

Ερωτήσεις - Απαντήσεις

ΑΔΙΣΕ - ΔΙΠΑΕ

Quiz για εξάσκηση:

- [Quiz \(Μόνο Βασική Ύλη\)](#)
- [Quiz \(Εξτρα Πιθανή Ύλη\)](#)

Έξτρα βοηθητικό Υλικό:

- [GTI - MegaPack.docx](#), καθώς όλες οι γνώσεις του GTI θεωρούνται ήδη γνωστές, και επίσης μερικές ερωτήσεις τυχαίνει να είναι ολόιδιες.

Βασική Ύλη.....	2
.NET.....	17
XML.....	18
Laravel.....	22
React JS.....	23
Spring.....	25
Vue.js.....	27
Node.js.....	28
WebSockets.....	30
ThreeJS.....	32
Livewire.....	33
TailwindCSS.....	35
Alpine.js.....	37
FastAPI.....	38

Βασική Ύλη

1. Με ποια/-ες από τις παρακάτω εντολές/τρόπους μπορώ να φορτώσω «αυτόματα» ξανά μία ιστοσελίδα;
 - a. `<meta http-equiv="Content-Type" content="text/html;/>`
 - b. **`<META HTTP-EQUIV="refresh"CONTENT="15">`.**
 - c. Χρησιμοποιώ `SetTimeout` και καλώ την εντολή `window.open()`
 - d. Χρησιμοποιώ `SetTimeout` και καλώ την εντολή `history.back()`
2. Ποια από τα παρακάτω αποτελούν χαρακτηριστικά του HTTP πρωτοκόλλου:
 - a. Ταχύ
 - b. Σύνθετο
 - c. «Έχει μνήμη»
 - d. **Είναι stateless**
3. Τι από τα παρακάτω μπορεί να δημιουργήσει κάποιος στην HTML με τη σήμανση INPUT;
 - a. Image
 - b. **Checkboxes**
 - c. List box
 - d. **Text boxes**
4. Τι σημαίνει η δήλωση DTD;
 - a. Definition type document
 - b. **Document type definition**
 - c. Data type definition
 - d. Document type data
5. Σχετικά με την XML ποιο από τα παρακάτω είναι σωστά;
 - a. Η XML δεν προσφέρεται για διαλειτουργικότητα (interoperability) μεταξύ διαδικτυακών προγραμμάτων χωρίς το πρότυπο XSLT.
 - b. **Σχεδιάστηκε για να περιγράψει τη δομή δεδομένων.**
 - c. **Είναι κάτι περισσότερο από γλώσσα σήμανσης, μπορεί να λειτουργήσει ως μία μεταγλώσσα (metalanguage), δηλαδή μία γλώσσα που χρησιμοποιείται για να καθορίσει νέες γλώσσες.**
 - d. Σχεδιάστηκε για να αντικαταστήσει την HTML.
6. Όσον αφορά τα συστήματα διαχείρισης περιεχομένου (CMS) τι από τα παρακάτω είναι σωστά;
 - a. Δεν μπορώ να χρησιμοποιήσω τεχνικές/κώδικα AJAX
 - b. Είναι πιο ασφαλή από ένα «hand-made» δικτυακό τόπο
 - c. **Έχουν σαν στόχο να διαχωρίσουν το περιεχόμενο ενός site από τον τρόπο εμφάνισης του περιεχομένου**

- d. Μπορώ να τα επεκτείνω χρησιμοποιώντας υπάρχουσες βιβλιοθήκες που υπάρχουν σε διάφορες μορφές.**
7. Σε εφαρμογές AJAX:
- a. Μέσω του XMLHttpRequest object ο JavaScript κώδικας που τρέχει στο browser μπορεί να επικοινωνεί με τον server**
 - b. Μέσω του session variables ο PHP κώδικας που τρέχει στο servers μπορεί να επικοινωνεί με το client
 - c. Μέσω του XMLHttpRequest ο PHP κώδικας που τρέχει στο browser μπορεί να επικοινωνεί με το server
8. Ποιες από τις παρακάτω τεχνολογίες **δεν** χρησιμοποιείται στο Ajax;
- a. XMLHttpRequest
 - b. JavaScript
 - c. Java**
 - d. DOM
9. Όταν μια εφαρμογή μας σε php πρέπει να επικοινωνήσει με την mysql:
- a. Δεν μπορούμε να χρησιμοποιήσουμε IIS, διότι ο IIS επικοινωνεί μόνο με Microsoft Access και Microsoft SQL Server.
 - b. Πρέπει υποχρεωτικά να χρησιμοποιήσουμε τον apache web server διότι μόνο αυτός έχει την δυνατότητα σύνδεσης με mysql
 - c. Δεν παίζει ρόλο ποιόν web server χρησιμοποιούμε. Είναι ανεξάρτητη η σύνδεση php-mysql με τον web server.**
10. Όταν εκτελούμε μέσα από php ένα sql query τότε:
- a. Θα πρέπει να διαβάσουμε τον πίνακα αποτελεσμάτων με την κατάλληλη εντολή και να κάνουμε μέσα από το script "print"
 - b. Τα αποτελέσματα εκτυπώνονται αυτόματα στην οθόνη του χρήστη
 - c. Θα πρέπει να ελέγξουμε αν η εντολή έδωσε μήνυμα λάθους και να ενημερώσουμε κατάλληλα τον χρήστη**
11. Ποιος/οί είναι ο/οι λόγος/οι που προτιμάται η mysql σε διαδικτυακές εφαρμογές;
- a. Είναι δωρεάν**
 - b. Έχει μεγαλύτερη απόδοση από τις εμπορικές βάσεις δεδομένων
 - c. Είναι εύκολη στην εγκατάσταση και χρήση.**
12. Όταν μια εφαρμογή php επικοινωνεί με mysql:
- a. 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**
 - b. Δεν προτιμούμε τα prepared statements επειδή δεν υλοποιούνται στην αντίστοιχη βιβλιοθήκη της php
 - c. Δεν προτιμούμε τα prepared statements επειδή είναι αργά (εκτελείται η εντολή prepare και μετά η εντολή execute)- άρα δύο εντολές
 - d. 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

e. Προτιμούμε να χρησιμοποιούμε prepared statements επειδή είναι πιο ασφαλή σε πιθανή επίθεση τύπου SQL injection

13. Ποιες είναι οι βιβλιοθήκες της php μέσω των οποίων μπορούμε να επικοινωνήσουμε με την βάση mysql;

- a. PDO
- b. mysqli
- c. JDBC
- d. Ado.net

14. Ποιο/α από τα παρακάτω είναι απολύτως ασφαλή σε php, ως προς το ενδεχόμενο εκτέλεσης ανεξέλεγκτου κώδικα php από εξωτερικό χρήστη:

- a. require \$file;
- b. require \$_REQUEST['file'];
- c. **require "myfile.php";**
- d. **require "../lib/myfile.php";**

15. Ένας χρήστης επισκέπτεται την σελίδα <http://www.example.com/test.php>

- a. Ο χρήστης μπορεί να δει τον κώδικα php αλλά όχι τα javascript που συμπεριλαμβάνονται
- b. Ο χρήστης δεν μπορεί να δει τον κώδικα php αλλά μπορεί να δει τα javascript που συμπεριλαμβάνονται μόνο αν είναι ενσωματωμένα στην html
- c. **Ο χρήστης δεν μπορεί να δει τον κώδικα php αλλά μπορεί να δει τα javascript που συμπεριλαμβάνονται**
- d. Ο χρήστης μπορεί να ενεργοποιήσει τον debugger του browser και να δει ολόκληρο τον κώδικα του αρχείου test.php

16. Για να μπορεί μία σελίδα να κρατάει τον πίνακα SESSION σε php:

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

17. Δίνεται ο παρακάτω κώδικας:

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

- a. Η μεταβλητή \$a δεν είναι «ορατή» στην συνάρτηση x() άρα δεν υπάρχει τρόπος πρόσβασης στην \$a μέσα από την x()
- b. **Η μεταβλητή \$a δεν είναι «ορατή» στην συνάρτηση x() μόνο αν την δηλώσουμε global \$a μέσα στην x() τότε αποκτάται πρόσβαση στην \$a**
- c. Η μεταβλητή \$a είναι «ορατή» στην συνάρτηση x()

18. 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;

- a. SESSION
- b. GET
- c. COOKIE
- d. **POST**

19. Ένα URL εκτός από τα υποχρεωτικά τμήματα μπορεί να περιέχει και τα παρακάτω τμήματα:

- a. **Port number**
- b. **Ορίσματα/δεδομένα μεθόδου GET**
- c. Έκδοση browser
- d. **Διαδρομή φακέλου**
- e. Ορίσματα/δεδομένα μεθόδου POST

20. 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')); ?>
```

- a. The cat sat on the donkey
- b. A dog stood on the mat
- c. PHP Fatal error
- d. **The dog stood on the donkey**

21. Ο κώδικας php τοποθετείται σε ένα ζευγάρι delimiter. Ποιο είναι το σωστό;

- a. <&>...</&>
- b. <?php>...</?>
- c. **<?php...?>**
- d. <script>...</script>

22. Ποιο είναι το output του παρακάτω κώδικα;

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

- a. **Stale Mcaffee. Born in 1950**
- b. Mcaffee. Born in
- c. Mcaffee. Born in 1950
- d. Stale Mcaffee. Born in

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

- a. **isset()**
- b. unset()
- c. set()
- d. καμία από τις παραπάνω

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

- a. \$cars = "volvo", "vw", "toyota";
 - b. \$cars = ["volvo", "vw", "toyota"];**
 - c. \$cars = array("volvo", "vw", "toyota");**
 - d. \$cars = array["volvo", "vw", "toyota"];
25. Ποιο-α από τα παρακάτω ισχύει-ουν αναφορικά με τη γραμμή κώδικα:
while(\$row= \$result->fetch_assoc()) {...}
- a. Το row είναι ένα associative array με τόσα κελιά όσα και τα πεδία που ανακτήθηκαν από τη ΒΔ**
 - b. Η συγκεκριμένη εντολή δεν μπορεί να χρησιμοποιηθεί σε συνδυασμό με prepared i. Statement
 - c. Το while loop θα εκτελεστεί τόσες φορές όσες και οι εγγραφές που επιστράφηκαν από το ερώτημα στη ΒΔ**
 - d. Οι συναρτήσεις fetch_assoc() και fetch_array() είναι ισοδύναμες
26. Ποιο είναι το output του παρακάτω κώδικα;
<?php function invite(&\$s) {\$s= 'keep calm ';} \$x = 'pass exams ';invite(\$x);echo\$x; ?>
- a. pass exams
 - b. keep calm**
 - c. keep calm pass exams
 - d. pass exams keep calm
27. 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)); ?>
- a. Δεν θα εκτελεστεί επειδή υπάρχει συντακτικό λάθος
 - b. 21521
 - c. 11**
 - d. 1521
28. Τα ορίσματα GET σε ένα URL χωρίζονται μεταξύ τους με:
- a. &**
 - b. #
 - c. %
 - d. ?
29. Ποια από τις παρακάτω προτάσεις είναι **λάθος**;
- a. Η τεχνολογία AJAX είναι μία client-side προσέγγιση για την επικοινωνία με διάφορες άλλες server-side τεχνολογίες (PHP, JSP, ASP, κτλ.)
 - b. Η τεχνολογία AJAX επιτρέπει να ενημερώνουμε ένα μέρος της σελίδας με δεδομένα από τον εξυπηρετητή χωρίς να χρειάζεται επαναφόρτωση όλης της σελίδας
 - c. Η τεχνολογία AJAX βασίζεται στο αντικείμενο XMLHttpRequest
 - d. Η τεχνολογία AJAX απαιτεί την εκμάθηση μιας νέας ολόκληρης γλώσσας client-side προγραμματισμού**

30. Η εντολή: `xmlHttpRequest.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
31. Ποιο από τα παρακάτω είναι **αλήθεια**;
- To 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
32. Τι από τα παρακάτω ισχύει για ένα Restful Web API (Rest API);
- 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
33. 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 "/getCars"**
 - Για κάθε ενέργεια πρέπει να οριστεί ένα URL, π.χ /getCats, /deleteCar,/searchCars,/updateCars
34. 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**
35. 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**
36. 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**
37. 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);
```

- a. Η σύνταξη JSON δεν είναι σωστή. Όποτε δεν θα γίνει parse ο JSON κώδικας
 - b. Θα γράψει στην ιστοσελίδα τη λέξη "Jones"**
 - c. Για να γίνει parse το JSON πρέπει υποχρεωτικά να χρησιμοποιηθεί η συνάρτηση eval
 - d. Η JSON σύνταξη ορίζει ένα αντικείμενο. Το key είναι το "employees" ενώ το value είναι ένας πίνακας αντικειμένων**
38. Ο παρακάτω κώδικας (jquery)
- ```
var elems = $("#george p") ;
```
- a. Θα επιστρέψει όλα τα στοιχεία paragraphs που βρίσκονται κάτω από ένα στοιχείο της κλάσης george
  - b. Θα επιστρέψει όλα τα στοιχεία paragraphs και όλα τα στοιχεία με το id "george"
  - c. Θα επιστρέψει όλα τα paragraphs και όλα τα στοιχεία της κλάσης george
  - d. Θα επιστρέψει όλα τα στοιχεία paragraphs που βρίσκονται κάτω από ένα στοιχείο με το id"george"**
39. Ποιο από τα παρακάτω ισχύει; (jquery)
- a. document.getElementById("myid")==\$("#myid")[0];**
  - b. document.getElementById("myid")==("#myid");
  - c. document.querySelectorAll("p")[0]==\$("p")[0];**
  - d. document.getElementById("myid")==\$("#myid").get(0)**
  - e. document.getElementById("myid") == \$("#myid");
  - f. document.querySelectorAll("p") == \$("p");
  - g. document.getElementById("myid")=\$("#myid")[0];
  - h. document.getElementById("myid")[0]==("#myid")[0];
40. Η JQuery είναι :
- a. Μια Βιβλιοθήκη για την γλώσσα Javascript**
  - b. Μια γλώσσα προγραμματισμού ειδικά σχεδιασμένη για τον Apache Web Server
  - c. Μια γλώσσα προγραμματισμού
  - d. Μια κλάση Java με την οποία πραγματοποιούμε queries σε μια βάση δεδομένων
41. Για το πρωτόκολλο HTTP το POST είναι:
- a. Μέθοδος Request**
  - b. Πεδίο μέσα στο κυρίως σώμα του Request
  - c. Πεδίο μέσα στις κεφαλίδες του Request
42. Ένα HTTP Response (απάντηση) περιέχει έναν κωδικό (Response status code)
- a. Υπάρχουν προκαθορισμένοι κωδικοί και η εξήγηση τους και ο προγραμματιστής μπορεί να ορίσει και τους δικούς του**
  - b. Ο κωδικός αυτός μπορεί να είναι οποιοδήποτε string
  - c. κωδικός αυτός μπορεί να είναι οποιοδήποτε νούμερο θέλουμε
43. Ο παρακάτω κώδικας είναι:
- ```
$(function() { // do stuff with the DOM } );
```
- a. Ο κώδικας μέσα στην συνάρτηση θα εκτελεστεί μετά το φόρτωμα της**

σελίδας

- b. Είναι η κλήση της συνάρτησης με όνομα \$ και όρισμα μια anonymous function
 - c. Λάθος. Δεν υπάρχει κάτι τέτοιο.
 - d. Μια anonymous function.
44. Για να χρησιμοποιήσουμε το εξωτερικό DTD έχουμε τη σύνταξη:
- a. `<? Xml version = "A.0" standalone = "no"?><! ΣΥΣΤΗΜΑ ΕΓΓΡΑΦΟΥ ΕΓΓΡΑΦΟΥ "order.dtd";>`
 - b. `<? Xml έκδοση = "A.0" αυτόνομο = "ναι"?><! ΣΥΣΤΗΜΑ ΕΓΓΡΑΦΟΥ ΕΓΓΡΑΦΟΥ "order.dtd";>`
45. Ένα Web URL μπορούμε να το κατεβάσουμε:
- a. **Υλοποιώντας ένα πρόγραμμα που μπορεί να επικοινωνεί με το πρωτόκολλο HTTP**
 - b. **Με πολλές εντολές από την γραμμή εντολών όπως wget, curl, GET κλπ**
 - c. **Με τη βοήθεια ενός browser**
 - d. Μονο με έναν Browser και κανέναν άλλον τρόπο
46. Ποιο-α τμήματα περιεχει ένα HTTP Request:
- a. **Μια γραμμή που περιέχει method+URL+protocol version**
 - b. **Το κυρίως σώμα του Request**
 - c. **Τις κεφαλίδες του Request**
 - d. Το SOAP Envelope
47. Όταν σε ένα URL οι εξής χαρακτήρες είναι ειδικοί:
- a. **Το ``+'' σημαίνει Space**
 - b. Το ``+'' σημαίνει συνένωση Strings
 - c. **Το ``&'' διαχωρίζει τις παραμέτρους μεταξύ τους**
 - d. **Το ``%'' ακολουθείται από έναν 2ψηφιο δεκαεξαδικό και αντιστοιχεί στον UTF-8 χαρακτήρα με τον συγκεκριμένο κωδικό**
 - e. Το ``?'' διαχωρίζει τις παραμέτρους μεταξύ τους
48. Ένα πρόγραμμα php μπορεί να εκτελεστεί?
- a. **Μέσω διακομιστή web που έχει ενεργοποιημένη την κατάλληλη ενότητα (module)**
 - b. **Από τη γραμμή εντολών εάν είναι εγκατεστημένο το πρόγραμμα-πελάτης PHP**
 - c. Απευθείας από μια εφαρμογή επιτραπέζιου υπολογιστή
 - d. Χρήση εικονικής μηχανής Java
49. Το Datagrid Web Control υποστηρίζει ?
- a. **Paging**
 - b. **Data Binding**
 - c. Speech Recognition
 - d. 3D Rendering
50. Συνηθισμένοι λόγοι που κάποιος θα επιλέξει την χρήση ενός CMS σε σχέση με ένα custom-made web application είναι;
- a. **Ύπαρξη διαφορετικών templates/themes**

- b. Χρήσης επεκτάσεων (components widgets, plugins)
- c. Search-engine-optimisation-friendly markup
- d. Μεγαλύτερη ταχύτητα στην εκτέλεση της εφαρμογής

51. Ποιο-α από τις παρακάτω εφαρμογές είναι εξυπηρετητές ιστού:

- a. MOSAIC
- b. IIS
- c. Nginx
- d. Nexus

52. Όταν μέσα από ένα PHP αρχείο παραγουμε JSON μορφή:

- a. Δεν πρέπει με κανέναν τρόπο να ορίσουμε content type
- b. Πρέπει να θέσουμε header('Content-type: application/json').
- c. 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.
- d. Πρέπει οπωσδήποτε να ορίσουμε header('Content-type:xxx/yyy') ακόμη και αν το xxx/yyy δεν αντιστοιχεί στο json

53. Ποιο/α από τα παρακάτω είναι σωστό/ά;

- a. `$("#button").click(function(){$("#p").hide();});`
- b. `$("#button").click(function hide(){});`
- c. `$("#button").click(function hide());`
- d. `$("#button").click(this.hide());`

54. Η HTML, όπως αρχικά δημιουργήθηκε, βασίζεται σε ποια γλώσσα;

- a. SGML
- b. DTD
- c. XHTML
- d. XML

55. Στο παρακάτω php παράδειγμα

```
$a = 'hello' ; $hello = 'a';  
print "{$a}";
```

- a. Θα τυπωθεί "\$hello"
- b. Θα τυπωθεί "\$a"
- c. Θα τυπωθεί "a"
- d. Θα τυπωθεί "hello"

56. Με βάση το HTTP , πληροφορίες για την γλώσσα (Ελληνικά, Αγγλικά, κτλ) μπορεί να υπάρχουν:

- a. Στο header μιας HTTP Response (απάντησης)
- b. Στο body μιας HTTP Response (απάντησης)
- c. Στο header μιας HTTP Request (αίτησης)
- d. Στο body μιας HTTP Request (αίτησης)

57. Ποιό/ά από τα παρακάτω είναι σωστό/ά; (jquery)

- a. `$('#li').html('This is an element');`
- b. `$('#li').html = 'This is an element';`

- c. `$(‘li’).innerHTML=‘This is an element’;`
 - d. `$(‘li’).text=‘This is an element’;`
58. Εάν κάνουμε pull μεταξύ 2 repositories ΔΥΟ φορές συνεχόμενα:
- a. Τα 2 repositories σίγουρα ΔΕΝ θα συγχρονιστούν πλήρως ανεξάρτητα από τις άλλες αλλαγές που υπήρχαν
 - b. Τα 2 repositories θα συγχρονιστούν πλήρως
 - c. Δεν χρειάζεται 2ο pull. Ακόμη και με το 1ο τα 2 repositories θα συγχρονιστούν πλήρως
 - d. **Τα 2 repositories μπορεί να μην συγχρονιστούν πλήρως ανάλογα με τις αλλαγές που υπήρχαν**
59. Σε ένα Version Control System διαγράφονται αυτόματα παλιές εκδόσεις κώδικά όταν:
- a. Παρέλθει χρόνος 2 μηνών
 - b. **Ποτέ**
 - c. Αν κάποιο αρχείο δεν χρησιμοποιείται
 - d. Παρέλθει ο χρόνος που έχει ορίσει ο Administrator για το κάθε project
60. Ποιο/α από τα παρακάτω είναι σωστό/ά; (jquery)
- a. `var t = <p>Test</p>;
$(“body”).append(t);`
 - b. **`$(“body”).append("<p>Test</p>");`**
 - c. `$(“body”).append(<p>Test</p>);`
 - d. **`var t = "<p>Test</p>;
$(“body”).append(t);`**
 - e. **`var t = $('<p></p>') .text ('Test');
$(“body”) .append(t);`**
61. Ποιές από τις παρακάτω jquery μεθόδους έχουν σχέση με Ajax;
- a. **`$('#maincontent').load('shopinfo.php');`**
 - b. `$('#maincontent').html('shopinfo.php');`
 - c. `$.json_parse('#maincontent');`
 - d. **`$('#maincontent').get('shopinfo.php'){success: function() { // do something}};`**
 - e. **`$.ajax('shopinfo.php', {success: function() { // do something}});`**
62. Με ένα σύστημα VCS (Version Control System):
- a. Ελέγχεται εάν σε κάποιον υπολογιστή είναι εγκατεστημένη η σωστή έκδοση του λογισμικού που θέλουμε
 - b. Ο χρήστης μπορεί να ελέγχει τις εκδόσεις των προγραμμάτων που κατεβάζει στον υπολογιστή του.
 - c. **Μπορούν να συνεργαστούν πολλοί προγραμματιστές ώστε να γράφουν κώδικά στα ίδια αρχεία ενός project.**
 - d. **Ο προγραμματιστής διαχειρίζεται τις εκδόσεις του κώδικα του**
63. 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')); ?>
```

- a. The cat sat on the donkey
  - b. A dog stood on the mat**
  - c. PHP Fatal error
  - d. The dog stood on the donkey
64. Ποια από τα παρακάτω ισχύει ;
- a. JSON είναι μια μορφή για ανταλλαγή δεδομένων**
  - b. Η JSON είναι μορφή της XML
  - c. Το JSON φαρμάτ μπορεί να γίνει parse από τον js controller**
  - d. Με json μπορούμε να μεταφέρουμε ένα αντικείμενο από την php του server σε javascript του client**
  - e. Το JSON φαρμάτ είναι συμβατό με js αλλά όχι με php
65. 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.
- a. Εξαρτάται από τον server που χρησιμοποιούμε
  - b. Λάθος
  - c. Σωστό**

Επεξήγηση:

**Για να εκτελεστεί ένα πρόγραμμα php:**

○ είτε μέσω του web server (και του module php)

Προϋποθέσεις:

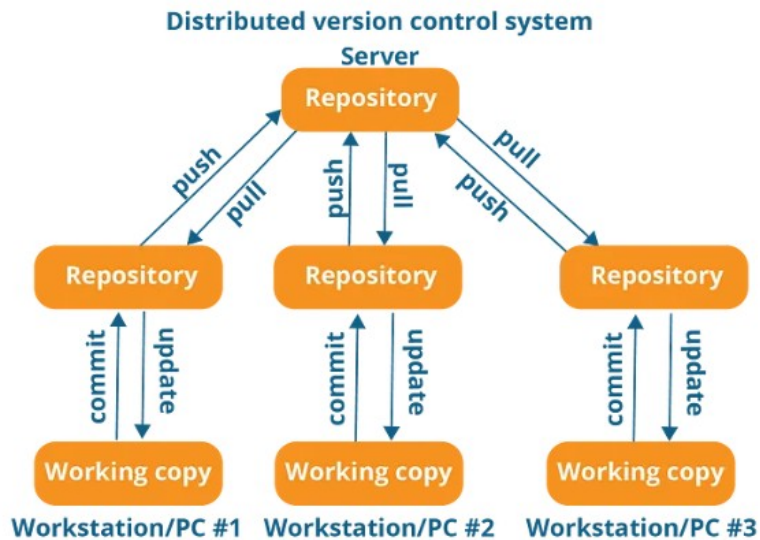
- ✖ Υπάρχει εγκατεστημένος web server (apache ή IIS ή...)
- ✖ Υπάρχει εγκατεστημένη η php
- ✖ Υπάρχει εγκατεστημένο το κατάλληλο module για τον web browser (ή έχει ρυθμιστεί η php να τρέχει από cgi – php-cli)
- ✖ Στον web server έχουν γίνει οι κατάλληλες ρυθμίσεις ώστε όταν του ζητείται ένα αρχείο .php να το «δίνει» πρώτα για εκτέλεση στο php\_module (cgi, php, php-fpm)
- ✖ Τα αρχεία php (όπως και όλα τα αρχεία html, css, js) βρίσκονται σε φάκελο στον οποίο έχει ρυθμιστεί να έχει πρόσβαση ο web server.

○ Είτε από command line (php-cli) σαν desktop εφαρμογή.

66. Έστω ότι χρησιμοποιούμε κάποιο Distributed Version Control System . Όταν κάνουμε update:
- a. Αποθηκεύονται αλλαγές στο τοπικό Repository από το Working Copy
  - b. Μεταφέρονται οι αλλαγές από το τοπικό Repository στο Working Copy**
  - c. Μεταφέρονται οι αλλαγές από το τοπικό Repository στο απομακρυσμένο
  - d. Μεταφέρονται οι αλλαγές από το απομακρυσμένο Repository στο τοπικό Repository

Επεξήγηση:

Every contributor has a local copy or “clone” of the main repository i.e. Everyone maintains a local repository on its own, which is actually the copy or clone of the central repository on their hard drive. They can commit and update their local repository without any interference.



67. 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.
68. Ποιο 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
69. Έστω ο JSON κώδικας:
- ```
{“name”:“Jack B. Nimble”, “at large”:true,“grade”:“A”,
“format”:{“type”:“rect”,“width”:1920}}
```
- Τι από τα παρακάτω ισχύει:
- Το “format”:{“type”:“rect”,“width”:1920} είναι αντικείμενο**
 - Το “A” είναι key
 - Υπάρχει λάθος στη σύνταξη
 - Το “at large” είναι value
70. Το GIT είναι:
- Version Control System 1ης γενιάς
 - Distributed Version Control System**
 - Version Control System (Centralized)
71. Ποιό/ά από τα παρακάτω είναι σωστό;
- \$("#p").click(do_it);
function do_it(event) { alert('click'); }**

- b. `$("#p").click(function(){ alert('click'); });`
 - c. `$("#p").click(do_it);`
`function do_it() { alert('click'); }`
 - d. `$("#p").click(do_it) { alert('click'); });`
 - e. `$("#p").click(do_it());`
72. Τα Web Sockets σε σχέση με τα HTTP Requests:
- a. **Επιτρέπουν να παραμένει η σύνδεση ενεργή συνεχώς, ενώ το HTTP "κλείνει" την σύνδεση αυτόματα.**
 - b. Δεν υποστηρίζονται από τους περισσότερους browsers ενώ το HTTP υποστηρίζεται από όλους.
 - c. **Επιτρέπουν πλήρη αμφίδρομη επικοινωνία (μετά το handshake) ενώ το HTTP μόνο αιτήσεις από τον client στον server**
73. Πότε **μπορεί** να δημιουργηθεί διένεξη (conflict) σε ένα VCS;
- a. Όταν γίνεται version merging.
 - b. **Τη στιγμή που δυο προγραμματιστές θα προσπαθήσουν να διορθώσουν το ίδιο αρχείο συγχρόνως.**
 - c. Όταν δυο προγραμματιστές μαλώνουν.

Επεξήγηση:

Η πραγματική διένεξη δεν προκύπτει απλώς επειδή δύο προγραμματιστές επεξεργάζονται το ίδιο αρχείο ταυτόχρονα, αλλά όταν οι αλλαγές τους συγχωνεύονται ξανά στο κοινόχρηστο αποθετήριο και αυτές οι αλλαγές έρχονται σε διένεξη μεταξύ τους. Τα σύγχρονα εργαλεία VCS επιτρέπουν σε πολλούς προγραμματιστές να εργάζονται στο ίδιο αρχείο, αλλά οι διενέξεις θα πρέπει να επιλυθούν κατά τη συγχώνευση αυτών των αλλαγών.

74. Πως ορίζουμε ένα cookie στην PHP,
- a. `defincookie()`
 - b. **`setcookie()`**
 - c. `createcookie()`
 - d. `makecookie()`
75. Έστω ότι κατασκευάζεται μια εφαρμογή όπου ο χρήστης κάνει Login. Θέλετε να έχετε πρόσβαση στα στοιχεία login του χρήστη σε όλες τις σελίδες του διαδικτυακού τόπου. Ποιο μέθοδο θα χρησιμοποιούσατε;
- a. **SESSION**
 - b. POST
 - c. COOKIE
 - d. GET

Επεξήγηση:

Τα SESSIONS σε PHP (ή άλλες γλώσσες που υποστηρίζουν τη διαχείριση περιόδων σύνδεσης) σάς επιτρέπουν να αποθηκεύετε πληροφορίες χρήστη στο διακομιστή κατά τη διάρκεια της επίσκεψης του χρήστη. Όταν ένας χρήστης συνδέεται, μπορείτε να αποθηκεύσετε τις πληροφορίες σύνδεσής του σε μια μεταβλητή περιόδου λειτουργίας, στην οποία μπορείτε στη συνέχεια να έχετε πρόσβαση από οποιαδήποτε σελίδα στον

ιστότοπό σας μέχρι να αποσυνδεθεί ο χρήστης ή να λήξει η περίοδος λειτουργίας. Αυτή η μέθοδος είναι ασφαλής επειδή τα δεδομένα αποθηκεύονται στον διακομιστή και μόνο ένα αναγνωριστικό περιόδου σύνδεσης μεταβιβάζεται μεταξύ του διακομιστή και του πελάτη, συνήθως σε ένα cookie.

Η χρήση POST ή GET θα λειτουργούσε μόνο για μεμονωμένα αιτήματα και απαντήσεις. Δεν είναι κατάλληλα για τη διατήρηση πληροφοριών κατάστασης ή χρήστη σε πολλές σελίδες. Τα COOKIES μπορούν να χρησιμοποιηθούν για τη διατήρηση της κατάστασης στην πλευρά του πελάτη σε πολλές σελίδες, αλλά είναι λιγότερο ασφαλή για την αποθήκευση ευαίσθητων πληροφοριών, όπως στοιχεία σύνδεσης, επειδή αποθηκεύονται στον υπολογιστή του πελάτη και μπορούν να υποκλαπούν ή να παραβιαστούν.

76. Στο μοντέλο MVC:

- a. Όταν γίνεται μια ενέργεια από τον χρήστη, το Model ενημερώνει το View.
- b. Το Model περιέχει την δομή των δεδομένων και τις απαραίτητες μεθόδους ανάκτησης αυτών.**
- c. Ο Controller περιέχει τα δεδομένα.

Επεξήγηση:

Σε ορισμένες υλοποιήσεις, το View μπορεί να εγγραφεί σε αλλαγές στο Model και να ενημερωθεί αναλόγως, αλλά αυτός ο μηχανισμός ενημέρωσης ενορχηστρώνεται μέσω του Controller και όχι απευθείας από το Model.

77. Ποια/-ες από τις παρακάτω τεχνολογίες/γλώσσες μπορούν να εκτελεστούν στον server;

- a. asp
- b. flash
- c. java applet
- d. Perl**

78. Ο πίνακας \$_SESSION χρησιμοποιείται για να αποθηκεύσει πληροφορίες που αφορούν στοιχεία πιστοποίησης (authentication), προτιμήσεις χρήστη, καλάθια αγορών και άλλα στοιχεία συνόδου (Session) με τον server. Τι από τα παρακάτω είναι σωστά;

- a. Τα δεδομένα του Session αποθηκεύονται στον server.
- b. Ο πίνακας Session δεν αποθηκεύεται τοπικά (στον χρήστη). Στην πλευρά του χρήστη αποθηκεύεται μόνο το SessionID.
- c. Τα cookies αποθηκεύονται στο server και χάνονται μετά από κλείσιμο του browser.
- d. Ο πίνακας session αποθηκεύεται στον web server και διατηρείται τουλάχιστον για 2 εβδομάδες.
- e. Ο πίνακας Session αποθηκεύεται τοπικά (στον χρήστη) και χάνεται μετά από κλείσιμο του browser.

79. Έστω ότι χρησιμοποιούμε το Github Desktop (ή και κάποιον άλλον client για το github). Όταν κάνουμε commit:

- a. Αποθηκεύονται οι αλλαγές στο Repository στον server του github από το working copy.
 - b. Αποθηκεύονται οι αλλαγές στο τοπικό Repository στον server του github από το working copy.**
 - c. Μεταφέρονται οι αλλαγές από το τοπικό Repository στο Repository στον server του github.
80. Για να επικοινωνήσει ένα php script με την mysql:
- a. Πρώτα πρέπει να γίνει σύνδεση του script με τον apache. Ο apache θα στείλει τα sql queries στην mysql.
 - b. Πρώτα πρέπει να γίνει σύνδεση του script στον mysql server.**
 - c. Δεν είναι απαραίτητο να γίνει σύνδεση του script στον mysql server διότι θα γίνει αυτόματα στο localhost όταν εκτελεστεί η πρώτη εντολή mysql.
 - d. Πρέπει ο apache web server να γνωρίζει τα στοιχεία σύνδεσης στην βάση (username/password) για να μπορεί να κάνει την σύνδεση.**
81. Έστω ο παρακάτω κώδικας
- ```
$stmt = $conn->prepare("INSERT INTO Mytable VALUES (?"); $stmt->bind_param("i", $a); $a = "John"; $stmt->execute();
```
- Ποιο-α από τα παρακάτω ισχύει/-ουν;
- a. Η συνάρτηση bind\_param αντικαθιστά το περιεχόμενο της μεταβλητής δε με το string "i"
  - b. Θα προστεθεί μια εγγραφή στον πίνακα Mytable της ΒΔ
  - c. Το πεδίο x του πίνακα Mytable είναι integer**
  - d. Υπάρχει λάθος και δεν θα εκτελεσθεί το insert into στη ΒΔ**

## Laravel

### (Δικές μας Ερωτήσεις)

1. Το εργαλείο artisan της laravel: **(ερώτηση από εξεταστική)**
  - a. δεν είναι τμήμα της laravel αλλά της php.
  - b. περιέχει πολλά scripts για την διευκόλυνση κάποιων διαδικασιών.**
  - c. είναι το τμήμα με το οποίο δημιουργούνται τα γραφικά στην laravel.
  - d. είναι τμήμα της laravel το οποίο αφορά αποκλειστικά την δημιουργία του model.
2. Factories και Seeders: **(ερώτηση από εξεταστική)**
  - a. Τα Factories χρησιμοποιούνται από τους Seeders για την δημιουργία των Models.**
  - b. Οι Seeders είναι κλάσεις μέσα από τις οποίες μπορούμε να δημιουργήσουμε δεδομένα στην βάση.**
  - c. Τα Factories είναι οι Constructors των routes της laravel.
  - d. Οι Seeders είναι ο τρόπος με τον οποίον εισάγουμε δεδομένα/ορίσματα στα routes της laravel.



Επεξήγηση:

Αν εννοεί:

“Στη Laravel, τα Factories χρησιμοποιούνται για τη δημιουργία δεδομένων δοκιμών (test data) για τα μοντέλα (Models).”

Τότε το Α είναι σωστό.

3. Η δημιουργία ενός Web API με τη laravel είναι εφικτή με: **(ερώτηση από εξεταστική)**
  - a. Τον ορισμό των migrations.
  - b. Την χρήση της Vue.js.
  - c. Την κατασκευή ενός συνόλου από "Routes".**
  - d. Την χρήση της React.js.
  
4. Για τι χρησιμοποιείται κυρίως η Laravel;
  - a. Δημιουργία εφαρμογών ιστού που ακολουθούν το αρχιτεκτονικό πρότυπο μοντέλο-προβολή-ελεγκτής (MVC)**
  - b. Ανάπτυξη desktop εφαρμογών
  - c. Διαχείριση βάσεων δεδομένων απευθείας χωρίς τη συγγραφή SQL
  - d. Δημιουργία λειτουργικών συστημάτων
  
5. Ποιος δημιούργησε τη Laravel και πότε κυκλοφόρησε για πρώτη φορά;
  - a. Rasmus Lerdorf, 1995
  - b. Mark Zuckerberg, 2004
  - c. Taylor Otwell, 2011**
  - d. Dries Buytaert, 2001
  
6. Τι είναι το Eloquent ORM στο Laravel;
  - a. Ένα εργαλείο γραμμής εντολών για τη διαχείριση μεταναστεύσεων βάσεων δεδομένων
  - b. Μια υλοποίηση ενεργών εγγραφών για αποτελεσματικότερη εργασία με βάσεις δεδομένων**
  - c. Μια ενότητα για τη δημιουργία RESTful APIs
  - d. Μια μηχανή προτύπων για την απόδοση προβολών
  
7. Πώς χειρίζεται η Laravel τη δρομολόγηση;
  - a. Μέσω ενός απλού και εκφραστικού API στο αρχείο routes/web.php**
  
8. Ποια δυνατότητα παρέχει η Laravel για τον εύκολο χειρισμό πολύπλοκων συστημάτων ελέγχου ταυτότητας χρηστών;
  - a. Laravel Fortify και Jetstream**
  
9. Ποιος είναι ο σκοπός του Laravel Mix;
  - a. Για τη μεταγλώττιση και τη βελτιστοποίηση περιουσιακών στοιχείων όπως CSS και JavaScript**

# React JS

## (Δικές μας Ερωτήσεις)

1. Τι είναι η React.js;
  - a. **JavaScript Library**
2. Στην React, ποιος είναι ο σκοπός του **JSX**;
  - a. **Μας επιτρέπει να γράφουμε HTML στη React.**
3. Ποιος είναι ο σκοπός του **hook useState** στην React;
  - a. **Διαχείριση της κατάστασης σε functional components**
4. Πώς μπορείτε να μεταφέρετε δεδομένα από ένα γονικό component σε ένα child component στην React;
  - a. **Μέσω του συστήματος props**
5. Ποιος είναι ο σκοπός της βιβλιοθήκης **React Router**;
  - a. **Για την υλοποίηση δρομολόγησης σε εφαρμογές React**
6. Τι είναι το **Virtual DOM** στη React και γιατί χρησιμοποιείται;
  - a. **Το Virtual DOM είναι ένα αντικείμενο JavaScript που αναπαριστά ένα αντίγραφο του πραγματικού DOM, το οποίο χρησιμοποιείται για τη βελτιστοποίηση της απόδοσης της εφαρμογής.**

### Επεξήγηση:

Το Virtual DOM είναι μια βασική έννοια στη React που περιλαμβάνει τη δημιουργία μιας εικονικής αναπαράστασης του πραγματικού DOM στη μνήμη. Αυτή η εικονική αναπαράσταση επιτρέπει στο React να ενημερώνει και να αποδίδει αποτελεσματικά τις αλλαγές στο UI κάνοντας πρώτα αλλαγές στο Virtual DOM και στη συνέχεια εφαρμόζοντας μόνο τις απαραίτητες ενημερώσεις στο πραγματικό DOM. Αυτή η διαδικασία συμβάλλει στη βελτίωση των επιδόσεων μειώνοντας τον αριθμό των άμεσων χειρισμών στο πραγματικό DOM.

7. Ποιος διατηρεί η React;
  - a. **Facebook (Meta)**
  
8. Σε τι χρησιμοποιείται κυρίως η React στην ανάπτυξη Ιστού;
  - a. **Δημιουργία διεπαφών χρήστη και μονοσέλιδων εφαρμογών**
  
9. Τι είναι ένα συστατικό (component) στη React;
  - a. **Ένα επαναχρησιμοποιήσιμο κομμάτι διεπαφής χρήστη**
  
10. Πώς η React επιτυγχάνει επαναχρησιμοποίηση στα στοιχεία της;
  - a. **Μέσα από props και σύνθεση**
  
11. Μπορείτε να χρησιμοποιήσετε τη React για την ανάπτυξη εφαρμογών για κινητά;
  - a. **Ναι, χρησιμοποιώντας το React Native τόσο για εφαρμογές iOS όσο και για Android**
  
12. Τι σημαίνει ο όρος «δηλωτικό» (declarative) στο πλαίσιο της React;
  - a. **Περιγράφετε την κατάσταση διεπαφής χρήστη που θέλετε και η React διασφαλίζει ότι το DOM ταιριάζει με αυτήν την κατάσταση**

## WebSockets

### (Δικές μας Ερωτήσεις)

1. Τι είναι τα WebSockets;
  - a. **Μία μέθοδος αμφίδρομης επικοινωνίας σε πραγματικό χρόνο μεταξύ πελάτη και διακομιστή.**
  
2. Ποιο είναι το βασικό πλεονέκτημα που προσφέρουν τα WebSockets;
  - a. **Διατήρηση μόνιμης σύνδεσης για αμφίδρομη επικοινωνία χωρίς συνεχή επαναφόρτωση της σελίδας.**

3. Ποια θύρα (port) συνήθως χρησιμοποιείται από το WebSocket πρωτόκολλο όταν είναι ασφαλής σύνδεση (wss://);
  - a. **Η θύρα 443, όπως και το HTTPS.**
4. Ποιο από τα παρακάτω ΔΕΝ είναι στάδιο/φάση της WebSocket επικοινωνίας;
  - a. **Εκτέλεση άμεσης μεταγλώττισης (compilation) του JavaScript κώδικα στον server πριν τη σύνδεση.**
  - b. Handshake ανάμεσα σε client και server.
5. Εγκατάσταση αμφίδρομης, συνεχιζόμενης σύνδεσης μετά το handshake.
6. Αποστολή επικεφαλίδας (header) όμοιας με HTTP αίτημα κατά την αρχικοποίηση.
7. Πώς γίνεται η έναρξη μίας WebSocket σύνδεσης από τον browser;
  - a. **Χρησιμοποιώντας την κλάση WebSocket στην JavaScript, π.χ. `new WebSocket("ws://...");`.**
8. Ποιος είναι ο συνηθισμένος τρόπος να τερματιστεί σωστά μία WebSocket σύνδεση;
  - a. **Κλείνοντας τη σύνδεση μέσω JavaScript, π.χ. `socket.close()`.**
9. Τι χαρακτηρίζει το πλαίσιο επικοινωνίας μέσω WebSockets σε σχέση με τα παραδοσιακά HTTP αιτήματα;
  - a. **Τα WebSockets παρέχουν αμφίδρομη, συνεχή επικοινωνία αντί να χρειάζεται να γίνεται συνεχώς νέα αίτηση από τον client.**
10. Ποια βιβλιοθήκη ή τεχνολογία μπορούμε να χρησιμοποιήσουμε για τη διαχείριση WebSocket συνδέσεων στη JavaScript;
  - a. **Την ενσωματωμένη κλάση WebSocket που υποστηρίζει αμφίδρομη επικοινωνία.**
11. Σε μια εφαρμογή WebSocket, τι επιτυγχάνεται με το λεγόμενο “heartbeat” ή “ping/pong” μηχανισμό;
  - a. **Διατηρείται η σύνδεση ενεργή και ελέγχεται αν τόσο ο client όσο και ο server είναι ακόμη συνδεδεμένοι.**

12. Όταν λέμε ότι τα WebSockets είναι “full-duplex” τι εννοούμε;

- a. **Ο client και ο server μπορούν να στέλνουν δεδομένα ταυτόχρονα σε μια κοινή μόνιμη σύνδεση.**

## ThreeJS

### (Δικές μας Ερωτήσεις)

1. Τι είναι το Three.js;

- a. **Μία βιβλιοθήκη JavaScript για δημιουργία και εμφάνιση τρισδιάστατων γραφικών στο web.**

2. Σε ποια γλώσσα προγραμματισμού βασίζεται το Three.js;

- a. **Στη γλώσσα JavaScript, που εκτελείται στον browser.**

3. Ποιος είναι ο κύριος σκοπός του Three.js;

- a. **Να παρέχει ευκολία για τη δημιουργία τρισδιάστατων γραφικών και animations μέσα σε έναν web browser.**

4. Ποιο αντικείμενο στο Three.js χρησιμοποιείται συνήθως για να αναπαραστή την «κάμερα»;

- a. **Το αντικείμενο “PerspectiveCamera” ή άλλοι τύποι κάμερας όπως “OrthographicCamera”.**

5. Πώς προσθέτουμε τρισδιάστατα αντικείμενα (meshes) στη σκηνή (scene) στο Three.js;

- a. **Δημιουργούμε ένα geometry και ένα material, φτιάχνουμε ένα mesh και στη συνέχεια καλούμε scene.add(mesh).**

6. Σε τι χρησιμεύει ο Renderer στο Three.js (π.χ. WebGLRenderer);

- a. **Στη μετατροπή μιας τρισδιάστατης σκηνής σε εικόνα που μπορούμε να δούμε στον browser.**

7. Τι είναι τα “materials” στο Three.js;

- a. Ρυθμίσεις που καθορίζουν τον τρόπο εμφάνισης επιφανειών (π.χ. χρώμα, υφή) σε ένα τρισδιάστατο αντικείμενο.
8. Ποιος είναι ο ρόλος του “scene” στο Three.js;
- a. Παρέχει ένα δοχείο στο οποίο τοποθετούμε κάμερα και αντικείμενα (meshes, φώτα κ.λπ.).
9. Με ποιον τρόπο μπορούμε να προσθέσουμε φωτισμό (lights) στο Three.js;
- a. Προσθέτοντας αντικείμενα φωτισμού (π.χ. AmbientLight, DirectionalLight) στο scene.
10. Ποια από τις παρακάτω περιπτώσεις αναφέρεται σε πραγματική χρήση του Three.js;
- a. Ανάπτυξη τρισδιάστατων οπτικοποιήσεων και απλών web παιχνιδιών.

## Livewire

### (Δικές μας Ερωτήσεις)

1. Τι είναι το Livewire στη Laravel;
- a. Μια βιβλιοθήκη που επιτρέπει την κατασκευή δυναμικών component-based διεπαφών χωρίς σημαντική χρήση JavaScript.
2. Ποιος είναι ο κύριος στόχος του Livewire;
- a. Να βοηθά στη δημιουργία διαδραστικών δυναμικών στοιχείων (components) σε Laravel, μειώνοντας την ανάγκη χρήσης frameworks JavaScript.
3. Πώς ενσωματώνονται τα Livewire components σε μια Blade σελίδα;
- a. Με δήλωση @livewire('component-name') μέσα στο Blade template.
4. Ποιο είναι το κύριο πλεονέκτημα της χρήσης του Livewire;
- a. Επιτρέπει τη δημιουργία διαδραστικών σελίδων χωρίς να γράφουμε σχεδόν καθόλου JavaScript κώδικα.

5. Τι χαρακτηρίζει ένα Livewire component;
  - a. **Αποτελείται από μια κλάση PHP (με λογική) και ένα Blade αρχείο (με τη διεπαφή).**
6. Πώς γίνεται η επικοινωνία μεταξύ της διεπαφής και του κώδικα στα Livewire components;
  - a. **Με αυτόματη αποστολή AJAX-like αιτημάτων από τη Livewire βιβλιοθήκη όταν αλλάζουν δεδομένα στο component.**
7. Ποια είναι η βασική οδηγία στο Blade που επιτρέπει την αυτόματη φόρτωση των Livewire scripts;
  - a. **@livewireStyles και @livewireScripts τα οποία τοποθετούμε συνήθως στο layout Blade αρχείο.**
8. Με ποιο τρόπο μπορούμε να δημιουργήσουμε ένα νέο Livewire component;
  - a. **Με την artisan command: php artisan make:livewire TestComponent.**
9. Ποιο από τα παρακάτω είναι παράδειγμα binding μεταβλητής σε Livewire component;
  - a. **wire:model="search" που δίνει μία μεταβλητή PHP στο component με ένα input πεδίο.**
10. Τι πρέπει να προσέχουμε κατά την εγκατάσταση και ενημέρωση του Livewire;
  - a. **Ότι έχουμε τη σωστή έκδοση Laravel (συνήθως >= 7.x) και ενημερωμένο composer.json πριν κάνουμε composer require livewire/livewire.**

## TailwindCSS

### (Δικές μας Ερωτήσεις)

1. Τι είναι το Tailwind CSS;
  - a. **Ένα utility-first CSS framework που παρέχει έτοιμες κλάσεις για γρήγορη δημιουργία custom σχεδιασμού.**
2. Ποιο είναι το χαρακτηριστικό γνώρισμα του Tailwind CSS σε σύγκριση με άλλα CSS frameworks;

- a. Χρησιμοποιεί utility κλάσεις, επιτρέποντας σου να χτίζεις μοναδικά σχέδια χωρίς προκαθορισμένα components.
3. Πώς προσθέτουμε το Tailwind CSS σε ένα project;
  - a. Κάνουμε εγκατάσταση (π.χ. μέσω npm) και ρυθμίζουμε το αρχείο διαμόρφωσης, ή χρησιμοποιούμε CDN.
4. Τι κάνει βασικά ένα utility-first CSS framework όπως το Tailwind;
  - a. Παρέχει ένα πλήθος από μικρές, έτοιμες κλάσεις (π.χ. p-4, mt-2) για να ορίζεις layout, χρώματα, αποστάσεις κ.λπ.
5. Σε τι χρησιμεύει το αρχείο tailwind.config.js;
  - a. Επιτρέπει την προσαρμογή (customization) των classes, χρωμάτων, breakpoints και άλλων ρυθμίσεων του Tailwind CSS.
6. Ποιο από τα παρακάτω είναι ένα παράδειγμα κλάσης του Tailwind;
  - a. **bg-blue-500**
7. Ποια είναι η βασική ιδέα του Responsive Design στο Tailwind CSS;
  - a. Προσφέρει κλάσεις με prefix όπως sm:, md:, lg: που επιτρέπουν διαφορετικά styles σε διάφορα breakpoints.
8. Πώς γίνεται η διαδικασία παραγωγής (build) του τελικού αρχείου CSS με Tailwind;
  - a. Χρησιμοποιούμε εργαλεία όπως PostCSS ή webpack για να συμπίεσουμε και να παραμετροποιήσουμε το CSS.
9. Ποια η διαφορά μεταξύ “utility classes” και “component classes” στο Tailwind CSS;
  - a. Οι “utility classes” είναι απλές, προκαθορισμένες ιδιότητες (π.χ. margin, padding), ενώ οι “component classes” είναι πιο σύνθετες (π.χ. classes για κουμπιά, φόρμες) που ορίζονται από τον χρήστη ή το Tailwind UI.
10. Τι κάνει η εντολή npx tailwindcss init;
  - a. Δημιουργεί ένα αρχείο tailwind.config.js για να ρυθμίσεις το Tailwind.



# Alpine.js

## (Δικές μας Ερωτήσεις)

1. Τι είναι το Alpine.js;
  - a. Ένα ελαφρύ JavaScript framework για προσθήκη reactivity και διαδραστικότητας στο HTML.
2. Πώς ορίζουμε δεδομένα (reactive data) σε ένα Alpine.js component;
  - a. Με τη χρήση του directive `x-data="{...}"`, ορίζοντας τις μεταβλητές μέσα στα άγκιστρα.
3. Ποιο directive του Alpine.js χρησιμοποιείται συνήθως για δέσμευση (binding) τιμών σε attributes;
  - a. Το directive `x-bind`: επιτρέπει να δέσουμε εκφράσεις σε attributes στοιχείων.
4. Ποια είναι η γενική φιλοσοφία χρήσης του Alpine.js;
  - a. Να παρέχει μια ελαφριά εναλλακτική για βασική reactivity και interactions, χωρίς να γράφουμε πολλή JavaScript.
5. Ποια η διαφορά μεταξύ του Alpine.js και του jQuery σε βασικό επίπεδο;
  - a. Το Alpine.js επικεντρώνεται στην αντιδραστικότητα και την απλή διαχείριση κατάστασης (state), ενώ το jQuery επικεντρώνεται κυρίως σε DOM επιλεκτές και τροποποίηση.
6. Ποιο από τα παρακάτω είναι σωστό για την εγκατάσταση/φόρτωση του Alpine.js;
  - a. Μπορεί να προστεθεί μέσω CDN `<script>` tag ή να εγκατασταθεί με npm.
7. Τι είναι το `x-on:click` στο Alpine.js;
  - a. Ένα directive που συσχετίζει ένα click event με μια έκφραση ή συνάρτηση.
8. Ποιο είναι το όφελος της προσέγγισης του Alpine.js στα μικρά, δυναμικά τμήματα ενός site;

- a. Μπορούμε να προσθέσουμε απλή διαδραστικότητα σε συγκεκριμένα sections, χωρίς βαρύ framework.
9. Πώς γίνεται η διασύνδεση μεταξύ πολλών Alpine.js components σε μία σελίδα;
- a. Μπορούμε να κάνουμε “emit” γεγονότα από ένα component και να “listen” σε άλλο, χρησιμοποιώντας custom events ή απλή προσέγγιση στο DOM.
10. Ποιος είναι ο συνήθης τρόπος αποσφαλμάτωσης (debug) ενός Alpine.js component;
- a. Μέσω της κονσόλας του browser (Developer Tools) και παρακολουθώντας τα updates που γίνονται στα στοιχεία.

## FastAPI

### (Δικές μας Ερωτήσεις)

1. Τι είναι το FastAPI;
  - a. Ένα σύγχρονο framework σε Python για γρήγορη δημιουργία RESTful APIs.
2. Ποιο από τα παρακάτω χαρακτηρίζει το FastAPI;
  - a. Υποστηρίζει τύπους δεδομένων (type hints) της Python για αυτόματη τεκμηρίωση και validation των endpoints.
3. Πώς εκτελούμε μια FastAPI εφαρμογή αφού έχουμε ορίσει το αρχείο main.py;
  - a. Συνήθως εκτελούμε την εντολή `unicorn main:app --reload` ώστε να τρέξει ο server της FastAPI.
4. Τι είναι η αυτόματη τεκμηρίωση (documentation) που προσφέρει το FastAPI;
  - a. Αυτόματη δημιουργία διαδραστικού περιβάλλοντος τεκμηρίωσης (Swagger ή Redoc) χάρη στα type hints και τα decorators.
5. Σε τι εξυπηρετούν τα “path parameters” στο FastAPI;
  - a. Επιτρέπουν να ορίσουμε δυναμικά τμήματα σε ένα endpoint (π.χ. `/users/{user_id}`), προσπελάζοντας την τιμή στην Python συνάρτηση.
6. Πώς μπορούμε να ορίσουμε ένα βασικό GET endpoint στο FastAPI;

- a. Ορίζοντας μια συνάρτηση στο Python αρχείο και διακοσμώντας την με `@app.get("/path")`.**
- 7. Ποιο είναι το κύριο πλεονέκτημα του FastAPI σε σχέση με πολλά άλλα Python frameworks;
  - a. Υποστηρίζει `async/await`, αξιοποιώντας την ασύγχρονη εκτέλεση για υψηλές επιδόσεις.**
- 8. Τι σημαίνει η χρήση του `@app.post("/users")` decorator;
  - a. Ρυθμίζει ένα endpoint στο FastAPI που ανταποκρίνεται σε αιτήματα POST στο μονοπάτι `/users`.**
- 9. Με ποιον τρόπο διαχειρίζεται το FastAPI την επεξεργασία του body ενός αιτήματος;
  - a. Εφόσον δηλώσουμε `type hints` στις Python συναρτήσεις, γίνεται αυτόματα `parse` και αντιστοίχιση παραμέτρων.**
- 10. Ποιο χαρακτηριστικό του FastAPI βοηθάει στην καλύτερη ανάπτυξη σε περιβάλλοντα τύπου `Microservices`;
  - a. Η ευκολία στον ορισμό και στην ασύγχρονη επεξεργασία endpoints, διευκολύνοντας τη σύνθεση μικρο-υπηρεσιών.**

---

Σημείωση από τον συγγραφέα του MegaPack:

Σας άρεσε το παραπάνω υλικό; Υποστηρίξτε τη δουλειά μου αγοράζοντας μου έναν καφέ από το παρακάτω link, και βοηθήστε με να δημιουργήσω ακόμα καλύτερο περιεχόμενο!

[Buy me a Coffee](#)