

- **Με ποιες από τις παρακάτω εντολές/τρόπους μπορώ να φορτώσω «αυτόματα» ξανά μία ιστοσελίδα;**

- `<meta http-equiv="Content-Type" content="text/html; />`
- `<META HTTP-EQUIV="refresh" CONTENT="15">`.
- Χρησιμοποιώ `SetTimeout` και καλώ την εντολή `window.open()`
- Χρησιμοποιώ `SetTimeout` και καλώ την εντολή `history.back()`

- **Ποια από τα παρακάτω αποτελούν χαρακτηριστικά του HTTP πρωτοκόλλου:**

- Ταχύ
- Σύνθετο
- «Έχει μνήμη»
- Είναι **stateless**

- **Τι από τα παρακάτω μπορεί να δημιουργήσει κάποιος στην HTML με τη σήμανση INPUT**

- **Image**
- **Check boxes**
- List box
- **Text boxes**

- **Σε εφαρμογές AJAX:**

- Μέσω του XMLHttpRequest object ο JavaScript κώδικας που τρέχει στο browser μπορεί να επικοινωνεί με τον server
- Μέσω του session variables ο PHP κώδικας που τρέχει στο servers μπορεί να επικοινωνεί με το client
- Μέσω του XMLHttpRequest ο PHP κώδικας που τρέχει στο browser μπορεί να επικοινωνεί με το server

- **Ποιες από τις παρακάτω τεχνολογίες δεν χρησιμοποιείται στο Ajax;**

- XMLHttpRequest
- JavaScript
- **Java**
- DOM

- **Όταν μια εφαρμογή μας σε php πρέπει να επικοινωνήσει με την mysql**

- Δεν μπορούμε να χρησιμοποιήσουμε IIS, διότι ο IIS επικοινωνεί μόνο με Microsoft Access και Microsoft SQL Server.
- Πρέπει υποχρεωτικά να χρησιμοποιήσουμε τον apache web server διότι μόνο αυτός έχει την δυνατότητα σύνδεσης με mysql
- Δεν παίζει ρόλο ποιόν web server χρησιμοποιούμε. Είναι ανεξάρτητη η σύνδεση php-mysql με τον web server.

- **Όταν εκτελούμε μέσα από php ένα sql query τότε:**

- Θα πρέπει να διαβάσουμε τον πίνακα αποτελεσμάτων με την κατάλληλη εντολή και να κάνουμε μέσα από το script "print"
- Τα αποτελέσματα εκτυπώνονται αυτόματα στην οθόνη του χρήστη
- Θα πρέπει να ελέγξουμε αν η εντολή έδωσε μήνυμα λάθους και να ενημερώσουμε κατάλληλα τον χρήστη

- **Ποιος/οί είναι/οι λόγος/οι που προτιμάται η mysql σε διαδικτυακές εφαρμογές;**
 - Είναι δωρεάν
 - Έχει μεγαλύτερη απόδοση από τις εμπορικές βάσεις δεδομένων
 - Είναι εύκολη στην εγκατάσταση και χρήση.
- **Όταν μια εφαρμογή php επικοινωνεί με mysql**
 - Προτιμούμε να χρησιμοποιούμε prepared statements επειδή είναι πιο γρήγορα σε περιπτώσεις που κάποιο ερώτημα SQL πρόκειται να εκτελεστεί πολλές φορές, μεταβάλλοντας μόνο κάποιες παραμέτρους σε κάθε εκτέλεση
 - Δεν προτιμούμε τα prepared statements επειδή δεν υλοποιούνται στην αντίστοιχη βιβλιοθήκη της php
 - Δεν προτιμούμε τα prepared statements επειδή είναι αργά (εκτελείται η εντολή prepare και μετά η εντολή execute)- άρα δύο εντολές
 - Προτιμούμε να χρησιμοποιούμε prepared statements επειδή μας δίνουν καλύτερη και ξεκάθαρη σύνταξη. Δεν χρειάζεται να χρησιμοποιήσουμε μεταβλητές της mysql μέσα στον SQL κώδικα. Η χρήση μεταβλητών/ορισμάτων γίνεται με το binding
 - Προτιμούμε να χρησιμοποιούμε prepared statements επειδή είναι πιο ασφαλή σε πιθανή επίθεση τύπου SQL injection
- **Ποιες είναι οι βιβλιοθήκες της php μέσω των οποίων μπορούμε να επικοινωνήσουμε με την βάση mysql;**
 - PDO
 - mysqli
 - JDBC
 - Ado.net
- **Ποιο/α από τα παρακάτω είναι απολύτως ασφαλή σε php, ως προς το ενδεχόμενο εκτέλεσης ανεξέλεγκτου κώδικα php από εξωτερικό χρήστη:**
 - require \$file;
 - require \$ REQUEST['file'];
 - require "myfile.php";
 - require "../lib/myfile.php";
- **Ένας χρήστης επισκέπτεται την σελίδα <http://www.example.com/test.php>**
 - Ο χρήστης μπορεί να δει τον κώδικα php αλλά όχι τα javascript που συμπεριλαμβάνονται
 - Ο χρήστης δεν μπορεί να δει τον κώδικα php αλλά μπορεί να δει τα javascript που συμπεριλαμβάνονται μόνο αν είναι ενσωματωμένα στην html
 - Ο χρήστης δεν μπορεί να δει τον κώδικα php αλλά μπορεί να δει τα javascript που συμπεριλαμβάνονται
 - Ο χρήστης μπορεί να ενεργοποιήσει τον debugger του browser και να δει ολόκληρο τον κώδικα του αρχείου test.php
- **Για να μπορεί μία σελίδα να κρατάει τον πίνακα SESSION σε php:**

- Ο browser του χρήστη πρέπει να μπορεί να αποθηκεύει session elements (μπορούμε να τα δούμε με την επιλογή του Debugger "Inspect Element")
- Ο χρήστης πρέπει να αποδεχθεί την πολιτική απορρήτου που εμφανίζει το Site για την αποθήκευση των cookies
- Ο browser του χρήστη πρέπει να αποθηκεύει cookies
- Ο php server πρέπει να συνδεθεί με την mysql
- **Δίνεται ο παρακάτω κώδικας**

```
$a = 1;
function x() { print $a; }
```

 - Η μεταβλητή \$a δεν είναι «ορατή» στην συνάρτηση x() αν δεν υπάρχει τρόπος πρόσβασης στην \$a μέσα από την x()
 - Η μεταβλητή \$a δεν είναι «ορατή» στην συνάρτηση x() μόνο αν την δηλώσουμε global \$a μέσα στην x() τότε αποκτάται πρόσβαση στην \$a
 - Η μεταβλητή \$a είναι «ορατή» στην συνάρτηση x()
- **Έστω ότι θέλουμε να δημιουργήσουμε μια registration form όπου ο χρήστης πληκτρολογεί τον επιθυμητό κωδικό. Ποια μέθοδο θα πρέπει να χρησιμοποιηθεί ώστε να διασφαλιστεί ότι ο κωδικός δεν θα γίνει φανερός;**
 - SESSION
 - GET
 - COOKIE
 - POST
- **Ένα URL εκτός από τα υποχρεωτικά τμήματα μπορεί να περιέχει και τα παρακάτω τμήματα:**
 - Port number
 - Ορίσματα/δεδομένα μεθόδου GET
 - Έκδοση browser
 - Διαδρομή φακέλου
 - Ορίσματα/δεδομένα μεθόδου POST
- **Ποιο είναι το output του παρακάτω κώδικα;**

```
<?php function test_params($a,$b,$arrOptionalParams=array()){
    $c = 'sat'; $d = 'mat';
    foreach($arrOptionalParams as $key => $value){
        ${$key} = $value;}

    echo "$a $b $c on the $d";}
test_params('The', 'dog', array('c' => 'stood', 'd' => 'donkey'));
```

 - The cat sat on the donkey
 - A dog stood on the mat
 - PHP Fatal error
 - The dog stood on the donkey
- **Ο κώδικας php τοποθετείται σε ένα ζευγάρι delimiter. Ποιο είναι το σωστό;**
 - <&>...</&>
 - <?php>...</?>
 - <?php...?>
 - <script>...</script>

- Ποιο είναι το output του παρακάτω κώδικα;

```
<?php function familyName($fname,$year=1950) {  
    echo "$fname Mcaffee. Born in $year";}  
familyName("Stale");?>
```

- Stale Mcaffee. Born in 1950
- Mcaffee. Bornin
- Mcaffee. Bornin 1950
- Stale Mcaffee. Bornin

- Ποια function χρησιμοποιούμε στην PHP για να διαπιστώσουμε αν μια μεταβλητή έχει οριστεί ή όχι;

- isset()
- unset()
- set()
- καμία από τις παραπάνω isos kai ayto

- Ποιος από τους παρακάτω τρόπους είναι σωστός για να ορίσουμε ένα array στη PHP;

- \$cars ="volvo","vw","toyota";
- \$cars =["volvo","vw","toyota"];
- \$cars= array("volvo","vw","toyota");
- \$cars= array["volvo","vw","toyota"];

- Ποιο-α από τα παρακάτω ισχύει-ουν αναφορικά με τη γραμμή κώδικα:

```
while($row= $result->fetch_assoc()) {...}
```

- Το row είναι ένα associative array με τόσα κελιά όσα και τα πεδία που ανακτήθηκαν από τη ΒΔ
- Η συγκεκριμένη εντολή δεν μπορεί να χρησιμοποιηθεί σε συνδυασμό με prepared i. Statement
- Το while loop θα εκτελεσθεί τόσες φορές όσες και οι εγγραφές που επιστράφηκαν από το ερώτημα στη ΒΔ
- Οι συναρτήσεις fetch_assoc() και fetch_array() είναι ισοδύναμες,

- Ποιο είναι το output του παρακάτω κώδικα;

```
<?php function invite(&$s) {$s= 'keep calm '}; $x = 'pass exams ';invite($x);echo $x;  
?>
```

- pass exams
- keep calm
- keep calm pass exams
- pass examskkeep calm

- Ποιο είναι το output του παρακάτω κώδικα;

```
<?php function my_sum($nums) {
```

```
$s = 2;
```

```
foreach($nums as $a) { $s+= $a;}
```

```
return $s;}
```

```
print my_sum( array(1,5,2,1)); ?>
```

- Δεν θα εκτελεστεί επειδή υπάρχει συντακτικό λάθος
- 21521
- 11
- 1521

- **Τα ορίσματα GET σε ένα url χωρίζονται μεταξύ τους με:**
 - &
 - #
 - %
 - ?
- **Ποια από τις παρακάτω προτάσεις είναι λάθος;**
 - Η τεχνολογία AJAX είναι μία client-side προσέγγιση για την επικοινωνία με διάφορες άλλες server-side τεχνολογίες (PHP, JSP, ASP, κτλ.)
 - Η τεχνολογία AJAX επιτρέπει να ενημερώνουμε ένα μέρος της σελίδας με δεδομένα από τον εξυπηρετητή χωρίς να χρειάζεται επαναφόρτωση όλης της σελίδας
 - Η τεχνολογία AJAX βασίζεται στο αντικείμενο XMLHttpRequest
 - Η τεχνολογία AJAX απαιτεί την εκμάθηση μίας νέας ολόκληρης γλώσσας client-side προγραμματισμού
- **Η εντολή: xmlHttp.onreadystatechange= setName;**
 - Ορίζει τη JavaScript συνάρτηση setName που θα κληθεί, όταν έρθει η απάντηση του εξυπηρετητή στη χρήση/κλήση AJAX
 - Ορίζει τη PHP συνάρτηση setName που θα κληθεί, όταν έρθει η απάντηση του εξυπηρετητή
 - Ορίζει την ιδιότητα setName που θα πάρει τιμή, όταν έρθει η απάντηση του εξυπηρετητή
- **Ποιο από τα παρακάτω είναι αλήθεια;**
 - Το AJAX μπορεί να λειτουργήσει με web εφαρμογή
 - Το AJAX είναι τεχνολογία ανεξάρτητη από πλατφόρμες
 - Το AJAX είναι τεχνολογία που εξαρτάται από την πλατφόρμα
 - Το AJAX μπορεί να λειτουργήσει μόνο με php στην πλευρά του server
- **Τα nodejs είναι:**
 - Μία ειδική έκδοση της javascript που τρέχει στον browser για realtime εφαρμογές
 - Μια μηχανή javascript που τρέχει σε οποιοδήποτε λειτουργικό
 - Η μηχανή javascript που τρέχει στον Chrome
- **Τι από τα παρακάτω ισχύει για ένα Restfull Web API(RestAPI);**
 - Χρησιμοποιούνται οι μέθοδοι του HTTP: GET, POST, PUT, DELETE
 - Πρέπει υποχρεωτικά να χρησιμοποιείται μόνο η μέθοδος GET.
 - Το API δεν σχετίζεται με τις HTTP μεθόδους
 - Χρησιμοποιούνται μόνο οι μέθοδοι του HTTP: GET, POST. Δεν υπάρχει PUT, DELETE
- **Σε ένα REST API:**
 - Για κάθε Resource ορίζεται URI (π.χ. /cars) και οι μέθοδοι που θα χρησιμοποιούνται (π.χ GET, POST, DELETE)
 - Για την αντιστοιχία URI-Resource καλό είναι να χρησιμοποιούνται ουσιαστικά και όχι ρήματα, π.χ "/cars" αλλά όχι "/getCats"
 - Για κάθε ενέργεια πρέπει να οριστεί ένα URL, π.χ /getCats,

/deleteCar,/searchCars,/updateCars

- **Σε σχέση με τα APIs:**

- Ένα Web-API είναι υποχρεωτικά REST API
- Ένα Web-API βασίζεται στο HTTP πρωτόκολλο
- Ένα REST API δεν μπορεί να χρησιμοποιηθεί σε web εφαρμογές
- Ένα Web-API μπορεί και να μην είναι REST API

- **Ποιο-α από τα παρακάτω ισχύουν σχετικά με τη γλώσσα JSON:**

- Η σύνταξη ενός αντικειμένου ξεκινάει με [και τελειώνει με]
- Το key είναι πάντα string
- Ένα αντικείμενο μπορεί να περιλαμβάνει ένα άλλο αντικείμενο
- Τα ζευγάρια key/value χωρίζονται μεταξύ τους με κόμμα

- **Ποιο-α από τα παρακάτω ισχύουν σχετικά με τη γλώσσα JSON:**

- Το key χωρίζεται από το value με κόμμα
- Τα JSON values μπορεί να είναι Strings, Numbers, Booleans, Objects, Arrays, null
- Στο τέλος κάθε ζευγαριού key/value πρέπει να μπαίνει το ελληνικό ερωτηματικό
- Οι πίνακες μπορούν να περιλαμβάνουν στοιχεία διαφορετικού τύπου

- **Ποια-ες προτάσεις ισχύουν σχετικά με τον παρακάτω κώδικα;**

```
var a = '{ "employees" : [ { "firstName":"John",  
"lastName":"Doe" }, { "firstName":"Anna", "lastName":"Smith" },  
{ "firstName":"Peter", "lastName":"Jones" } ]}';  
var x = JSON.parse(a);document.write(x.employees[2].lastName);
```

- Η σύνταξη JSON δεν είναι σωστή. Όποτε δεν θα γίνει parse ο JSON κώδικας
- Θα γράψει στην ιστοσελίδα τη λέξη "Jones"
- Για να γίνει parse το JSON πρέπει υποχρεωτικά να χρησιμοποιηθεί η συνάρτηση eval
- Η JSON σύνταξη ορίζει ένα αντικείμενο. Το key είναι το "employees" ενώ το value είναι ένας πίνακας αντικειμένων

- **Ο παρακάτω κώδικας: var elems = \$("#george p") ;**

- Θα επιστρέψει όλα τα στοιχεία paragraphs που βρίσκονται κάτω από ένα στοιχείο της κλάσης george
- Θα επιστρέψει όλα τα στοιχεία paragraphs και όλα τα στοιχεία με το id "george"
- Θα επιστρέψει όλα τα paragraphs και όλα τα στοιχεία της κλάσης george
- Θα επιστρέψει όλα τα στοιχεία paragraphs που βρίσκονται κάτω από ένα στοιχείο με το id "george"

- **Ποιο από τα παρακάτω ισχύει;**

- document.getElementById("myid")==\$("#myid")[0];
- document.getElementById("myid")==\$("#myid");
- document.querySelectorAll("p")[0]==\$("#p")[0];
- document.getElementById("myid")==\$("#myid").get(0)

- **Η JQuery είναι :**

- Μια Βιβλιοθήκη για την γλώσσα Javascript
- Μια γλώσσα προγραμματισμού ειδικά σχεδιασμένη για τον Apache Web Server
- Μια γλώσσα προγραμματισμού
- Μια κλάση Java με την οποία πραγματοποιούμε queries σε μια βάση δεδομένων

- **Μέσα σε ένα αρχείο json:**

- Τα Objects σηματοδοτούνται με '{ }' Οι πίνακες σηματοδοτούνται με '['']
- **Για το πρωτόκολλο HTTP το POST είναι:**
 - Μέθοδος Request
 - Πεδίο μέσα στο κυρίως σώμα του Request
 - Πεδίο μέσα στις κεφαλίδες του Request
- **Ένα HTTP Response (απάντηση) περιέχει έναν κωδικό (Response status code)**
 - Υπάρχουν προκαθορισμένοι κωδικοί και η εξήγηση τους και ο προγραμματιστής μπορεί να ορίσει και τους δικούς του
 - Ο κωδικός αυτός μπορεί να είναι οποιοδήποτε string
 - κωδικός αυτός μπορεί να είναι οποιοδήποτε νούμερο θέλουμε
- **`$(function() { // do stuff with the DOM });` Ο πανω κωδικας είναι:**

Ο κώδικας μέσα στην συνάρτηση θα εκτελεστεί μετά το φόρτωμα της σελίδας
Είναι η κλήση της συνάρτησης με όνομα \$ και όρισμα μια anonymous function

 - Λάθος. Δεν υπάρχει κάτι τέτοιο.
 - Μια anonymous function.
- **Στο Μοντελο MVC:**
 - Το model περιέχει τη δομή των δεδομένων και τις απαραίτητες μεθόδους ανάκτησης αυτών
- **Ένα Web URL μπορούμε να το κατεβάσουμε:**
 - Υλοποιώντας ένα πρόγραμμα που μπορεί να επικοινωνεί με το πρωτόκολλο HTTP
 - Με πολλές εντολές από την γραμμή εντολών όπως wget, curl, GET κλπ
 - Με τη βοήθεια ενός browser
 - Μονο με έναν Browser και κανέναν άλλον τρόπο
- **Ποιο/Ποια τμήματα περιεχει ένα HTTP Request:**
 - Μια γραμμή που περιέχει method+URL+protocol version
 - Το κυρίως σώμα του Request
 - Τις κεφαλίδες του Request
 - Το SOAP Envelope
- **Πως ορίζουμε ένα cookie στην PHP:**
 - `setcookie()`
- **Όταν σε ένα URL οι εξής χαρακτήρες είναι ειδικοί:**
 - Το ``+'' σημαίνει Space
 - Το ``+''' σημαίνει συνένωση Strings
 - Το ``&'' διαχωρίζει τις παραμέτρους μεταξύ τους
 - Το ``%''' ακολουθείται από έναν 2ψηφιο δεκαεξαδικό και αντιστοιχεί στον UTF-8 χαρακτήρα με τον συγκεκριμένο κωδικό
 - Το ``?''' διαχωρίζει τις παραμέτρους μεταξύ τους
- **Ποιο είναι σωστο ?**
 - `$("#p").click(do_it); function do_it(event){alert('click');}`
 - `$("#p").click(function(){alert('click');});`
 - `$("#p").click(do_it() { alert('click') ; }) ;`
 - `$("#p").click(do_it); function do_it() {alert('click') ; }`

- **Ποιο-α από τις παρακάτω εφαρμογές είναι εξυπηρετητές ιστού:**
 - MOSAIC
 - IIS
 - Ngnx
 - Nexus
- **Όταν μέσα από ένα PHP αρχείο παραγουμε JSON μορφή:**
 - Δεν πρέπει με κανέναν τρόπο να ορίσουμε content type
 - Πρέπει να θέσουμε header('Content-type: application/json').
 - Προτείνουμε να θέσουμε header('Content-type: text/plain') διότι το json είναι text. Αν θέσουμε header('Content-type: application/json') δεν θα βλέπουμε σωστά τα περιεχόμενα.
 - Πρέπει οπωσδήποτε να ορίσουμε header('Content-type: xxx/yyy') ακόμη και αν το xxx/yyy δεν αντιστοιχεί στο json
- **Ποιο/ά από τα παρακάτω είναι σωστό/ά;**
 - `$("#button").click(function(){$("#p").hide();});`
 - `$("#button").click(function hide(){});`
 - `$("#button").click(function hide());`
 - `$("#button").click(this.hide());` iso kai ayto
- **Στο παρακάτω php παράδειγμα `$a = 'hello' ; $hello = 'a'; print "${$a}";`**
 - Θα τυπωθεί "\$hello"
 - Θα τυπωθεί "\$a"
 - Θα τυπωθεί "a"
 - Θα τυπωθεί "hello"
- **Με βάση το HTTP , πληροφορίες για την γλώσσα(Ελληνικά, Αγγλικά, κτλ) μπορεί να υπάρχουν :**
 - Στο header μιας HTTP Response (απάντησης)
 - Στο body μιας HTTP Response (απάντησης)
 - Στο header μιας HTTP Request (αίτησης)
 - Στο body μιας HTTP Request (αίτησης)
- **Ποιό/ά από τα παρακάτω είναι σωστό/ά(jquery)**
 - `$('#li').html('This is an element');`
 - `$('#li').html = 'This is an element';`
 - `$('#li').innerHTML='This is an element';`
 - `$('#li').text='This is an element';`
- **Εάν κάνουμε pull μεταξύ 2 repositories ΔΥΟ φορές συνεχόμενα:**
 - Τα 2 repositories σίγουρα ΔΕΝ θα συγχρονιστούν πλήρως ανεξάρτητα από τις άλλες αλλαγές που υπήρχαν
 - Τα 2 repositories θα συγχρονιστούν πλήρως
 - Δεν χρειάζεται 2° pull. Ακόμη και με το 1° τα 2 repositories θα συγχρονιστούν πλήρως
 - Τα 2 repositories μπορεί να μην συγχρονιστούν πλήρως ανάλογα με τις αλλαγές που υπήρχαν

- Σε ένα **Version Control System** διαγράφονται αυτόματα παλιές εκδόσεις κώδικά όταν :
 - Παρέλθει χρόνος 2 μηνών
 - Ποτέ
 - Αν κάποιο αρχείο δεν χρησιμοποιείται
 - Παρέλθει ο χρόνος που έχει ορίσει ο Administrator για το κάθε project
- **Ποιό/ά από τα παρακάτω είναι σωστό/ά(jquery)**
 - `var t = <p>Test</p>;`
`$("#body").append(t);`
 - `$("#body").append("<p>Test</p>");`
 - `$("#body").append(<p>Test</p>);`
 - `var t = "<p>Test</p>";`
`$("#body").append(t);`
 - `var t = $('<p></p>') .text ('Test');`
`$("#body") .append(t);`
- **Ποιές από τις παρακάτω jquery μεθόδους έχουν σχέση με Ajax;**
 - `$('#maincontent').load('shopinfo.php');`
 - `$('#maincontent').html('shopinfo.php');`
 - `$.json_parse('#maincontent');`
 - `$('#maincontent').get('shopinfo.php' {success: function() { // do something}});`
 - `$ajax('shopinfo.php', {success: function() { // do something}});`
- **Με ένα σύστημα VCS(Version Control System):**
 - Ελέγχεται εάν σε κάποιον υπολογιστή είναι εγκατεστημένη η σωστή έκδοση του λογισμικού που θέλουμε
 - Ο χρήστης μπορεί να ελέγχει τις εκδόσεις των προγραμμάτων που κατεβάζει στον υπολογιστή του
 - Μπορούν να συνεργαστούν πολλοί προγραμματιστές ώστε να γράφουν κώδικά στα ίδια αρχεία ενός project.
 - Ο προγραμματιστής διαχειρίζεται τις εκδόσεις του κώδικα του
- **Ποιο είναι το output του παρακάτω κώδικα;**

```
<?php function test_params($a,$b,$arrOptionalParams=array()){
```

```
    $c = 'sat'; $d = 'mat';
```

```
    foreach($arrOptionalParams as $key => $value){
```

```
        ${$key} = $value;}
```

```
    echo "$a $b $c on the $d";}
```

```
test_params('A', 'dog', array('c' => 'stood')); ?>
```

- The cat sat on the donkey
- A dog stood on the mat
- PHP Fatal error
- The dog stood on the donkey
- **Ποιό από τα παρακάτω ισχύει ;**
 - JSON είναι μια μορφή για ανταλλαγή δεδομένων
 - Η JSON είναι μορφή της XML
 - Το JSON φορμάτ μπορεί να γίνει parse από τον js controller
 - Με json μπορούμε να μεταφέρουμε ένα αντικείμενο από την php του server σε javascript του client
 - Το JSON φορμάτ είναι συμβατό με js αλλά όχι με php
- **Οι Web Servers δεν έχουν ενσωματωμένη την δυνατότητα να εκτελούν προγράμματα php για να παράγουν δυναμικές ιστοσελίδες. Αυτό γίνεται με την εγκατάσταση του κατάλληλου module.**
 - Εξαρτάται από τον server που χρησιμοποιούμε
 - Λάθος
 - Σωστό
- **Έστω ότι χρησιμοποιούμε κάποιο Distributed Version Control System . Όταν κάνουμε update:**
 - Αποθηκεύονται αλλαγές στο τοπικό Repository από το Working Copy
 - Μεταφέρονται οι αλλαγές από το τοπικό Repository στο απομακρυσμένο.
 - Μεταφέρονται οι αλλαγές από το τοπικό Repository στο Working Copy
 - Μεταφέρονται οι αλλαγές από το απομακρυσμένο Repository στο τοπικό Repository
- **Πότε υπάρχει περίπτωση να χάσω την “συνέχεια” του ιστορικού ενός αρχείου σε ένα VCS;**
 - Όταν κάνω rename το αρχείο
 - Όταν κάνω Delete το αρχείο
 - Όταν κάνω commit το αρχείο και χάνονται οι παλιές εκδόσεις του.
- **Ποιό από τα παρακάτω ισχύει;**
 - Η json δεν έχει εκδόσεις
 - Η json2.0 είναι καλύτερη από την json1.0
 - Όταν λέμε json χωρίς σήμανση έκδοσης , εννοούμε την json1.0
- **Το GIT είναι :**
 - Version Control System 1^{ης} γενιάς
 - Distributed Version Control System
 - Version Control System (Centralized