

Theoretical questions

Explain shortly each question and give an example for each in PHP

- 1) What is a web server?
- 2) What is the difference in strings written in single and double quotes?
- 3) What are some main built-in functions to sort arrays?
- 4) What is the difference between require and include?
- 5) What are PHP superglobals? Explain shortly at least 9 of them.
- 6) What are some differences between \$_GET and \$_POST?
- 7) What is htmlentities() function?
- 8) How do you open a file? List some file open modes and explain each.
- 9) How can you read data from a file and write data to a file?
- 10) What are cookies?
- 11) How do you store an array in a cookie? How do you read this array back from cookie?
- 12) What are sessions?
- 13) What is session propagation?
- 14) What are three main categories of errors in PHP? Explain each.
- 15) How can you suppress errors?
- 16) What are persistent connections?
- 17) What is filtering?
- 18) What is escaping?
- 19) What is session fixation?
- 20) What is PDO?
- 21) What is a class constructor?
- 22) What is a class destructor?
- 23) What are magic methods?
- 24) What is a namespace?
- 25) What is a trait?

Apache Virtual Hosts

- 1) Write Apache configuration codes to display 2 different websites on ports 80 and 8080. Each web site has different paths on the server.
- 2) Write Apache configuration codes to display 2 different websites with names www.siteone.com and www.anothersite.com from the same server. Each website has different paths on the server
- 3) Suppose we have a web site outside of xamp\htdocs and we want to access it from our web server. Configure this directory with any alias that you want.
For example: we have a site in the httpdocs called www.test.com and another folder in /home/homework. We want to access the homework in this way:
www.test.com/homework

Look at “VirtualHosts” and “Directory and Aliases” PDFs on LMS to solve all three of them.

.htaccess

Suppose we are working on localhost. Write PHP code and .htaccess configuration file to display the following links and outputs in the browser.

- 1) When we browse on **localhost/foo/2/3** display on the html page: The first number is 2 and the second number is 3

Solution

index.php

```
<form action="<?php echo $_SERVER['PHP_SELF'] ?>" method="get">

    Number 1: <input type="number" name="number1"><br>

    Number 2: <input type="number" name="number2"><br>

    <button type="submit" name="submit" value="submit">Submit</button>

</form>

<?php

    if (isset($_GET['submit'])) {

        $nr1 = $_GET['number1'];

        $nr2 = $_GET['number2'];

        echo "The first number is $nr1 and the second number is $nr2";

    }

?>
```

.htaccess file

RewriteEngine on

RewriteRule ^foo/([0-9]+)/([0-9]+) index.php?number1=\$1&number2=\$2&submit=submit

- 2) Display the same message as in 1, even if we browse to **localhost/2/3.bar**

PHP

- 1) Write PHP script to update values from the database. The database Epoka has a table named User with two fields: email and password.
 - a) Create a form to update the current logged in user. The id of the current logged in user you can find it from \$_SESSION['user_id']. When you first open the form, show the current email and password (do not leave the input fields empty). To connect to the database use credentials: Server: 127.0.0.1, database: Epoka, Username: root, Password: "".
 - b) When submitting the data, validate the email with regular expression. Email

format should be: first letter of the first name followed by last name followed by 2 numbers followed by "@epoka.edu.al" or "@epoka.al" or "@cen.epoka.edu.al".

Password should be at least 6 characters long and contain a special character.

c) In a validation.txt file, write the user_id of any unsuccessful attempt to update, separated by commas. For example: If I try to update my email wrongly, write in the validation.txt file my user id (get it from Session).