college **CDI**

Server-side technologies (TCS)

Class 8

Uploading files using PHP

PHP allows you to upload files using form. For instance, a candidate may want to use your contact form and joint an PDF version of a resume.

However, the server is always at risks when you allow someone to upload something. Therefore, make sure to always apply validation.

```php
<!DOCTYPE html>
<html lang="en">
<head>
<title>Uploading a file</title>
</head>
<body>

<form action="#" method="post" enctype="multipart/form-data">
        Select your image :
        <input type="file" name="userImage">
        <input type="submit" value="Upload Image" name="submit">
</form>

<?php
$target_dir = "uploads/";                                           // directory on server
$target_file = $target_dir . basename($_FILES["userImage"]["name"]);
$uploadOk = 1;
$imageFileType = strtolower(pathinfo($target_file,PATHINFO_EXTENSION));

if(isset($_POST["submit"])) {                                       // checks if the file is really an image
                $check = getimagesize($_FILES["userImage"]["tmp_name"]);
                if($check !== false) {
                        echo "This file is an image - " . $check["mime"] . ".";
                        $uploadOk = 1;
                } else {
                        echo "This file is not an image.";
                        $uploadOk = 0;
                }
}
?>
</body>
</html>
```

Other possible validations

Checks if the file already exists in the uploads folder :

```
if (file_exists($target_file)) {
        echo "Sorry, file already exists.";
        $uploadOk = 0;
}
```

Checks file size :

```
if ($_FILES["userImage"]["size"] > 500000) {
        echo "Sorry, your file is too large.";
        $uploadOk = 0;
}
```

Allowing only specific formats :

```
if($imageFileType != "jpg" && $imageFileType != "png" && $imageFileType != "jpeg" ) {
        echo "Sorry, only JPG, JPEG and PNG are allowed.";
        $uploadOk = 0;
}
```

Checking if file has been uploaded correctly :

```
if ($uploadOk == 0) {
        echo "Sorry, your file was not uploaded.";
        // if everything is ok, try to upload file
        } else {
                if (move_uploaded_file($_FILES["userImage"]["tmp_name"], $target_file)) {
                        echo "The file ". basename( $_FILES["userImage"]["name"]) . " has been uploaded.";
                } else {
                echo "Sorry, there was an error uploading your file.";
        }
}
```

Sending mail with PHP

PHP allows you to create programs which makes it possible to send mail without having to use a mail sotfware nor a mail service, directly from a web site's page.

mail()

This function allows you to send emails directly from a script. Although, for the mail functions to be effective, a working email system must be installed and activated (program defined by the configuration settings of the php.ini file). It is usually installed and available on most servers.

However, you might have to change the configuration if you are working off-line with an application such as Mamp.

Syntax:
mail(*to,subject,message,headers,parameters*);

to	(Required) Email address of the receiver.
subject	(Required) Subject line of the message.
message	(Required) Message to be sent. Line break must be made using **\n**. Maximum line length of 70 characters.
headers	(Optional) Additional headers, (e.g.: From, Cc, Bcc). Additional headers must be separated using **\r\n**.
parameters	(Optional) Additional parameters to the *sendmail* program.

Example:

```php
<?php
$msg = "First line of the message\nSecond line of the message";
$msg = wordwrap($msg,70);        // wordwrap max. 70 characters
mail("someone@somewhere.com","My subject line",$msg);
?>
```

Explanation:
The two lines message of a maximum length of 70 characters per line would be sent to the specified email address with the specified subject line upon loading the page.
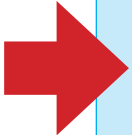
Using extra headers

```php
Example :

<?php
$to = "someone@somewhere.com";
$subject = "This is my subject";
$headers = "From: me@server.com" . "\r\n";
$headers .= "CC: someoneelse@server.com";
$msg = "First line of the message\nSecond line of the message.";
$msg = wordwrap($msg,70);

mail($to,$subject,$msg,$headers);
?>
```

Sending a HMTL email

```php
<?php
$to = "someone@somewhere.com";
$subject = "My HTML email";

$headers = "From: me@server.com" . "\r\n";
$headers .= "CC: someoneelse@server.com" . "\r\n";
$headers .= "MIME-Version: 1.0" . "\r\n";
$headers .= "Content-type:text/html;charset=UTF-8" . "\r\n";

$msg = "
        <html>
        <head>
        <title>My HTML email</title>
        </head>
        <body>
        <h1>Hello world!</h1>
        <p>This is my message</p>
        </body>
        </html>
";

mail($to,$subject,$msg,$headers);
?>
```

* Content-type must always be set when sending HTML email.

Final project :

Create a web page that will be updated using a form in an admin control pannel

index.php

Is the one-pager web site to update.

This page must contain a header and a footer, as well as a main an aside sections with different contents. The main section will have a title and a welcome text and an aside section showing a picture and a short description of the picture.

Use CSS to make the overall aspect as nice as possible.

admin.php

Is the form used to update the index page (admin control panel)

Fields will be used to change the title and the welcome word of the main section as well as the picture and the picture description of the aside section.