału 4 do 6 :)
[Artur@sirius test_php]$ php case.php
Wylosowałem jedynkę!
[Artur@sirius test_php]$
```

## Pętla **while**

```
<?php
$x=1;

while ($x<=5)
{
    echo "Numer: $x <br>";
    $x++;
}
?>
```

```
$q = mysqli_query($link, "select * from users");

echo "<table>";
echo "<tr><th>imię</th><th>nazwisko</th></tr>";
while ($w = mysqli_fetch_array($q)) { ... }
```

## Funkcje

```
<?php
function szczyt ($nazwa, $wysokosc)
{
echo "<tr><td>$nazwa</td><td>$wysokosc</td></tr>";
}

echo "<table>";
szczyt ("Rysy", "2503");
szczyt ("Beskid", "2012");

echo "</table>";

?>
```

## Funkcje – zwracanie wartości, parametr opcjonalny

```
<?php
function suma($x,$y=0)
{
return $x + $y;
}

echo "wynik: " . suma(1,2);

?>
```

```
3>
```

```
wynik: 3
```

## Zmienne globalne

```
$GLOBALS  
$_SERVER  
$_POST  
$_GET  
$_SESSION  
$_COOKIE
```

```
$_COOKIE
```



## \$GLOBALS

```
<?php
$x = 10;
$y = 20;

function suma()
{
$GLOBALS['z'] = $GLOBALS['x'] + $GLOBALS['y'];
}

suma();
echo $z . " ({"$GLOBALS['z']})";
?>
```

## **\$\_SERVER**

```
<?php
echo "plik skryptu: " . $_SERVER['PHP_SELF'];
echo "<br>";
echo "nazwa serwera: " . $_SERVER['SERVER_NAME'];
echo "<br>";
echo "nazwa hosta: " . $_SERVER['HTTP_HOST'];
echo "<br>";
echo "przeglądarka klienta: " . $_SERVER['HTTP_USER_AGENT'];
echo "<br>";
echo "nazwa oprogramowania serwera: " . $_SERVER['SERVER_SOFTWARE'];
echo "<br>";
echo "adres IP klienta: " . $_SERVER['REMOTE_ADDR'];
echo "<br>";
echo "nazwa hosta klienta: " . $_SERVER['REMOTE_HOST'];
echo "<br>";
echo "port klienta: " . $_SERVER['REMOTE_PORT'];?>
```

## Wysyłanie danych na serwer: `$_POST`

```
<?php

if (isset($_POST['x'])) { $wynik = $_POST['x'] + $_POST['y']; } else
{$wynik="";}

?>

<form method="post" action="post.php">
<pre>
Dodawanie:
X: <input type="text" name="x">
Y: <input type="text" name="y">

Wynik: <input type="text" value="<? echo $wynik; ?>">
<input type="submit" value="Dodaj">

</pre>
</form>
```

## Wysyłanie danych na serwer: \$\_POST

```
<?php

if ($_POST['operacja'] == "plus")
if (isset($_POST['x'])) { $wynik = $_POST['x'] + $_POST['y']; } else
{$wynik="";}
elseif ($_POST['operacja'] == "minus")
if (isset($_POST['x'])) { $wynik = $_POST['x'] - $_POST['y']; } else
{$wynik="";}

?>

<form method="post" action="post.php" autocomplete=off>
<pre>
Działania matematyczne:
X: <input type=text name="x">
Y: <input type=text name="y">

<input type=radio value="plus" name=operacja checked> + <input type=radio
value="minus" name=operacja> -

Wynik: <input type=text value="<? echo $wynik; ?>">
<input type=submit value="Wykonaj">

</pre>
</form>
```

## Wysyłanie danych na serwer: **\$\_POST** – *prześlij opinię, cz. 1*

```
<?
if (isset($_POST['login'])) {
echo "Twoja opinia: <br>";
echo "Treść: " . $_POST['opinia'] ;
}
?>

<form method=post id=forma>
<pre>
Login: <input type=text name=login>
Opinia:
<textarea name=opinia></textarea>
<input type=button value="Wyślij"
onClick="document.getElementById('forma').submit();" >
</pre>
</form>
```

## Wysyłanie danych na serwer: \$\_POST – prześlij opinię, cz. 2

Twoja opinia:

Treść: **przeczytaj to!!**

Login:

Opinia:

```
<a style=font-size:50px>przeczytaj  
to!!</a>
```

Login:

Opinia:

```
<script>  
document.location.href="http://hacker.ru";  
</script>
```

Login:

Opinia:

```
<noscript>
```



Twoja opinia:  
Treść:

## Wysyłanie danych na serwer: **\$\_POST** – *prześlij opinię, cz. 3*

```
<?
if (isset($_POST['login'])) {
echo "Twoja opinia: <br>";
echo "Treść: " . htmlspecialchars($_POST['opinia']) ;
}
?>
```

Twoja opinia:

Treść: <b>AAAAAAAAA!!!!</b> <noscript>

Login:

Opinia:

<b>AAAAAAAAA!!!!</b>

<noscript>|

Wyślij

## Wysyłanie danych na serwer: \$\_GET

```
<?
foreach ($_GET as $x => $y) {
echo "$x => $y <br>";
}
?>

<form method=get>
<pre>
Login: <input type=text name=login>
Hasło: <input type=password name=haslo>
<input type=submit value="OK">
</pre>
</form>
```



← 🔑 [redacted] get.php?login=<b>login</b>&haslo=haslo

login => login  
haslo => haslo

Login: <b>login</b>  
Hasło: ●●●●●●

OK



## Wysyłanie danych na serwer: \$\_GET

```
<?
if (isset($_GET['f'])) {
include $_GET['f'];
//include_once $_GET['f'];
//require $_GET['f'];
}
?>

<a href=" ?f=plik1.php">plik 1</a>
<a href=" ?f=pl12&p=4">plik 2</a>
```

## Wysyłanie danych na serwer: \$\_GET

```
<?
if (isset($_GET['f'])) {
    switch ($_GET['f']) {
        default:
            if (file_exists($_GET['f'])) {
                echo "plik istnieje, ale nie mogę go odczytać!";
            } else {echo "plik nie istnieje!";} ; break;
        case "plik1.php": include_once("plik1.php"); break;
        case "plik2.php": include_once("plik2.php"); break;
    } //switch
}
?>

<a href="?f=plik1.php">plik 1</a>
<a href="?f=plik2">plik 2</a>
```

## **\$\_SESSION**

```
<?php  
  
session_start();  
$_SESSION['login'] = $_POST['login'];  
$_SESSION['id_los'] = md5(rand(-100,100));  
  
?>
```

## **\$\_SESSION**

```
<?

if (isset($_GET['logout'])) {
session_unset();
session_destroy();
setcookie(session_name(), '', -1, '/');
} else {
session_start();
}

?>

<html>
<body>
```

## **\$\_SESSION** – konfiguracja w *php.ini*

```
[Artur@sirius test_php]$ more /etc/php.ini | grep -i session.use_trans_sid  
session.use_trans_sid = 0
```

## **\$\_SESSION** – konfiguracja w *php.ini*

```
<?
session_start();

if ($_GET['x'] == 1) {

$_SESSION['tajne_id'] = '1234567890';

}

echo "{$_SESSION['tajne_id']}";

?>
```

## **\$\_SESSION** – konfiguracja w *php.ini*

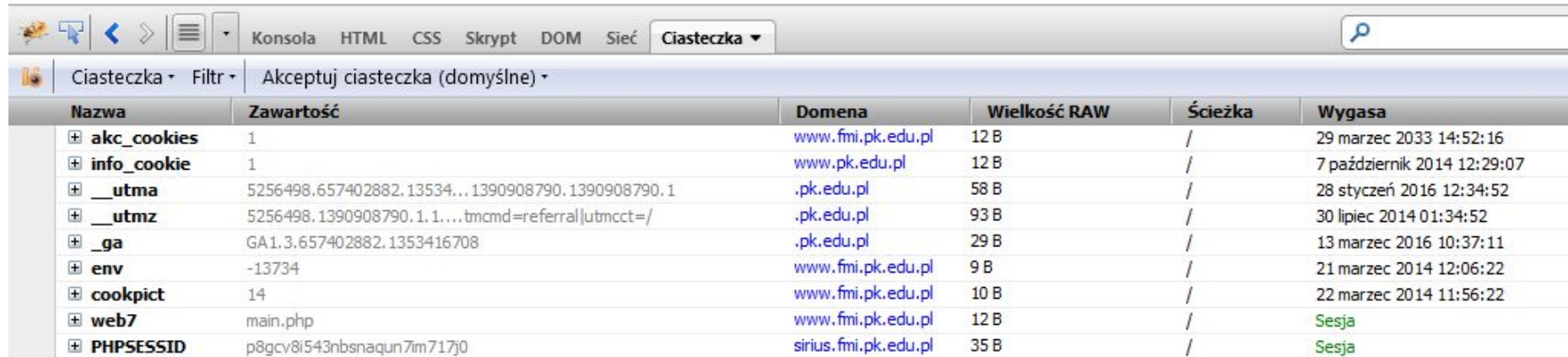
▲ Nagłówki żądania (0,368 KB)
<b>Host:</b> "sirius.fmi.pk.edu.pl"
<b>User-Agent:</b> "Mozilla/5.0 (Windows NT 6.3; WOW64; rv:32.0) Gecko/20100101 Firefox/32.0"
<b>Accept:</b> "text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8"
<b>Accept-Language:</b> "pl,en-US;q=0.7,en;q=0.3"
<b>Accept-Encoding:</b> "gzip, deflate"
<b>Cookie:</b> "PHPSESSID=II9oev52r0n4qcnmctajspmd72"
<b>Connection:</b> "keep-alive"

---

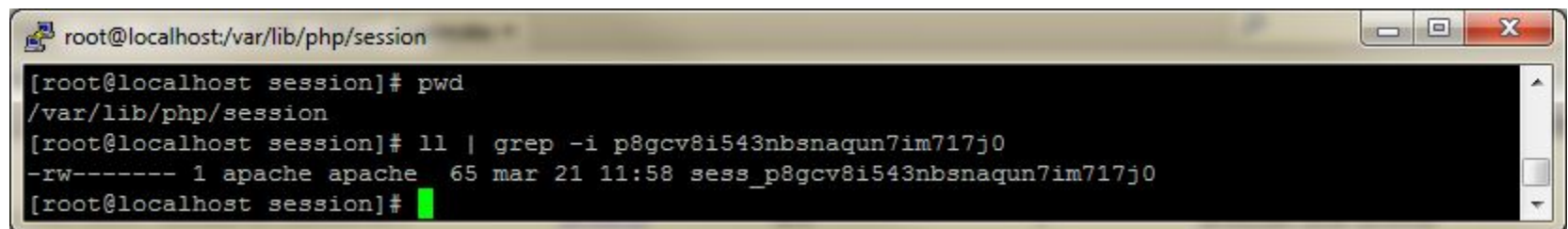
[sesja.php?PHPSESSID=II9oev52r0n4qcnmctajspmd72](#)

---

## \$\_SESSION



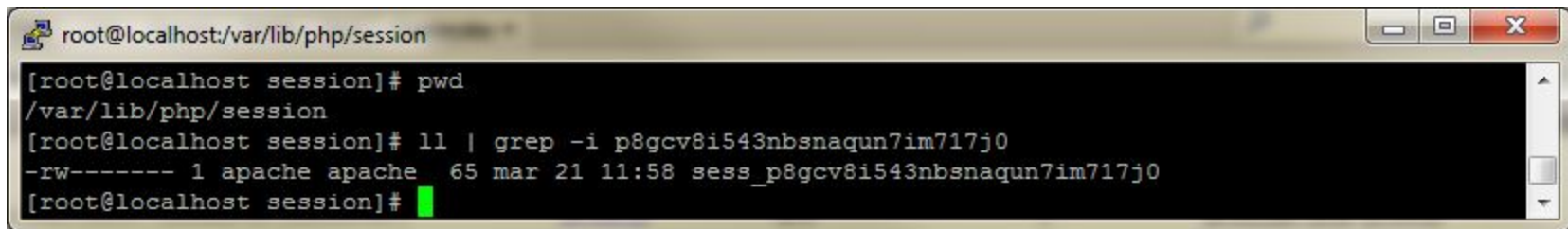
Nazwa	Zawartość	Domena	Wielkość RAW	Ścieżka	Wygasa
akc_cookies	1	www.fmi.pk.edu.pl	12 B	/	29 marzec 2033 14:52:16
info_cookie	1	www.pk.edu.pl	12 B	/	7 październik 2014 12:29:07
__utma	5256498.657402882.13534...1390908790.1390908790.1	.pk.edu.pl	58 B	/	28 styczeń 2016 12:34:52
__utmz	5256498.1390908790.1.1....tmcmd=referral utmctt=	.pk.edu.pl	93 B	/	30 lipiec 2014 01:34:52
_ga	GA1.3.657402882.1353416708	.pk.edu.pl	29 B	/	13 marzec 2016 10:37:11
env	-13734	www.fmi.pk.edu.pl	9 B	/	21 marzec 2014 12:06:22
cookpict	14	www.fmi.pk.edu.pl	10 B	/	22 marzec 2014 11:56:22
web7	main.php	www.fmi.pk.edu.pl	12 B	/	Sesja
PHPSESSID	p8gcv8i543nbsnaqun7im717j0	sirius.fmi.pk.edu.pl	35 B	/	Sesja



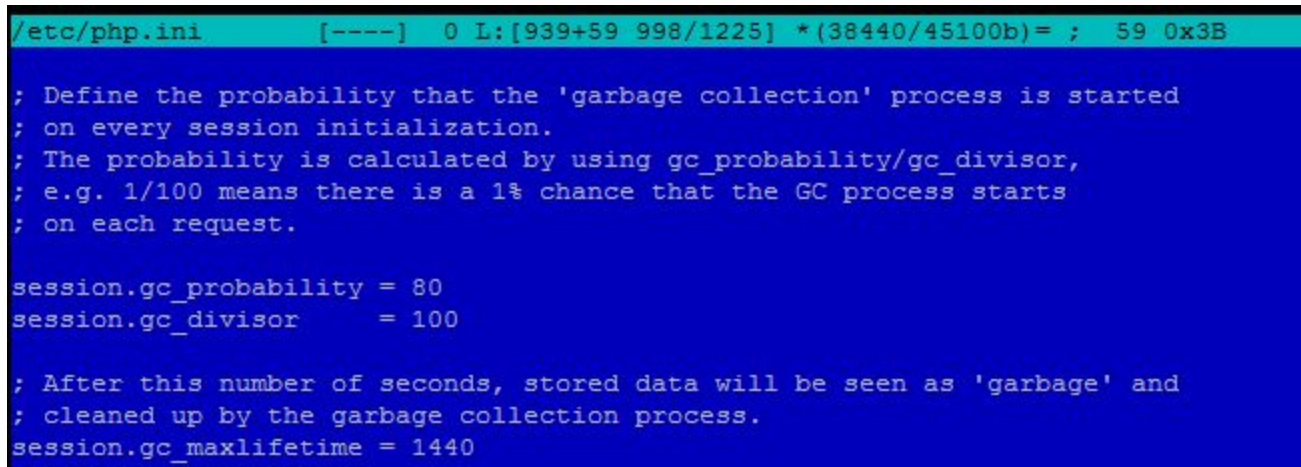
```
root@localhost:var/lib/php/session
[root@localhost session]# pwd
/var/lib/php/session
[root@localhost session]# ll | grep -i p8gcv8i543nbsnaqun7im717j0
-rw----- 1 apache apache 65 mar 21 11:58 sess_p8gcv8i543nbsnaqun7im717j0
[root@localhost session]#
```



## \$\_SESSION – garbage collection



```
root@localhost:var/lib/php/session
[root@localhost session]# pwd
/var/lib/php/session
[root@localhost session]# ll | grep -i p8gcv8i543nbsnaqun7im717j0
-rw----- 1 apache apache 65 mar 21 11:58 sess_p8gcv8i543nbsnaqun7im717j0
[root@localhost session]#
```



```
/etc/php.ini [----] 0 L:[939+59 998/1225] *(38440/45100b)= ; 59 0x3B

; Define the probability that the 'garbage collection' process is started
; on every session initialization.
; The probability is calculated by using gc_probability/gc_divisor,
; e.g. 1/100 means there is a 1% chance that the GC process starts
; on each request.

session.gc_probability = 80
session.gc_divisor     = 100

; After this number of seconds, stored data will be seen as 'garbage' and
; cleaned up by the garbage collection process.
session.gc_maxlifetime = 1440
```

## \$\_COOKIE

```
bool setcookie ( string $name [, string $value [, int $expire = 0 [, string $path [, string  
$domain [, bool $secure = false [, bool $httponly = false ]]]]] )
```

**name** – nazwa ciasteczka

**value** – wartość, którą przechowuje ciasteczko

**expire** – czas ważności wyrażony w sekundach: **time()+sek**

**path** – ścieżka na serwerze, np. /~login/

**domain** – domena,

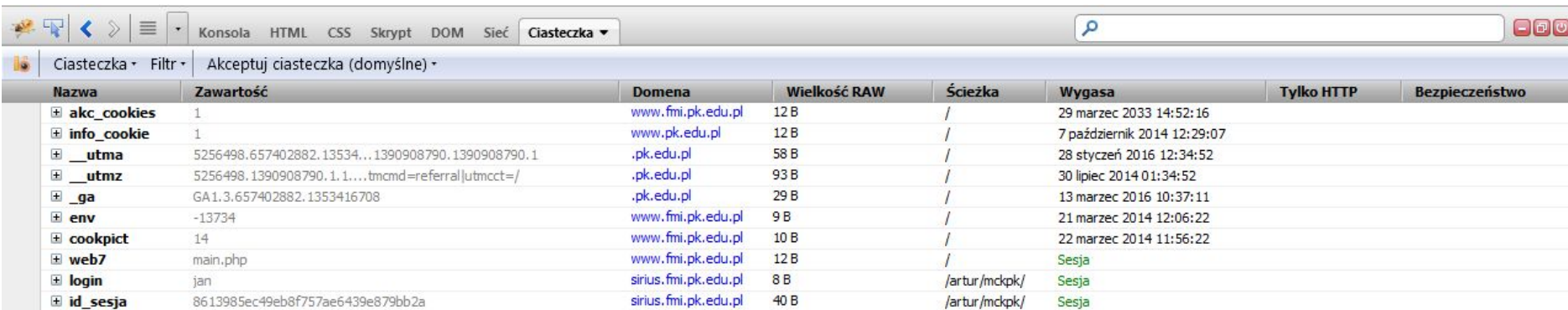
**secure** – połączenie szyfrowane,

**httponly** – jeżeli true – cookie nie będzie dostępny dla skryptów

## **\$\_COOKIE**

```
<?php  
  
setcookie("login", $_POST['login']); //, time()+3600);  
setcookie("id_sesja", md5(rand(-100,100))) ;//, time()+3600);  
  
?>
```

## \$\_COOKIE – plik loginc.php



The screenshot shows a web browser's cookie manager interface. The title bar indicates the current view is 'Ciasteczka' (Cookies). The main area displays a table of cookies with columns for Name, Content, Domain, Size, Path, Expiry, and Security. The cookies listed include 'akc\_cookies', 'info\_cookie', '\_\_utma', '\_\_utmz', '\_ga', 'env', 'cookpict', 'web7', 'login', and 'id\_sesja'.

Nazwa	Zawartość	Domena	Wielkość RAW	Ścieżka	Wygasa	Tylko HTTP	Bezpieczeństwo
akc_cookies	1	www.fmi.pk.edu.pl	12 B	/	29 marzec 2033 14:52:16		
info_cookie	1	www.pk.edu.pl	12 B	/	7 październik 2014 12:29:07		
__utma	5256498.657402882.13534...1390908790.1390908790.1	.pk.edu.pl	58 B	/	28 styczeń 2016 12:34:52		
__utmz	5256498.1390908790.1.1...tcmd=referral utmct=/	.pk.edu.pl	93 B	/	30 lipiec 2014 01:34:52		
_ga	GA1.3.657402882.1353416708	.pk.edu.pl	29 B	/	13 marzec 2016 10:37:11		
env	-13734	www.fmi.pk.edu.pl	9 B	/	21 marzec 2014 12:06:22		
cookpict	14	www.fmi.pk.edu.pl	10 B	/	22 marzec 2014 11:56:22		
web7	main.php	www.fmi.pk.edu.pl	12 B	/	Sesja		
login	jan	sirius.fmi.pk.edu.pl	8 B	/artur/mckpk/	Sesja		
id_sesja	8613985ec49eb8f757ae6439e879bb2a	sirius.fmi.pk.edu.pl	40 B	/artur/mckpk/	Sesja		

## **\$\_COOKIE**

```
<?

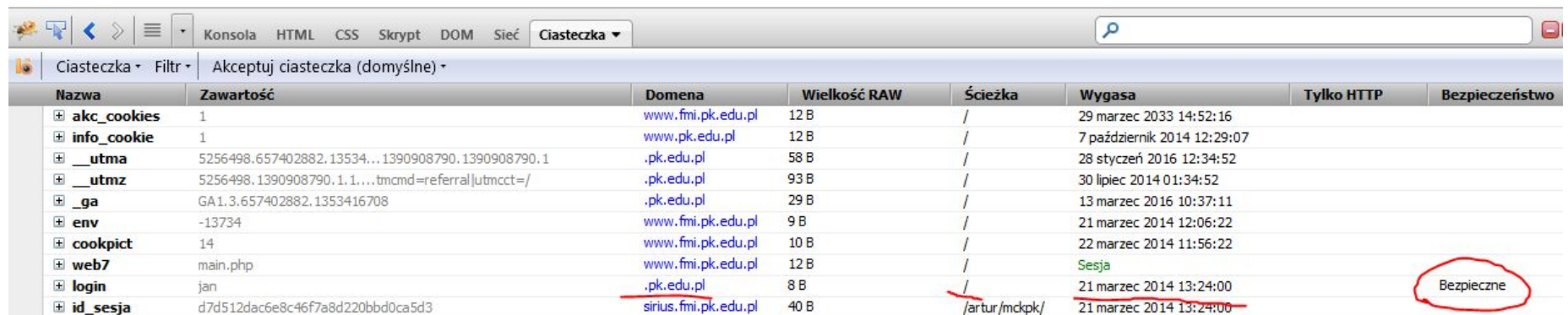
if (isset($_GET['logout'])) {
setcookie("login","", time()-1); unset($_COOKIE['login']);
setcookie("id_sesja","", time()-1);
unset($_COOKIE['id_sesja']);
}

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

## \$\_COOKIE

```
<?php  
  
setcookie("login", $_POST['login'], time()+3600, "/",  
".pk.edu.pl",1);  
setcookie("id_sesja", md5(rand(-100,100)), time()+3600);  
  
?>  
<html>  
<body>
```

## \$\_COOKIE – plik loginc.php



Nazwa	Zawartość	Domena	Wielkość RAW	Ścieżka	Wygasa	Tylko HTTP	Bezpieczeństwo
akc_cookies	1	www.fmi.pk.edu.pl	12 B	/	29 marzec 2033 14:52:16		
info_cookie	1	www.pk.edu.pl	12 B	/	7 październik 2014 12:29:07		
__utma	5256498.657402882.13534...1390908790.1390908790.1	.pk.edu.pl	58 B	/	28 styczeń 2016 12:34:52		
__utmz	5256498.1390908790.1.1....tmcmd=referral utmctt=	.pk.edu.pl	93 B	/	30 lipiec 2014 01:34:52		
_ga	GA1.3.657402882.1353416708	.pk.edu.pl	29 B	/	13 marzec 2016 10:37:11		
env	-13734	www.fmi.pk.edu.pl	9 B	/	21 marzec 2014 12:06:22		
cookpict	14	www.fmi.pk.edu.pl	10 B	/	22 marzec 2014 11:56:22		
web7	main.php	www.fmi.pk.edu.pl	12 B	/	Sesja		
login	jan	.pk.edu.pl	8 B	/	21 marzec 2014 13:24:00		
id_sesja	d7d512dac6e8c46f7a8d220bbd0ca5d3	sirius.fmi.pk.edu.pl	40 B	/artur/mckpk/	21 marzec 2014 13:24:00		Bezpieczne

## \$\_COOKIE

```
<?  
  
setcookie("login","",0, "/", ".pk.edu.pl",1);  
unset($_COOKIE['login']);  
setcookie("id_sesja","",time()-1);  
unset($_COOKIE['id_sesja']);  
}  
  
?>  
<html>  
<body>  
<?
```





## **Metodologia zalogowania, utrzymania sesji i wylogowania**

# Proces zalogowania

## Przykładowe tabele

```
CREATE TABLE `users` (  
  `ID_users` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `login` varchar(50) NOT NULL,  
  `pass` varchar(32) NOT NULL,  
  PRIMARY KEY (`ID_users`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `sesja` (  
  `ID_sesja` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `ID_users` int(10) unsigned NOT NULL,  
  `id` varchar(64) NOT NULL,  
  `ip` varchar(39) DEFAULT NULL,  
  `web` varchar(200) DEFAULT NULL,  
  `time` timestamp NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE  
CURRENT_TIMESTAMP,  
  PRIMARY KEY (`ID_sesja`),  
  KEY `fkIDu` (`ID_users`),  
  CONSTRAINT `fkIDu` FOREIGN KEY (`ID_users`) REFERENCES `users`  
(`ID_users`) ON DELETE CASCADE ON UPDATE CASCADE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

# Proces zalogowania

## Przykładowy formularz

```
<form method=post>  
login: <input type=text name=login><br>  
hasło: <input type=password name=pass><br>  
<input type=submit value="Zaloguj">  
</form>
```

login:

hasło:

# Proces zalogowania

## Przykładowy proces sprawdzenia wysłanych danych logowania

```
<?
$link = mysqli_connect("localhost", "test", "haslo","test");

foreach ($_POST as $k=>$v) {$POST[$k] = mysqli_real_escape_string($link, $v);}
foreach ($_SERVER as $k=>$v) {$SERVER[$k] = mysqli_real_escape_string($link, $v);}

if (isset($_POST['login'])) {

$q = mysqli_fetch_assoc( mysqli_query($link, "select count(*) cnt, ID_users from users where
login='{$_POST['login']}' and pass = md5('{$_POST['pass']}')"););

if ($q['cnt']) {
    $id = md5(rand(-10000,10000) . microtime()) . md5(crc32(microtime()) .
$_SERVER['REMOTE_ADDR']);
mysqli_query($link, "delete from sesja where ID_users = '$q[ID_users]'");
mysqli_query($link, "
insert into sesja (ID_users, id, ip, web) values
('$q[ID_users]','$id','$_SERVER[REMOTE_ADDR]','$_SERVER[HTTP_USER_AGENT]')");
if (! mysqli_errno($link)) {
    setcookie("id", $id);
    echo "zalogowano pomyślnie!";
    header("location:zalogowano.php");
} else {echo "błąd podczas logowania!";}

} else {
    echo "błąd logowania!";
} }
?>
```

# Proces zalogowania

## Po zalogowaniu

▼ Ciasteczka odpowiedzi

**id:** "4b06b24e1f26fc299f43e8bd0a826d505257c4a598593a608f4811adb98a388a"

▼ Ciasteczka odpowiedzi

ID_sesja	ID_users	id	ip	web	time
I	2	1 4b06b24e1f26fc299f43e8bd0a826d50!	149.156.136.3	Mozilla/5.0 (Windows NT 6.	2015-11-27 11:02:42

## Proces zalogowania

### Po zalogowaniu – sprawdzamy zalogowanie

```
<?
$link = mysqli_connect("localhost", "test", "haslo","test");

foreach ($_COOKIE as $k=>$v) {$COOKIE[$k] = mysqli_real_escape_string($link,
$v);}
foreach ($_SERVER as $k=>$v) {$SERVER[$k] = mysqli_real_escape_string($link,
$v);}

if (! isset($_COOKIE['id'])) {header("location:logowanie.php");exit;}

$q = mysqli_fetch_assoc(mysqli_query($link, "select ID_users from sesja where
id = '$COOKIE[id]' and web = '$SERVER[HTTP_USER_AGENT]' AND ip =
'$SERVER[REMOTE_ADDR]';"));

if (! empty($q['ID_users'])) {
    echo "Zalogowany użytkownik o ID: " . $q['ID_users'] ;
} else {
    header("location:logowanie.php");exit;
}
?> Witamy w panelu użytkownika!
```

# Proces zalogowania

## Po zalogowaniu – wylogowanie

```
if (isset($_GET['logout'])) {  
    $q = mysqli_query($link, "delete from sesja where  
    id = '$_COOKIE[id]' and web = '$_SERVER[HTTP_USER_AGENT]';");  
    setcookie("id", 0, time() - 1);  
    unset($_COOKIE['id']);  
}
```

Zalogowany użytkownik o ID: 2 ([wyloguj](#))

```
<a href="?logout">(wyloguj)</a>
```

Filtruj ciasteczka

▼ Ciasteczka ządania

**\_utma:** "5256498.1017658489.1396515726.1400143825.1400143825.1"

**\_ga:** "GA1.3.1017658489.1396515726"

ID_sesja	ID_users	id	ip	web	time
▶ (Null)	(Null)	(Null)	(Null)	(Null)	(Null)

# Proces zalogowania

```
public String getHTMLcode(string Url, string proxy = "", string userAgent = "Mozilla/5.0 (Windows NT 6.3; WOW64; rv:34.0) Gecko/20100101 Firefox/34.0")
{
    HttpWebRequest myRequest;
    try
    {
        if (Url.IndexOf("http") != 0) { Url = "http://apps.webofknowledge.com" + Url; }
        myRequest = (HttpWebRequest)WebRequest.Create(Url);
        // Console.WriteLine ("URL: " + Url);
    }
    negocjacja sesji w ramach błędu
    myRequest.Method = "GET";
    myRequest.UserAgent = userAgent ;
    myRequest.Accept = "text/html,application/xhtml+xml,application/xml;q=0.9;*/:*;q=0.8";
}
```

jak również:

[https://pl.wikipedia.org/wiki/IP\\_spoofing](https://pl.wikipedia.org/wiki/IP_spoofing)



# Logowanie do systemu - zabezpieczenie logowania

## Przykłady braku zabezpieczenia protokołem *https*

Często odwiedzane  Pierwsze kroki  Galeria obiektów Web ... Sugerowane witryny

### System ankietyowy "Ocena nauczycieli akademickich"

Login:  
artur

Hasło:  
●●●●●●

Zaloguj się

Do logowania należy zastosować login i hasło używane przy logowaniu do serwisu informacyjnego pracowników PK.

# Logowanie do systemu - zabezpieczenie logowania

## Przykłady braku zabezpieczenia protokołem *https*

```
root@localhost:~  
[root@localhost ~]# tcpflow -i eno2 src host 10.0.0.3 and dst port 80 -c  
tcpflow[3347]: listening on eno2  
010.000.000.003.64491-149.156.132.053.00080: POST /pracownik/logowanie.php HTTP/  
1.1  
Host: ankiety.pk.edu.pl  
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:41.0) Gecko/20100101 Firefox/  
41.0  
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8  
Accept-Language: pl,en-US;q=0.7,en;q=0.3  
Accept-Encoding: gzip, deflate  
Content-Type: application/x-www-form-urlencoded; charset=UTF-8  
Referer: http://ankiety.pk.edu.pl/pracownik/login.php  
Content-Length: 25  
Cookie: _ga=GA1.3.1245974602.1422113417; PHPSESSID=5n2gdcmbp83jd77humhrb9crj5  
Connection: keep-alive  
Pragma: no-cache  
Cache-Control: no-cache  
  
login=artur&haslo=haselko
```

obiektów Web ... Sugerowane witryny

### ankietowy cieli akademickich"

Login:

Hasło:

Zaloguj się



Do logowania należy zastosować login i hasło używane przy logowaniu do serwisu informacyjnego pracowników PK

**Upload plików na serwer do katalogu i do bazy danych**

## Wysyłanie maili

```
function send_email($email, $title, $src, $reply_email,$from_user ){
    $from_user= mail_escape_header($from_user);

    $headers="From:$from_user <mail@adres.edu.pl>\r\nX-Mailer:mailer";
    $headers=$headers."r\nReply-To:$reply_email\r\nContent-Type: text/html;
    charset=UTF-8\r\nContent-Transfer-Encoding: 8bit\n";

    mail($email, mail_escape_header($title), $src,$headers );
}
```

## Wysyłanie maili

```
function mail_escape_header($subject){
    $subject = preg_replace('/([^\a-z ])/ie',
'sprintf("=%02x",ord(StripSlashes("\1")))', $subject);
    $subject = str_replace(' ', '_', $subject);
    return "=?utf-8?Q?$subject?=";
}
```

# **Połączenie z bazą danych MySQL/MariaDB**

## Połączenie z bazą danych MySQL/MariaDB

```
$polaczenie = mysqli_connect("localhost", "uzytkownik", "haslo", "baza") or  
die ("błąd połączenia: " . mysqli_connect_error ());  
  
if ($polaczenie) {} else {echo "błąd połączenia "};  
  
if (mysqli_connect_errno()) {echo "błąd połączenia: " . mysqli_connect_error  
() . " (" . mysqli_connect_errno() . ")" ; }  
  
$pyt = mysqli_query($polaczenie, "select now() czas");  
  
while ($tabl = mysqli_fetch_array($pyt)) {  
echo $tabl['czas'];  
}
```

## Zapytania do bazy danych – przykład pobierania danych

```
$pyt = mysqli_query($polaczenie, "select * from users");  
  
echo "  
<style>  
table, tr, th, td {border: 1px solid black;border-collapse:collapse}  
</style>  
";  
  
echo "<table>";  
echo "<tr><th>imię</th><th>nazwisko</th></tr>";  
while ($stabl = mysqli_fetch_array($pyt)) {  
  
echo "<tr><td>{$stabl['imię']}</td><td>{$stabl['nazwisko']}</td></tr>";  
}  
  
echo "</table>";
```

ID_users	login	pass	imię	nazwisko	data_urodzenia
2	anna	81dc9bdb52d04dc20036d	Anna	Nowak	1970-01-01
4	jan	202cb962ac59075b964b0	Jan	Kowalski	2015-02-02

imię	nazwisko
Jan	Kowalski
Anna	Nowak



## Zapytania do bazy danych – przykład umieszczania danych

```
<form method=post action="dodaj_studenta.php">
<table>
<tr><td>Login: </td><td><input type=text name=login required></td></tr>
<tr><td>Hasło: </td><td><input type=password name=haslo required></td></tr>
<tr><td>Imię: </td><td><input type=text name=imie required></td></tr>
<tr><td>Nazwisko: </td><td><input type=text name=nazwisko required></td></tr>

<tr><td colspan=2><input type=submit value="Dodaj studenta"></td></tr>
</table>
</form>
```

## Zapytania do bazy danych – przykład umieszczania danych

```
<?
if (isset($_POST['login'])) {
$_POST['imie'] = mysqli_real_escape_string($polaczenie, $_POST['imie']);
$_POST['nazwisko'] = mysqli_real_escape_string($polaczenie ,
$_POST['nazwisko']);
$_POST['login'] = mysqli_real_escape_string($polaczenie , $_POST['login']);
$_POST['haslo'] = mysqli_real_escape_string($polaczenie , $_POST['haslo']);

    $q=mysqli_query($polaczenie, "
insert into studenci (login, haslo, imie, nazwisko)
values ('{$_POST['login']}', '{$_POST['haslo']}' ,
'{$_POST['imie']}' , '{$_POST['nazwisko']}'");
");

if (strlen (mysqli_error($polaczenie)) > 0) {
    echo "<br><br>Nie dodano! <br><br>";
    //" . mysqli_error($polaczenie)."<br>
} else
{
    echo "<br><br>Dodano pomyślnie<br><br>";
}

}
?>
```

## Upload plików na serwer do katalogu i do bazy danych

```
<html>
<body>

<form action="upload_file.php" method="post"
enctype="multipart/form-data">
<label for="file">Filename:</label>
<input type="file" name="file" id="file"><br>
<input type="submit" name="submit" value="Submit">
</form>

</body>
</html>
```

## Upload plików na serwer do katalogu i do bazy danych

```
<?php
if ($_FILES["file"]["error"] > 0) {
    echo "Error: " . $_FILES["file"]["error"] . "<br>";
} else {
    echo "Upload: " . $_FILES["file"]["name"] . "<br>";
    echo "Type: " . $_FILES["file"]["type"] . "<br>";
    echo "Size: " . ($_FILES["file"]["size"] / 1024) . " kB<br>";
    echo "Stored in: " . $_FILES["file"]["tmp_name"];
}
?>
```

## Upload plików na serwer do katalogu i do bazy danych

```
<?php
$allowedExts = array("gif", "jpeg", "jpg", "png");
$temp = explode(".", $_FILES["file"]["name"]);
$extension = end($temp);

if ((($_FILES["file"]["type"] == "image/gif")
|| ($_FILES["file"]["type"] == "image/jpeg")
|| ($_FILES["file"]["type"] == "image/jpg")
|| ($_FILES["file"]["type"] == "image/pjpeg")
|| ($_FILES["file"]["type"] == "image/x-png")
|| ($_FILES["file"]["type"] == "image/png"))
&& ($_FILES["file"]["size"] < 20000)
&& in_array($extension, $allowedExts)) {
    if ($_FILES["file"]["error"] > 0) {
        echo "Return Code: " . $_FILES["file"]["error"] . "<br>";
    } else {
        echo "Upload: " . $_FILES["file"]["name"] . "<br>";
        echo "Type: " . $_FILES["file"]["type"] . "<br>";
        echo "Size: " . ($_FILES["file"]["size"] / 1024) . " kB<br>";
        echo "Temp file: " . $_FILES["file"]["tmp_name"] . "<br>";
        if (file_exists("upload/" . $_FILES["file"]["name"])) {
            echo $_FILES["file"]["name"] . " already exists. ";
        } else {
            move_uploaded_file($_FILES["file"]["tmp_name"],
                "upload/" . $_FILES["file"]["name"]);
            echo "Stored in: " . "upload/" . $_FILES["file"]["name"];
        }
    }
} else {
    echo "Invalid file";
}
?>
```

## Wyświetlanie pliku php jako pliku graficznego

```
<?
header('Content-Type: image/jpeg');
header("Cache-Control: no-cache, must-revalidate"); // HTTP/1.1
header("Expires: Sat, 26 Jul 1997 05:00:00 GMT");

$pliki = scandir("obrazki");

$im = @imagecreatefromjpeg("obrazki/" .
$pliki[rand(2, count($pliki)-1)]);

imagejpeg($im);

imagedestroy($im);
?>
```

## Upload plików na serwer do katalogu i do bazy danych

```
CREATE TABLE upload (  
ID_upload INT NOT NULL AUTO_INCREMENT,  
name VARCHAR(30) NOT NULL,  
type VARCHAR(30) NOT NULL,  
size INT NOT NULL,  
content MEDIUMBLOB NOT NULL,  
PRIMARY KEY(id)  
);
```

## Upload plików na serwer do katalogu i do bazy danych

```
$fp      = fopen($tmpName, 'r');
$content = fread($fp, filesize($tmpName));
$content = addslashes($content);
fclose($fp);

$query = "INSERT INTO upload (name, size, type, content ) ".
"VALUES ('$fileName', '$fileSize', '$fileType', '$content')";

mysqli_query($link, $query);
```



## Upload plików na serwer do katalogu i do bazy danych

```
$query = "SELECT name, type, size, content " .  
        "FROM upload WHERE ID_upload = '{$_GET['id']}'";  
  
$result = mysqli_query($link, $query);  
list($name, $type, $size, $content) = mysqli_fetch_array($result);  
  
header("Content-length: $size");  
header("Content-type: $type");  
header("Content-Disposition: attachment; filename=$name");  
echo $content;  
  
exit;  
}
```

C. D. N.