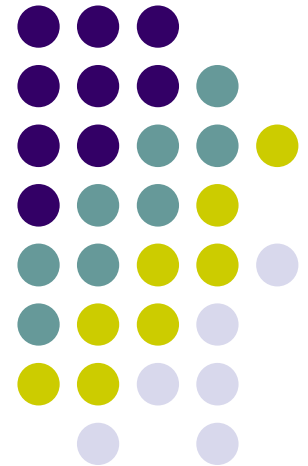
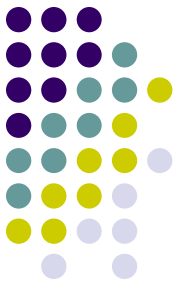


Τεχνολογίες Διαδικτύου

Server Side Scripting I
PHP



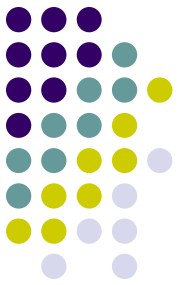


Εισαγωγή

PHP

PHP:Hypertext **P**reprocessor

- ❑ Μηχανή που συνοδεύει web servers όπως Microsoft IIS, Apache HTTP, NGINX και άλλοι
- ❑ Η PHP είναι **γλώσσα** προγραμματισμού για web εφαρμογές
- ❑ Apache: Διαθέσιμος σε UNIX και WINDOWS περιβάλλοντα.
- ❑ Παράδειγμα συστήματος:
 - ❑ Apache 2.4.41
 - ❑ PHP 7.3.11
 - ❑ MySQL Community Server 8.0.18



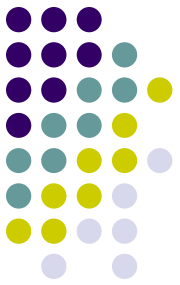
Εισαγωγή

- Απαιτούνται Βασικές Γνώσεις για
 - www, html και τα βασικά μέρη μιας web page
 - Κατανόηση μιας scripting γλώσσας όπως η JavaScript
 - Προγραμματισμός (C-like γλώσσα)
- Τα script αρχεία εκτελούνται στον server
- Τα αρχεία έχουν αντίστοιχα κατάληξη .php



Web server

- Για τις ανάγκες του μαθήματος, web server το PC σας:
- Windows
 - <http://www.wampserver.com/en/>
 - <https://www.apachefriends.org>
- Mac
 - <https://www.mamp.info>
 - <https://www.apachefriends.org>
- Unix/Linux
 - Συνήθως 2-3 εντολές
- Ας δούμε λίγα παραπάνω για το WAMP

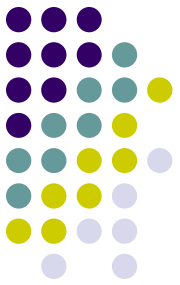


Αφού μάθαμε τόσα, γιατί και PHP;

Παράδειγμα

- Μία πιτσαρία θέλει να φτιάξει μια απλή web εφαρμογή ώστε οι πελάτες της να μπορούν να παραγγείλουν online φαγητό.
- Οι παραγγελίες θα μπορούν να πραγματοποιούνται είτε από επισκέπτες είτε από εγγεγραμμένους χρήστες.
- Στην περίπτωση του εγγεγραμμένου χρήστη, θα διατηρείται ιστορικό παραγγελιών με σκοπό την εμφάνιση προσφορών ή επανάληψη παραγγελίας.

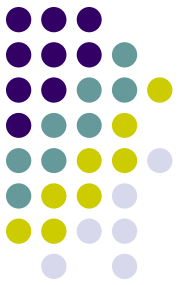
Πώς θα υλοποιηθεί με αυτά που έχω μάθει ήδη;



Εισαγωγή

Τι μπορούν να κάνουν τα ServerSide Scripts

- ✓ Να τροποποιήσουν **δυναμικά** το περιεχόμενο μιας σελίδας
- ✓ Απαντούν σε ερωτήματα και δεδομένα χρηστών από φόρμες HTML
- ✓ Δίνουν πρόσβαση σε δεδομένα και Βάσεις Δεδομένων και επιστρέφουν αποτελέσματα στον Browser
- ✓ Διαμορφώνουν τη σελίδα για προσωποποιημένη χρήση



Εισαγωγή

Τι μπορούν να κάνουν τα Server Side Scripts

- ✓ Παρέχει ασφάλεια, γιατί ο server-side κώδικας δεν είναι εμφανής στον πελάτη-Browser
 - ✓ Ας δούμε το multiple choice test
 - ✓ <http://athos.ceid.upatras.gr/intech/multiple/>
- ✓ Οι επιστρεφόμενες σελίδες είναι HTML (όχι απαραίτητα valid, εξαρτάται από τον χρήστη), μπορούν να τις δουν όλοι οι browsers
- ✓ «Μειονέκτημα»: Server side κώδικας με bugs
→ Θέματα ασφάλειας (hacked)

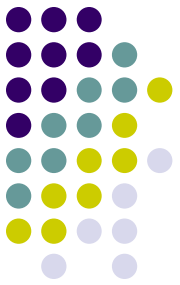


Server Side Scripting

- ASP και ASP.NET
- Python
- Java/ JSP
- Ruby
- NodeJS (server side JavaScript)
- PHP Frameworks (π.χ. Laravel, CakePHP)
- PHP CMS/ Frameworks (π.χ. Drupal – site CEID)

Είναι καλύτερη η PHP???

“...religious questions...”



Παραδείγματα Χρήσης

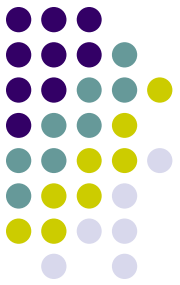
.PHP

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

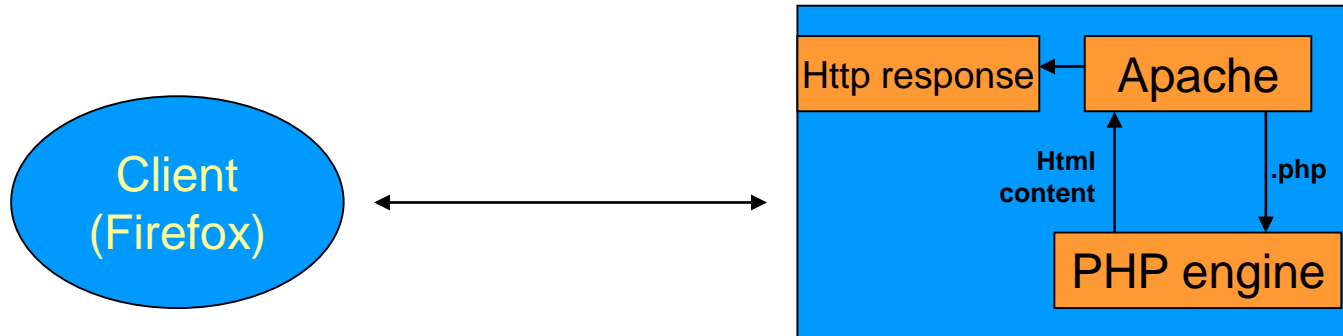
```
<?php  
echo "Hello, World!";  
?>  
</b>
```

```
<?php  
echo " <b> Hello, World! </b>"  
?>
```

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



Πώς δουλεύει αυτό το απλό script



Πως Δουλεύουν τα ServerSide Scripts

- Όταν ο Browser ζητά ένα HTML αρχείο, ο server του επιστρέφει το αρχείο αυτό ως έχει, βάσει του πρωτοκόλλου
- Όταν ζητηθεί ένα PHP, ο server περνά το αίτημα στη μηχανή PHP
 - Κατάληξη .php, ας δούμε το test.html
- Η μηχανή PHP διαβάσει τα script αρχεία γραμμή-γραμμή και εκτελεί των κώδικα
- Τέλος, το αρχείο επιστρέφεται στον Browser ως απλή HTML



Παραδείγματα Χρήσης

.PHP

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

```
<?
```

```
$greeting="Hello ";
```

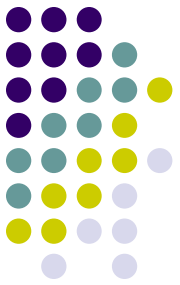
```
$num=3+2;
```

```
$num=$num+1;
```

```
echo $greeting.$num." people!";
```

```
?>
```

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



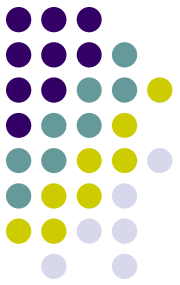
Παραδείγματα Χρήσης

.PHP

```
<?
$h=strftime("%H");

print "<p>".strftime("%m/%d/%Y %H:%M:%S %p")."</p>";

if ($h<12)
    print "Kalhmera";
else if ($h==12)
    print "Kalo mesimeri";
else
    print "Kalo apogeyma";
?>
```



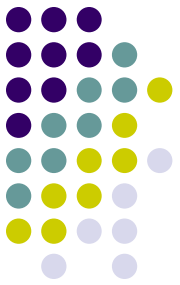
Παραδείγματα Χρήσης

.PHP

<?

```
for ($i=1; $i<=6; $i=$i+1){  
    print "<h".$i.">This is header ".$i."</h".$i.">";  
}
```

?>



Παραδείγματα Χρήσης

.PHP

```
<?
$students[0]="Kiki";
$students[1]="Eirini";

for ($i=0; $i<=1; $i=$i+1){
    print $students[$i]."<br>";
}

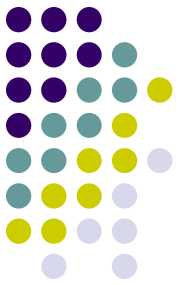
?>
```

.PHP

```
<?
$students[0]="Kiki";
$students[1]="Eirini";

print_r($students);
}

?>
```



Παραδείγματα Χρήσης

.PHP

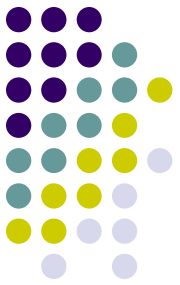
```
<?
$students[0]="Antreas";
$students[1]="Fanis";
$students[2]="Kwstas";

foreach ($students as $name){

    print $name."<br>";

}

?>
```



Παραδείγματα Χρήσης

.PHP

```
<h1> Footer Test Page</h1>
```

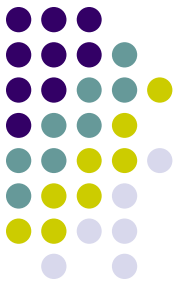
```
<p>
```

This is the first paragraph.

```
</p>
```

```
<p>2nd p</p><p>3rd p</p><p>4th p</p>
```

```
<? include ("footer.htm") ?>
```



Παραδείγματα Χρήσης

Σχετικά με τα προγράμματα Web Servers

- Παρουσίαση των HTTP Headers
 - PHP: [Headers.php](#)
[Headers.html](#) (*HTTP_REFERER*)
[phpinfo\(\)](#)



Παραδείγματα Χρήσης

.PHP

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

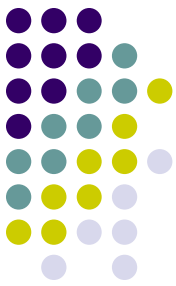
```
<?  
$strUser=$_SERVER["HTTP_USER_AGENT"];  
print $strUser;  
?>
```

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

Παράδειγμα:

<http://www.danasoft.com/vipersig.jpg>

Save info



Τετάρτη, 23 Μαρτίου 2016

ΑΝΑΖΗΤΗΣΗ



19°C

Πάμε κατευθείαν στον
Απρίλιο



Η Πρωτοποριακή Ελληνική Ηλεκτρονική Εφημερίδα

Το Κουλούρι

Πολιτική

Κοινωνία

Κόσμος

Επιστήμη

Οικονομία

Πολιτισμός

Αθλητισμός

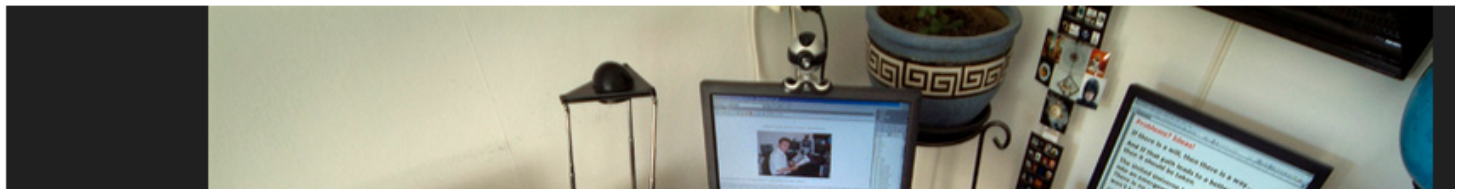
Νεκρός από πυρά αστυνομικών άοπλος webmaster στην Αριζόνα που δεν έβαζε προειδοποίηση για τα cookies

ΦΩΤΟΓΡΑΦΙΚΑ ΝΕΑ 17 ΦΕΒΡΟΥΑΡΙΟΥ 2016

EMAIL

FACEBOOK

TWITTER





Save info

- Το HTTP πρωτόκολλο είναι stateless
- Λύσεις:
- Server side λύσεις:
 - Sessions
- Client side λύσεις (browser)
 - Cookies ([cookie.php](#))
 - HTML5 web storage (“cookies version 2”)

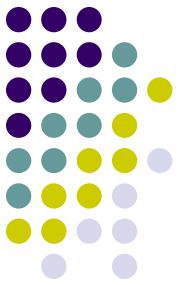
Sessions



Μεταβλητές Session

- Οι μεταβλητές Session χρησιμοποιούνται για τη διατήρηση πληροφορίας σε περίπτωση διαδοχικών προσβάσεων. Με αυτόν τον τρόπο είναι δυνατή η προσωποποίηση του περιεχομένου.
- Συνήθως η πληροφορία που αποθηκεύεται στις μεταβλητές αυτές είναι τα username & userid. Για να δημιουργηθεί μια session μεταβλητή, την αποθηκεύουμε σε ένα Session Object.
- Το session object χρησιμοποιείται για να αποθηκευτεί πληροφορία ή και να αλλαχτούν οι ρυθμίσεις ενός user session. Οι μεταβλητές που αποθηκεύονται στο session object κρατούν την πληροφορία για έναν χρήστη κάθε φορά και είναι διαθέσιμες σε όλες τις σελίδες μιας εφαρμογής.
- Τα sessions σε αντίθεση με τα cookies αποθηκεύονται στον server και όχι στον client.

Cookies VS Sessions



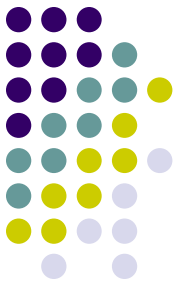
Cookies

- Τα cookies έχουν μεγάλο χρόνο ζωής
- Δεν υπάρχει πρόβλημα με web clustres

Sessions

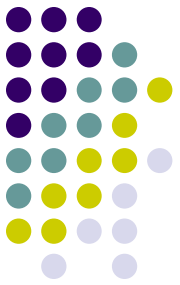
- Οι χρήστες δε θέλουν πληροφορία να αποθηκεύεται στο pc τους
- **Ο χρήστης δεν γνωρίζει, και ούτε μπορεί να δει, τι αποθηκεύεις στον server**
- Δεν χρειάζεται να μεταφέρεται η πληροφορία στον client
- Μεγαλύτερο μέγεθος από τα cookies

Φόρμα



.PHP

```
<form name="login_form" method="post" action="login.php">  
Username: <input type="text" name="username"><br>  
Password: <input type="password" name="password"><br>  
<input type="submit" value="Login">  
</form>
```



Φόρμα

\$_GET[]

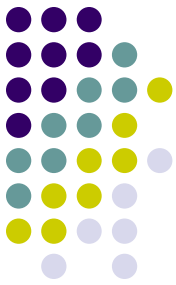
- Η πληροφορία που αποστέλλεται από τη φόρμα με τη μέθοδο GET είναι ορατή σε όλους (στο πεδίο διευθύνσεων). Θυμίζουμε ότι η GET περιορίζει την ποσότητα πληροφορίας που μπορεί να σταλεί

\$_POST[]

- Η πληροφορία που στέλνεται από τη φόρμα με αυτή τη μέθοδο είναι αόρατη στον πελάτη. Η μέθοδος POST δεν έχει περιορισμούς ως προς το μέγεθος της πληροφορίας.

Φόρμα

[Login.PHP](#)

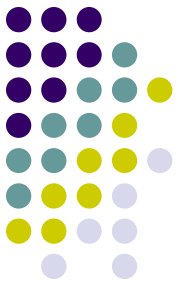


```
<?php
session_start();
?>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-7"/>
</head>
<?php

if (isset($_SESSION['session_username']))
{
    echo "Έχεις κάνει ήδη login <b>".$_SESSION['session_username']. "</b>! Μια φορά αρκεί.";
    echo "<br><a href='logoff.php'>Log off</a>";
}
else
{
    if (strlen($_POST['username']) < 6 || strlen($_POST['password']) < 6)
    {
        echo "<h2>Δεν έχετε δώσει τα σωστά δεδομένα!</h2>";
        echo "<br /><a href='login_form.php'><img src='meme.png' /></a>";
    }
    else
    {
        $_SESSION['session_username'] = $_POST['username'];
        echo "Welcome <b>".$_SESSION['session_username'];
        echo "<br><a href='logoff.php'>Log off</a>";
    }
}

?>
```

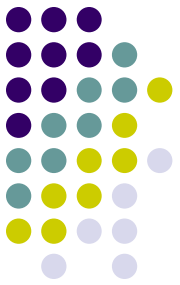
Φόρμα



.PHP

```
<?
    session_start();
?>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-7"/>
</head>
<?
    session_unset();
?>
<p>Ευχαριστούμε που μας επισκεφθήκατε</p>
</html>
```

//Ας δούμε και το [session-details.php](#)



Links

- www.php.net
- www.phpbuilder.com
- www.w3schools.com/php
- www.wampserver.com
- <http://www.apachefriends.org/en/xampp.html>