

Lecture 10 (w11)

2025/2026

Databases, Web Programming and Interfacing

DWPI

- Databases, Web Programming and Interfacing
 - An VI IT₄T
 - 1C/1L/1P
- Timetable
 - Friday, every week (fiecare saptamana) 1C + 2L (17-20)
 - Test + Project Presentation: **last week** (Weekend)

Grade

- 10% - Test/Exam – last week – 1h
- 40% - Project
 - Personal(80%)/Team(20%)
- Previous database project
 - receives a **web user interface**
 - **split** in individual assignments
 - bonus (individual) for **supplemental embedded systems access!**
 - admin/user – web user interface
 - data supplier – embedded system (even partial)

Info

- https://rf-opto.etti.tuiasi.ro/master_it.php

Laboratorul de Microunde si Opti x +

← → ↻ ⚠ Not secure | rf-opto.etti.tuiasi.ro/master_it.php



English | Romana

Main Courses **Master** Staff Research Students

Radiocommunication Systems Microwave IC Satellite Comm. Web Design Ethics

Databases, Web Programming and Interfacing

Course: DWPI (2021-2022)

Course Coordinator: Assoc.P. Dr. Radu-Florin Damian
Code: ITT.IA.601
Discipline Type: DIS; Required, Specialty
Credits: 5
Enrollment Year: 6, Sem. 11

Activities

Course: Instructor: Assoc.P. Dr. Radu-Florin Damian, 1 Hours/Week, Year, Timetable:
Laboratory: Instructor: Assoc.P. Dr. Radu-Florin Damian, 1 Hours/Week, Year, Timetable:
Project: Instructor: Assoc.P. Dr. Radu-Florin Damian, 1 Hours/Week, Year, Timetable:

Evaluation

Type: Verification

C: 10%, (Tests during semester)
C: 10%, (Tests during semester)
D: 40%, (Homework/Specialty papers)
D: 40%, (Homework/Specialty papers)

Materials

Textbooks

PHP5 and MySQL Bible (pdf, 15.97 MB, en, )
PAW 2021 Curs 1 (pdf, 15.1 MB, ro, )
PAW Curs 1 (video) (mp4, 467.67 MB, ro, )

Project/Design

Server CentOS pentru VMWare Player (cloud) (link, 0 Bytes, en, )
Instalare CentOS (pdf, 2.54 MB, en, )

Online – Registration no.

- access to **online exams** requires the **password** received by email

The password is communicated during the lectures. It is necessary to

Password

Registration no.

Name of the student

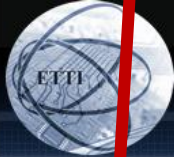
Proposed email 1

Proposed email 2

Write the code below

6fb6953

Send

 **RF-OPTO** 

English | Romana |

Main Courses Master Staff Research **Students**

Login Tutoring

Login

Use the Registration no. and your email or the password received by email

Registration no.

Email/Password

Write the code below

5dd64f9

Send

Online

- access email/password

Main Courses Master Staff Research

Grades Student List Exams Photos

POPESCU GOPO ION

**Fotografia
nu exista**

Date:

Grupa	5700 (2019/2020)
Specializarea	Inginerie electronica si telec
Marca	7000000

You access the site as **this student!**

Main Courses Master Staff Research

Grades Student List Exams Photos

POPESCU GOPO ION

**Fotografia
nu exista**

Date:

Grupa	5700 (2019/2020)
Specializarea	Inginerie electronica si telec
Marca	7000000

You access the site as **this student (including exams)!**

Online

- access email/password

Main Courses Master Staff Research

Grades Student List Exams Photos

POPESCU GOPO ION

Fotografia nu exista

Date:

Grupa	5700 (2019/2020)
Specializarea	Inginerie electronica si telec
Marca	7000000

You access the site as **this student!**

Main Courses Master Staff Research

Grades Student List Exams Photos

POPESCU GOPO ION

Fotografia nu exista

Date:

Grupa	5700 (2019/2020)
Specializarea	Inginerie electronica si telec
Marca	7000000

You access the site as **this student (including exams)!**

Password

■ received by email

Important message from RF-OPTO

Inbox x



Radu-Florin Damian

to me, POPESCU



Romanian

> English

[Translate message](#)



Laboratorul de Microunde si Optoelectronica
Facultatea de Electronica, Telecomunicatii si Tehnologia Informatiei
Universitatea Tehnica "Gh. Asachi" Iasi

In atentie: POPESCU GOPO ION

Parola pentru a accesa examenele pe server-ul **rf-opto** este

Parola: [REDACTED]

Identificati-va pe [server](#), cu parola, cat mai rapid, pentru confirmare.

Memorati acest mesaj intr-un loc sigur, pentru utilizare ulterioara

Attention: POPESCU GOPO ION

The password to access the exams on the **rf-opto** server is

Password: [REDACTED]

Login to the [server](#), with this password, as soon as possible, for confirmation.

Save this message in a safe place for later use

Reply

Reply all

Forward

Subject	Correspondents
Important message from RF-OPTO	POPESCU GOPO ION
Validation of MD/CR exam from 02/05/2020	[REDACTED]
[REDACTED]	[REDACTED]

From: Me <rdamian@etti.tuiasi.ro> ★

Subject: Important message from RF-OPTO

To: [REDACTED]

Cc: Me <rdamian@etti.tuiasi.ro> ★



Laboratorul de Microunde si Optoelectronica
Facultatea de Electronica, Telecomunicatii si Tehnologia Informatiei
Universitatea Tehnica "Gh. Asachi" Iasi

In atentie: POPESCU GOPO ION

Parola pentru a accesa examenele pe server-ul **rf-opto** este

Parola: [REDACTED]

Identificati-va pe [server](#), cu parola, cat mai rapid, pentru confirmare.

Memorati acest mesaj intr-un loc sigur, pentru utilizare ulterioara

Attention: POPESCU GOPO ION

The password to access the exams on the **rf-opto** server is

Password: [REDACTED]

Login to the [server](#), with this password, as soon as possible, for confirmation.

Save this message in a safe place for later use

Manual examen online

- The online exam app used for:
 - lectures (attendance)
 - laboratory
 - project
 - examinations

Materials

Other data

[Manual examen on-line](#) (pdf, 2.65 MB, ro, 🇷🇴)

[Simulare Examen](#) (video) (mp4, 65.12 MB, ro, 🇷🇴)

Microwave Devices and Circuits (Englis

Examen online

- always against a **timetable**
 - long period (project submission/laboratory results)
 - short period (tests: 15min, exam: 2h)

Announcement 23:59 (10/05/2020)	Support material 00:05 (11/05/2020)	Exam Topics 00:07 (11/05/2020)	Results 00:10 (11/05/2020)	End 00:20 (15/05/2020)	Confirmation 00:20 (16/05/2020)	Next timeframe in: 05 m 43 s Refresh now
---	---	--	--------------------------------------	----------------------------------	---	--

Announcement

This is a "fake" exam, introduced to familiarize you with the server interface and to perform the necessary actions during an exam: thesis scan, selfie, use email for co

Server Time

All exams are based on the server's time zone (it may be different from local time). For reference time on the server is now:

10/05/2020 23:59:16

Email addresses

- Team representative
 - list of **active** email addresses for all students
 - can be @student.etti.tuiasi.ro (also @gmail ~~@yahoo~~ etc.)
 - **rdamian@etti.tuiasi.ro**

2025/2026

Project

Project

- self selected homework: available until **20.12** (~2 weeks)
- **only** selection from a list after 20.12

Examene online: 2024/2025

Disciplina: DWPI (Databases, Web Programming and Interfacing (eng))

Pas 3

Nr.	Titlu	Start	Stop	Activ	Subiecte	Sustineri	Finalizate	C
1	Individual Subject Final Homework/Project	22/11/2024; 17:00	14/12/2024; 08:00	Activ	Homework project 2023.pdf	1	0	
2	Generic Test Exam	22/11/2024; 11:00	22/03/2025; 16:00	Activ	manual examen en 0.pdf	0	0	

Project

- Submission: **On-site**
- Presentation (in front of the colleagues) + files submission
- 3 files
 - **1 *.pdf** (print-screen while using the application, short usage instructions, a mini-user manual for the application) (**personal**)
 - **1 *.sql** (backup of the database required to run the application) (**team**)
 - archive of the application (inside: files *.php, *.jpg, folder tree etc., archived: ***.zip, *.7z** etc.) (**team/personal**)

Project grading

- **(2p)** the application runs on the **reference server** (can be downloaded from [rf-opto](#): Debian, php 8 **or** Ubuntu, php 7 **or** CentOS 7, php 5): extract files from the ***.zip** archive in a folder on the server, restore database from the ***.sql** backup file
- **(2p)** the ***.pdf** file containing the user manual exists and is appropriate for the submitted application
- **(2p)** the application **flowchart** has been submitted and contains appropriate data
- **(4p)** presentation on-site of the **application**

Online exam 05.12 - 20.12

- on-line exam for **choosing** the **individual** assignment **inside** the **team** (/individual) web **application**
- Data:
 - 1. **Selfie, file**, taken during the submission process (acting as a signature for the submission)
 - 2. **Title** of your (team) application/project, **text** (e.g. "Student Enrollment")
 - 3. **Team members, text** (e.g. "Popescu/Ionescu/Vasilescu")
 - 4. **Database, file**, Access (*.accdb) or SQL (*.sql), current database
 - 5. **Individual assignment, file**, Doc/Pdf/Txt describing the proposed individual activity (e.g. "I intend to design the scripts that will: 1. add records in the Students table + 2. extract the full list from Courses table + 3. list records from Students table following specified criteria")

Online exam 05.12 - 20.12

- While it is not required to keep the same team composition and/or the same database from the first half of the semester, it is **recommended**.
- The current database is not required to be kept identical until the end of the semester. Addition of tables/columns or change of data types is to be expected during application development. The current file is meant to be used only as support for understanding the proposed individual assignment.
- The individual assignment must not overlap with another team member assignment and cannot be changed during the semester. However, deviation from the assignment will be permitted (**only** if properly justified during the final presentation)
- A team integrated final application (same design, common application) will benefit from a bonus during the final submission. However individual grade will only depend on fulfillment of the individual assignment. Maximum grade can be obtained without the team bonus.

Online exam 19.12 – xx.o6

- on-line exam for the submission of the final individual work inside the team (/individual) web application.
- Data:
 - 1. **Selfie, file**, taken during the submission process (acting as a signature for the submission)
 - 2. **Title** of your (team) application/project, **text**
 - 3. **Team members, text**
 - 4. **PDF file, file**, includes a **flowchart** of your team application with identification of **individual assignment** inside the team application, and print-screen while using the application, short usage instructions, a mini-user manual for the application (**only** for your **individual assignment**)
 - 4. **SQL file, file**, (*.sql), backup of the final database required to run your submitted application
 - 5. **Archive file, file**, one archive of the application (inside: files *.php, *.jpg, folder tree etc., archived: *.zip, *.7z etc.)

Online exam 19.12 – xx.o6

- The files must be uploaded by **all team members, even if they are identical** (access to the common files is a minimal proof of team membership).
- The "SQL file" and "Archive file" can (should) be common, as is the case for the overall flowchart of the team application.
- However, the identification of individual assignment on the flowchart and the mini-user manual (print-screen, usage instructions) **must** be different between team members, without overlaps.

Introduction

Necessity

```
<td><?php echo $row['Documente'];?>&nbsp;  </td>
<td><?php echo $row['user_creat'];?>&nbsp;  </td>
<td class="smaller"><a href="control_lot.php?id=-<?php echo $row['ID_LOT'];?>">dezactiveaza</a><br /><a href=
"control_lot.php?id=<?php echo $row['ID_LOT'];?>">modifica</a></td>
</tr>
<?php $index++;
    } while ( $row = mysql_fetch_assoc($result));?>
</table><?php
}
else
{
    echo "<p>Nu exista loturi active</p>";
}
?>

<p class="title">Loturi inactive</p>
<?php
$query = "SELECT l.*, c.`nume_user` AS `user_creat`
FROM `lot` AS l
LEFT JOIN `users` AS c ON (l.`User`=c.`id_user`)
WHERE l.`Activ` = 0 ORDER BY l.`ID_LOT` DESC";
$result = mysql_query($query);
$total=0;
if ($result && (mysql_num_rows($result) > 0))
{
    $total=mysql_num_rows($result);
    $row = mysql_fetch_assoc($result);
}
if ($total>0)
{?>
<table align="center">
    <tr class="lista_titlu">
        <td align="center">Nu exista loturi inactive</td>
```

HTML

E:\Documents\Pagini Web\Safir dep\lista_lot.php - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Win

logfile.php lista_lot.php

```
539 <?php ?>
540
541 <div id="maincontent"><!-- InstanceBeginEditable name="Conti
542 <p><span class="title">Adauga</span><br/>
543 <a href="control_lot.php?id=0">Lot nou</a></p>
544 <p class="title">Loturi active</p>
545 <?php
546 $query = "SELECT l.*, c.`nume_user` AS `user_creat`
547 FROM `lot` AS l
548 LEFT JOIN `users` AS c ON (l.`User`=c.`id_user`)
549 WHERE l.`Activ` <> 0 ORDER BY l.`ID_LOT` DESC";
550 $result = mysql_query($query);
551 $total=0;
552 if ($result && (mysql_num_rows($result) > 0))
553 {
554 $total=mysql_num_rows($result);
555 $row = mysql_fetch_assoc($result);
556 }
557 if ($total>0)
558 {
559 <table align="center">
560 <tr class="lista_titlu">
561 <td align="center">Nr.</td>
562 <td>Numar</td>
563 <td>Data</td>
564 <td>Nr. pui</td>
565 <td>Gr. pui</td>
566 <td>Nr. pui morti</td>
567 <td>Gr. pui morti</td>
568 <td>Ferma</td>
569 <td>Documente</td>
570 <td>Creat</td>
571 <td>Comenzi</td>
572 </tr>
573 <?php $index=1;
574 do { ?>
575 <tr class="<?php if ($index%2) echo "lista_in
576 <td align="center"><?php echo $index; ?>&nbsp;</td>
577 <td><?php echo $row['Numar']; ?>&nbsp;</td>
578 <td><?php echo date("d/m/Y", strtotime($row['Data']));
579 <td><?php echo $row['Pui']; ?>&nbsp;</td>
580 <td><?php echo $row['Greutate']; ?>&nbsp;</td>
581 <td><?php echo $row['Pui_Morti']; ?>&nbsp;</td>
582 <td><?php echo $row['Greutate_Morti']; ?>&nbsp;</td>
583 <td><?php echo $row['Ferma']; ?>&nbsp;</td>
584 <td><?php echo $row['Documente']; ?>&nbsp;</td>
```

```
<table align="center">
<tr class="lista_titlu">
<td align="center">Nr.</td>
<td>Numar</td>
<td>Data</td>
<td>Nr. pui</td>
<td>Gr. pui</td>
```

HTML + PHP

```
E:\Documents\Pagini Web\Safir dep\lista_lot.php - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
logfile.php lista_lot.php
539 <?php ?>
540
541 <div id="maincontent"><!-- InstanceBeginEditable name="Continut" -->
542 <p><span class="title">Adauga</span><br/>
543 <a href="control_lot.php?id=0">Lot nou</a></p>
544 <p class="title">Loturi active</p>
545 <?php
546 $query = "SELECT l.*, c.`nume_user` AS `user_creat`
547 FROM `lot` AS l
548 LEFT JOIN `users` AS c ON (l.`User`=c.`id_user`)
549 WHERE l.`Activ` <> 0 ORDER BY l.`ID_LOT` DESC";
550 $result = mysql_query($query);
551 $total=0;
552 if ($result && (mysql_num_rows($result) > 0))
553 {
554 $total=mysql_num_rows($result);
555 $row = mysql_fetch_assoc($result);
556 }
557 if ($total>0)
558 {
559 <table align="center">
560 <tr class="lista_titlu">
561 <td align="center">Nr.</td>
562 <td>Numar</td>
563 <td>Data</td>
564 <td>Nr. pui</td>
565 <td>Gr. pui</td>
566 <td>Nr. pui morti</td>
567 <td>Gr. pui morti</td>
568 <td>Ferma</td>
569 <td>Documente</td>
570 <td>Creat</td>
571 <td>Comenzi</td>
572 </tr>
573 <?php $index=1;
574 do { ?>
575 <tr class="<?php if ($index%2) echo "lista_impar"; else echo "lista_p
576 <td align="center"><?php echo $index; ?>&nbsp;</td>
577 <td><?php echo $row['Numar']; ?>&nbsp;</td>
578 <td><?php echo date("d/m/Y", strtotime($row['Data'])); ?>&nbsp;</td>
579 <td><?php echo $row['Pui']; ?>&nbsp;</td>
580 <td><?php echo $row['Greutate']; ?>&nbsp;</td>
581 <td><?php echo $row['Pui_Morti']; ?>&nbsp;</td>
582 <td><?php echo $row['Greutate_Morti']; ?>&nbsp;</td>
583 <td><?php echo $row['Ferma']; ?>&nbsp;</td>
584 <td><?php echo $row['Documente']; ?>&nbsp;</td>
```

HTML + PHP + SQL

```
E:\Documents\Pagini Web\Safir dep\lista_lot.php - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
logfile.php lista_lot.php
539 <?php ?>
540
541 <div id="maincontent"><!-- InstanceBeginEditable name="Continut" -->
542 <p><span class="title">Adauga</span><br/>
543 <a href="control_lot.php?id=0">Lot nou</a></p>
544 <p class="title">Loturi active</p>
545 <?php
546 $query = "SELECT l.*, c.`nume_user` AS `user_creat`
547 FROM `lot` AS l
548 LEFT JOIN `users` AS c ON (l.`User`=c.`id_user`)
549 WHERE l.`Activ` <> 0 ORDER BY l.`ID_LOT` DESC";
550 $result = mysql_query($query);
551 $total=0;
552 if ($result && (mysql_num_rows($result) > 0))
553 {
554 $total=mysql_num_rows($result);
555 $row = mysql_fetch_assoc($result);
556 }
557 if ($total>0)
558 {?>
559 <table align="center">
560 <tr class="lista_titlu">
561 <td align="center">Nr.</td>
562 <td>Numar</td>
563 <td>Data</td>
564 <td>Nr. pui</td>
565 <td>Gr. pui</td>
566 <td>Nr. pui morti</td>
567 <td>Gr. pui morti</td>
568 <td>Ferma</td>
569 <td>Documente</td>
570 <td>Creat</td>
571 <td>Comenzi</td>
572 </tr>
573 <?php $index=1;
574 do { ?>
575 <tr class="<?php if ($index%2) echo "lista_impar"; else echo "lista_par";?>">
576 <td align="center"><?php echo $index; ?>&nbsp;</td>
577 <td><?php echo $row['Numar'];?>&nbsp;</td>
578 <td><?php echo date("d/m/Y", strtotime($row['Data']));?>&nbsp;</td>
579 <td><?php echo $row['Pui'];?>&nbsp;</td>
580 <td><?php echo $row['Greutate'];?>&nbsp;</td>
581 <td><?php echo $row['Pui_Morti'];?>&nbsp;</td>
582 <td><?php echo $row['Greutate_Morti'];?>&nbsp;</td>
583 <td><?php echo $row['Ferma'];?>&nbsp;</td>
584 <td><?php echo $row['Documente'];?>&nbsp;</td>
```

```
<?php
$query = "SELECT l.*, c.`nume_user` AS `user_creat`
FROM `lot` AS l
LEFT JOIN `users` AS c ON (l.`User`=c.`id_user`)
WHERE l.`Activ` <> 0 ORDER BY l.`ID_LOT` DESC";
$result = mysql_query($query);
$total=0;
if ($result && (mysql_num_rows($result) > 0))
{
$total=mysql_num_rows($result);
$row = mysql_fetch_assoc($result);
}
```

HTML

- Offers **the logical structure** of the document
- Required
 - the **final result** of any type of programming for a distributed application web is almost exclusively HTML code (HTML structure eventually manipulated by “client side scripting”/Javascript)
- **rf-opto Previous years** Curs 1 PAW 2021-2022 – Recapitulare HTML (RO + video)
- Course: Web Technologies (2012-2013) (EN)
 - <https://rf-opto.etti.tuiasi.ro/internet.php>
 - Web Technologies, Lecture 5
 - Web Technologies, Lecture 6

Learn more

- Web Hypertext Application Technology Working Group (WHATWG) HTML Living Standard
 - <https://html.spec.whatwg.org/>
- W3Schools Online Web Tutorials
 - <https://www.w3schools.com/html/>
 - <https://www.w3schools.com/css/>
 - + JavaScript, PHP, SQL, Python

Web application

Forms in HTML

- required for the user to send data to the server
- `<form>...</form>`
- Specific attributes:
 - action: address of the document which receives the data
 - `<form action="<?php echo $_SERVER['PHP_SELF'];?>">`
 - `<form action="actiuni/file2.html">`
 - method: method for transmitting data: post or get
 - `<form method="post" action= ... >`

Methods

- **post** : data is transmitted as a block (inside the body of the HTTP request)
- **get** : appends form-data into the URL :
`results.php?prob=81&an=2009`
- **get** must be used only for “idempotent” data,
 - no collateral effects
 - no change in server status (databases, etc)
- we can emulate a form (**get**) by writing links appropriately

Form elements

- inside `<form>...</form>`
 - input
 - select/option
 - textarea
 - button
- **all** elements will have a name
 - attribute: **name**=""
 - the name will be directly present in the URI generated by get, or will be a variable sent by post
- data is effectively sent by a **type="submit"** button

INPUT

- `<input .../>`
- Attributes:
 - type: text | password | checkbox | radio | submit | reset | file | hidden | image | button
 - name: variable name
 - value: data to send to the server on submitting the form (initial data in some cases)
 - checked/src/size/maxlength depending on type

Examples

```
<input name="textfield" type="text" value="ceva" />
```

```
<input name="Ok" type="submit" value="Trimite" />
```

```
<label><input name="check" type="checkbox" value="5" checked />check1</label>
```

```
<label><input name="RG1" type="radio" value="a" checked="checked" />but1</label>
```

```
<label><input name="RG1" type="radio" value="b" />but2</label>
```

```
<input name="hid" type="hidden" value="6" />
```

ceva

Trimite



check1



but1



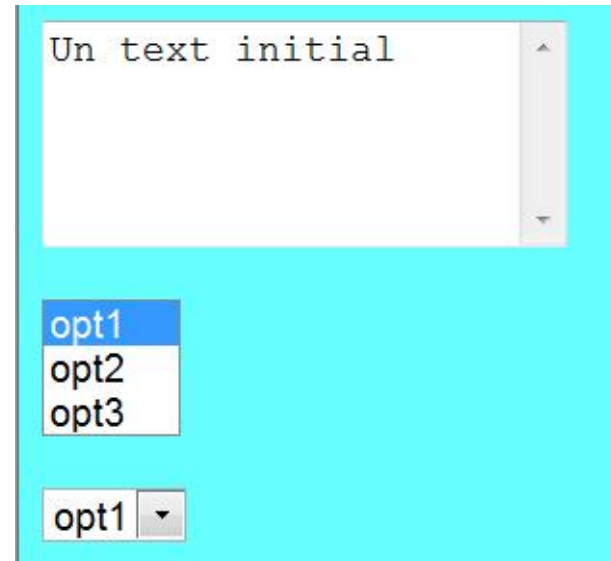
but2

TEXTAREA/SELECT

```
<textarea name="textarea" cols="20" rows="5">Un text  
initial</textarea><br /><br />
```

```
<select name="select." size="3">  
  <option value="1" selected="selected">opt1</option>  
  <option value="2">opt2</option>  
  <option value="3">opt3</option>  
</select><br /><br />
```

```
<select name="select...">  
  <option value="1" selected="selected">opt1</option>  
  <option value="2">opt2</option>  
  <option value="3">opt3</option>  
</select>
```



Un text initial

opt1
opt2
opt3

opt1

BUTTON

```
<button type="button" onClick="do ( );">Click Here</button>
```

```
<button type="submit" value="infoOnly">Request Info</button>
```

```
<button type="reset">Clear</button>
```

```
<button type="submit" id="sender"  
value="infoOnly">Request<br />Info<br /></button>
```

```
<button type="submit" id="sender"  
value="infoOnly"><p>Request</p><p><b><i>Info</i></b></p>  
</button>
```

- introduced to insert complex content in buttons (formatted text, images)
- Not always the best idea (Don't make me think!)

Click Here

Request Info 

 Clear

Request
Info



Request

Info



Exemplu – forma

- Mic magazin online
- Formular de comanda cu procesarea comenzii

```
<html>
<head>
<title>Magazin online XXX SRL</title>
</head>
<body>
<h1>Magazin online XXX SRL</h1>
<h2>Realizati comanda</h2>
<form action="rezultat.html" method="post">
<table border="0">
<tr bgcolor="#cccccc"><td width="150">Produs</td><td width="15">Cantitate</td></tr>
<tr><td>Carti</td><td align="center"><input type="text" name="carti_cant" size="3" maxlength="3" /></td></tr>
<tr><td>Caiete</td><td align="center"><input type="text" name="caiete_cant" size="3" maxlength="3" /></td></tr>
<tr><td>Penare</td><td align="center"><input type="text" name="penare_cant" size="3" maxlength="3" /></td></tr>
<tr><td colspan="2" align="center"><input type="submit" value="Trimite" /></td></tr>
</table>
</form>
</body>
</html>
```

Magazin online XXX SRL

Realizati comanda

Produs	Cantitate
Carti	1
Caiete	2
Penare	3

HTML 5 – Input validation



- `<input type="email">`
- `<input type="number">`
- `<input type="search">`
- `<input type="tel">`
- `<input type="url">`

HTML 5 – Input validation



```
<label for="email">Email:</label>
<input type="email" id="email" name="email">
<br /><br />
<label for="can">Cantitate (1 - 5):</label>
<input type="number" id="can" name="can" min="1" max="5">
<br /><br />
<label for="gsearch">Cautare Google:</label>
<input type="search" id="gsearch" name="gsearch">
<br /><br />
<label for="phone">Telefon:</label>
<input type="tel" id="phone" name="phone"
placeholder="0123.456.789" pattern="[0-9]{4}.[0-9]{3}.[0-9]{3}"
required>
<br /><br />
<label for="server">Server:</label>
<input type="url" id="server" name="server">
```

Email:

Cantitate (1 - 5):

Cautare Google:

Telefon:

Server:

HTML 5 – Input validation



Email:

Please include an '@' in the email address. 'ceva' is missing an '@'.

ceva@server

Cantitate (1 - 5):

Telefon:



Please match the requested format.

Server:



Please enter a URL.

Email:



Please include an '@' in the email address. 'ceva' is missing an '@'.

Telefon:

Server:



Please fill out this field.

Telefon:



Please match the requested format.

Server:

HTML 5 – Native browser UI



- `<input type="color">`
- `<input type="date">`
- `<input type="datetime-local">`
- `<input type="month">`
- `<input type="range">`
- `<input type="time">`
- `<input type="week">`

HTML 5 – Native browser UI



```
<label for="culoare">Culoare:</label>
<input type="color" id="culoare" name="culoare" value="#ff0000">
<br /><br />
<label for="nastere">Data nastere:</label>
<input type="date" id="nastere" name="nastere">
<br /><br />
<label for="examen">Examen (data si ora):</label>
<input type="datetime-local" id="examen" name="examen">
<br /><br />
<label for="fin">Finalizare (luna si an):</label>
<input type="month" id="fin" name="fin">
<br /><br />
<label for="vol">Volum (0 - 50):</label>
<input type="range" id="vol" name="vol" min="0" max="50">
<br /><br />
<label for="ora">Ora intalnire:</label>
<input type="time" id="ora" name="ora">
<br /><br />
<label for="sap">Saptamana:</label>
<input type="week" id="sap" name="sap">
```

Culoare:

Data nastere:

Examen (data si ora):

Finalizare (luna si an):

Volum (0 - 50):

Ora intalnire:

Saptamana:

HTML 5 – Native browser UI



Culoare:

Data nastere:

Examen (data):

Finalizare (luna):

Volum (0 - 50):

Ora intalnire:

Saptamana:

R G B ↕

Culoare:

Data nastere:

Examen (data):

Finalizare (luna):

Volum (0 - 50):

Ora intalnire:

March 2023

Su	Mo	Tu	We	Th	Fr	Sa
26	27	28	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18

HTML 5 – Native browser UI



Examen (data si ora):

Finalizare (luna si an): **March 2023** ↑ ↓

Volum (0 - 50):

Ora intalnire:

Saptamana:

Su	Mo	Tu	We	Th	Fr	Sa
26	27	28	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

[Clear](#) [Today](#)

02	21	PM
03	22	AM
04	23	
05	24	
06	25	
07	26	
08		

Finalizare (luna si an):

Volum (0 - 50):

Ora intalnire:

Saptamana:

2023

Jan	Feb	Mar	Apr
May	Jun	Jul	Aug
Sep	Oct	Nov	Dec

[Clear](#)

[This month](#)

HTML 5 – Native browser UI



Volum (0 - 50):



Ora intalnire:

05:30 PM

Saptamana: W

02 25

PM

03 26

AM

04 27

05 28

06 29

07 30

08 31

Trimite

Saptamana:

Week 13, 2023



Trimite

March 2023



Week	Su	Mo	Tu	We	Th	Fr	Sa
9	26	27	28	1	2	3	4
10	5	6	7	8	9	10	11
11	12	13	14	15	16	17	18
12	19	20	21	22	23	24	25
13	26	27	28	29	30	31	1
14	2	3	4	5	6	7	8

Clear

This week

Hypertext PreProcessor

PHP

Escaping from HTML

- A PHP file normally contains HTML tags (**IS** basically a HTML file), with some PHP code **sections** inside
- PHP interpreter
- the PHP interpreter looks for sections that it needs to interpret and their interior is processed as PHP code
- what is found **outside** these sections is sent to the web server **unmodified**
- most of the time the result is a virtual "text file" containing (pure) **HTML** code (+CSS/Javascript)

Escaping from HTML

- `<?php ... ?>`
 - XML type – default, always available, recommended
- `<? ... ?>`
 - short, usually deactivated
 - absent in PHP 7,8
- `<script language="php"> ... </script>`
 - script type, available
- `<% ... %>`
 - ASP type, usually deactivated
 - absent in PHP 7,8

Escaping from HTML

- **echo** a PHP language construct: output one or more strings (equivalent with puts() in C)
- can process data
 - `echo $a + $b;`
- in most cases “output” is the data sent to the client by the web server
- “output” can usually be considered:
 - the current document
 - that particular script position inside the document

Escaping possibilities

- All versions offer the same HTML source for the browser
- The one that leaves the HTML structure outside the escaping tags and only the dynamic data results from PHP processing is **recommended**
- HTML + PHP code is interpreted much more elegantly in WYSIWYG editors

```
<h2>Transaction result</h2>
<?php echo '<p>Command received</p>';?>
```

```
<h2>Transaction result</h2>
<p><?php echo 'Command received';?></p>
```

```
<?php echo '<h1>Online Shop XXX SRL</h1>';?>
<?php echo '<h2>Transaction result</h2>';?>
<?php echo '<p>Command received</p>';?>
```

```
<?php
echo '<h1>Online Shop XXX SRL</h1>';
echo '<h2>Transaction result</h2>';
echo '<p>Command received</p>';
?>
```

PHP – constants

- Like any programming language PHP is based on usage
 - constants
 - variables
 - functions
- Constants definition:
 - `define('BOOKPRICE', 100);`
- “case sensitive”
 - by convention, only in capital letters
 - `echo BOOKPRICE; // 100`

PHP – variables

- variable – **\$** character followed by the variable name
- the variable name is “case sensitive”
- a frequent mistake is forgetting the \$ char
 - PHP Notice: Use of undefined constant an – assumed \$an (**or 'an'**) in D:\\Server\\
- Data types
 - scalar
 - composite
 - special

PHP – data types

- scalar
 - Boolean
 - integer
 - float (double)
 - **string**
- composite
 - array
 - object
- special
 - resource
 - NULL

PHP – Variables

- declaring variables is **not required**, except when declaring a definition domain (eg. global variables)
 - `global $a, $b;`
`$c=$a+$b;`
- freeing the memory is not necessary, PHP does it automatically at the end of the script

PHP – Variables

```
$var = expression
```

- Control of variables is automatic, “on the fly”
 - A variable starts with the **\$** sign, followed by the name of the variable
 - PHP has no command for declaring a variable. It is created the moment you first assign a value to it
 - PHP automatically associates a data type to the variable, depending on its value
 - Variable names are case-sensitive (\$age and \$AGE are two different variables)

PHP – data types

- PHP automatically associates a data type to the variable, depending on its value
- automatic conversions are often **not** numerical but **“human”**
- In PHP 7, type declarations were added for function arguments, return values and class properties

```
<?php
echo $variable ; // type Null, not initialized – value NULL (only)
$variable = "0"; // $ variable  type string (ASCII 48)
$variable += 2; // $ variable  type integer (2)
$variable = $ variable + 1.3; // $variable  type float (3.3)
$variable = 5 + "10 objects"; // $variable  type integer (15)
$var2=5; // $var2  type integer (5)
$variable =$var2."10 objects"; // $variable  type string “510 objects”
?>
```

PHP – operators

- Similar (generally) to C/C++
- An operator takes one or more values (or expressions), performs an operation and yields another value
- Operators are divided in three groups
 - Unary operators: applied to one value
 - Binary operators: applied to two values
 - Ternary operators: applied to three values

PHP – operators

- Operators
 - Arithmetic
 - Assignment
 - Bitwise
 - Comparison
 - Incrementing/Decrementing
 - Logical
 - **String**

PHP – operators

- Arithmetic
 - `-$a` – Negation
 - `$a + $b` – Addition
 - `$a - $b` – Subtraction
 - `$a * $b` – Multiplication
 - `$a / $b` – Division
 - `$a % $b` – Modulo (remainder)
 - `$a ** $b` – Exponentiation
- **String**
 - **`$a.$b` – Concatenation String a with String b**

PHP – operators

■ Assignment

- `$a=$b`
- `$a+=$b` ($a=a+b$)
- `$a-=$b` ($a=a-b$)
- `$a/=$b` ($a=a/b$)
- `$a*=$b` ($a=a*b$)
- `$a%=$b` ($a=a\%b$)
- `$a**=$b` ($a=a^b$)
- **`$a.=$b` ($a=a$ concatenate b - strings)**

PHP – operators

- Bitwise operators
 - similar to C
 - `~, &, |, ^, <<, >>`
- Logical operators
 - offer a boolean result (always: true/false)
 - Some similar to C
 - `&&, ||, !`
 - Supplemental
 - `and, or, xor` – equivalent but with lower priority
 - `$a=55/0 or die('divide by 0');`

PHP – operators

- Comparison operators
 - most of the time offer a boolean result (true/false)
 - similar to C
 - `==` , `!=` , `>` , `<` , `<>` , `>=` , `<=`
 - Supplemental
 - `===` "Identical", equal value **AND** same type
 - `!==` "Not identical", different value **OR** different type
 - `<=>` "Spaceship", an integer -1/0/1 depending on relation between operands
 - `?:` "Ternary", `(expr1) ? (expr2) : (expr3)` evaluates to `expr2` if `expr1` evaluates to true, and `expr3` if `expr1` evaluates to false
 - `??` "Null Coalescing", `$var = expression ?? 'default value if NULL';`

Operator Precedence

non-associative	clone new	clone and new
left	[	array()
non-associative	++ --	increment/decrement
right	~ - (int) (float) (string) (array) (object) (bool) @	types
non-associative	instanceof	types
right	!	logical
left	* / %	arithmetic
left	+ - .	arithmetic and string
left	<< >>	bitwise
non-associative	< <= > >= <>	comparison
non-associative	== != === !==	comparison
left	&	bitwise and references
left	^	bitwise
left		bitwise
left	&&	logical
left		logical
left	? :	ternary
right	= += -= *= /= .= %= &= = ^= <<= >>=	assignment
left	and	logical
left	xor	logical
left	or	logical
left	,	many uses

Operator Precedence PHP8

non-associative	clone new	clone and new
left	[	array()
non-associative	++ --	increment/decrement
right	~ - (int) (float) (string) (array) (object) (bool) @	types
non-associative	instanceof	types
right	!	logical
left	* / %	arithmetic
left	+ - .	arithmetic and string
left	<< >>	bitwise
non-associative	< <= > >= <>	comparison
non-associative	== != === !== <=>	comparison
left	&	bitwise and references
left	^	bitwise
left		bitwise
left	&&	logical
left		logical
right	??	comparison
left	? :	ternary
right	= += -= *= /= .= %= &= = ^= <<= >>=	assignment
left	and	logical
left	xor	logical
left	or	logical
left	,	many uses

PHP – Functions

- conceptual similar to C/C++
- functions must not be declared before being used
- name of the functions is “case-insensitive”
- an extremely large number of functions useful in web applications exist in standard PHP libraries
- some libraries must be activated in php.ini
 - `extension=php_gd2.dll` // graphical processing functions
 - `extension=php_mysqli.dll` // MySQL access functions

Usage of PHP functions

- `<form action="result.php" method="post">`

`<p>Order received at date:`

`<?php echo date('d/m/Y')." time ".date('H:i');?></p>`

Magazin online XXX SRL

Rezultate comanda

Comanda receptionata la data: 10/03/2010 ora 13:36

`<body>`

`<h1>Shop online XXX SRL</h1>`

`<h2>Order result</h2>`

`<p>Order received at date:`

`10/03/2010 time 13:36</p>`

`</body>`

Control Structures

- most notions and syntax similar to C/C++
- statement-group: encapsulate a group of statements with curly braces {...}
- if / else / elseif – conditional execution

```
<?php
if ($a > $b) {
    echo "a mai mare ca b";
} elseif ($a == $b) {
    echo "a egal cu b";
} else {
    echo "a mai mic ca b";
}
?>
```

Control Structures

- while
- do-while
- for
- switch
- return
- break
- goto
- similar to C/C++ equivalents

```
$i = 1;  
while ($i <= 10) {  
    echo $i++;  
}
```

```
$i = 10;  
do {  
    echo $i--;  
} while ($i > 0);
```

```
for ($i = 1; $i <= 10; $i++) {  
    echo $i;  
}
```

```
switch ($i) {  
    case 0:  
        echo "i este 0";  
        break;  
    case 1:  
        echo "i este 1";  
        break;  
    default:  
        echo "i nici 1 nici 0";  
        break;  
}
```

Control Structures

- `include()`
- `require()`
- `include_once()`
- `require_once()`

- includes **and** evaluates the specified file
- used to avoid multiple evaluations of common code sections
- `require` will halt the script if the specified file is not found
- `..._once()` checks if the code from a file has already been included and it will **not** be included **again**

String Variables

PHP – data types

- scalar
 - Boolean
 - integer
 - float (double)
 - **string**
- composite
 - array
 - object
- special
 - resource
 - NULL

String Variables

- The ultimate goal of PHP is to populate the existing fields in an HTML skeleton with data (in the form of **text**)
- As a result, string data is treated more complex than the C/C++ counterpart
 - more ways of defining strings
 - more ways of interpreting strings
 - **much** more string related functions

String Variables

- string definition
 - single quoted: apostrophe ` `
 - double quoted: quotation marks " "
 - block definition
 - heredoc <<< X ... X;
 - nowdoc <<<'X' ... X;(PHP>5.3.0)

String Variables ` `

- single quotes ` ` are used for the definition of classic basic strings
 - a set of characters is defined
 - processing inside the string is reduced
 - \' is a literal single quote
 - \\ and \ are a literal backslash
 - **only!!!**

String Variables " "

- double-quotes " " are used for the definition of complex strings
 - complex processing inside the string more than the C/C++ equivalent
 - special ASCII characters, similar to C++: \n, \r, \t, \\\, \v, \e, \f, \x, \u
 - \" character double-quote
 - \\$ character \$
 - **variable names** inside the string will be expanded !!!

String Variables “ ”

- unescaped character dollar sign \$ means a variable name will follow
 - the parser will take as many tokens as possible to form a valid variable name (\$x, \$x->y, \$x[y])
 - if more complex variables are required (2 indexes array x[y][z] or non-integer indexes) enclose the variable name in curly braces {} to explicitly specify the end of the name

String Variables " "

- **simple** syntax for variable parsing

```
<?php
$juice = "apple";

echo "He drank some $juice juice.";
// He drank some apple juice.
echo "He drank some juice made of $juices.";
// He drank some juice made of . //s character valid pentru variabile

?>
```

String Variables " "

- **simple** syntax for variable parsing

```
<?php
$juices = array("apple", "orange", "koolaid1" => "purple");
class people {
    public $john = "John Smith";
}

$people = new people();
echo "$people->john drank some $juices[o] juice.";
// John Smith drank some apple juice.
?>
```

String Variables " "

- **complex** syntax for variable parsing {}

```
<?php
$juice = "apple";

echo "He drank some juice made of $juices.";
// He drank some juice made of . //s character valid pentru variabile
echo "He drank some juice made of ${juice}s."
// He drank some juice made of apples. // {} arata unde se incheie
numele variabilei
?>
```

String Variables “ ”

- **complex** syntax for variable parsing { }

```
<?php
$juices = array(array("apple", "orange"), "koolaid1" => "purple");
class people {
    public $name = "John Smith";
}

$obj->values[3] = new people();
echo "$obj->values[3]->name drank some $juices[0][1] juice.";
// drank some juice.
echo "{$obj->values[3]->name} drank some {$juices[0][1]} juice.";
// John Smith drank some apple juice.
?>
```

Here doc syntax

```
<?php
// no indentation
echo <<<END
    a
    b
    c
\n
END;

// 4 spaces of indentation
echo <<<END
    a
    b
    c
    END;

?>
```

Nowdoc syntax

- “Nowdocs are to single-quoted strings what heredocs are to double-quoted strings”
- **Multiline** string with minimal (**no**) processing

```
<?php
echo <<<'EOD'
Example of string spanning multiple lines
using nowdoc syntax. Backslashes are always treated
literally,
e.g. \\ and \'.
EOD;
?>
```

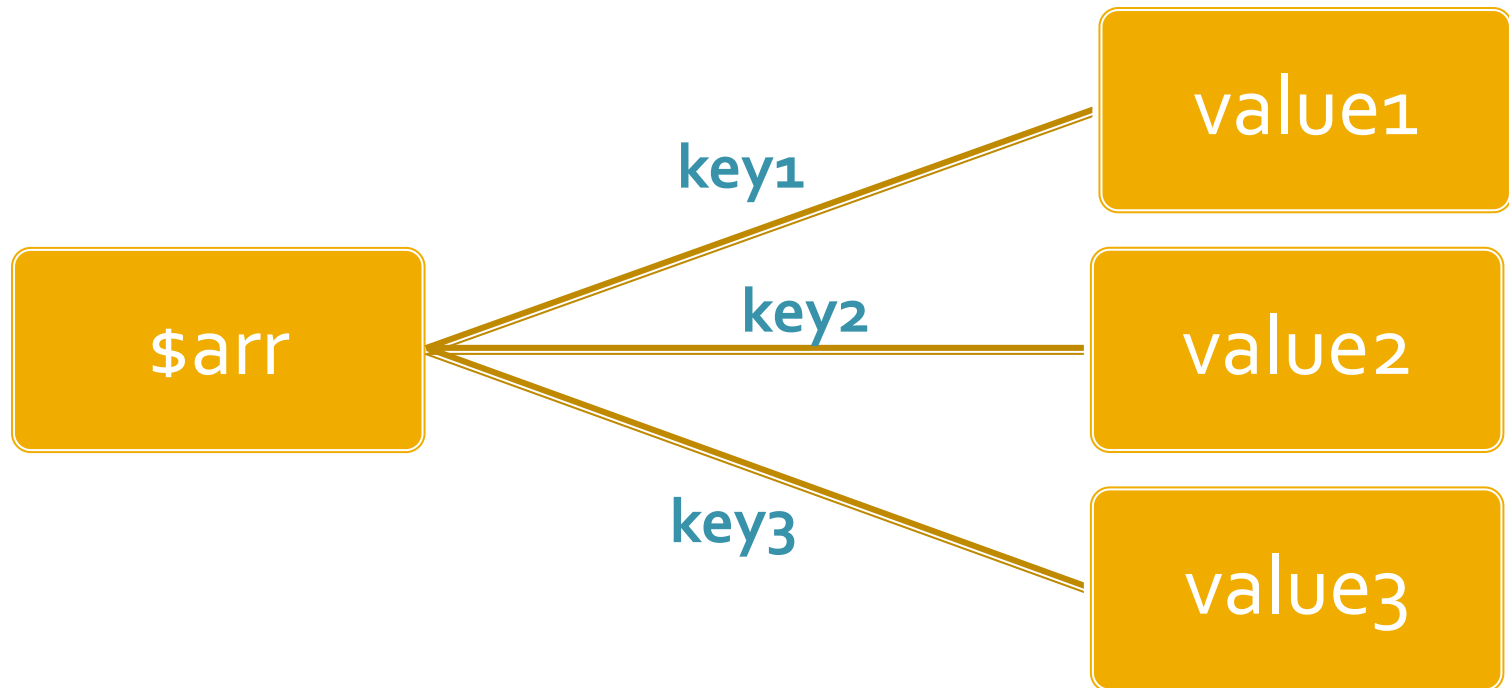
Arrays

Arrays in PHP

- An array in PHP is actually an ordered map. A map is a type that associates **values** to **keys**
- unlike C, Basic, **keys** are **not** required to be **integers**, can be **strings**
- default keys (if not otherwise specified) are consecutive integers with first key 0 (C syntax).
- defining a key / value pair
 - key => value
- create an array
 - \$arr = **array**("definition of key / value pairs")
 - pairs: key => value, key => value, ...

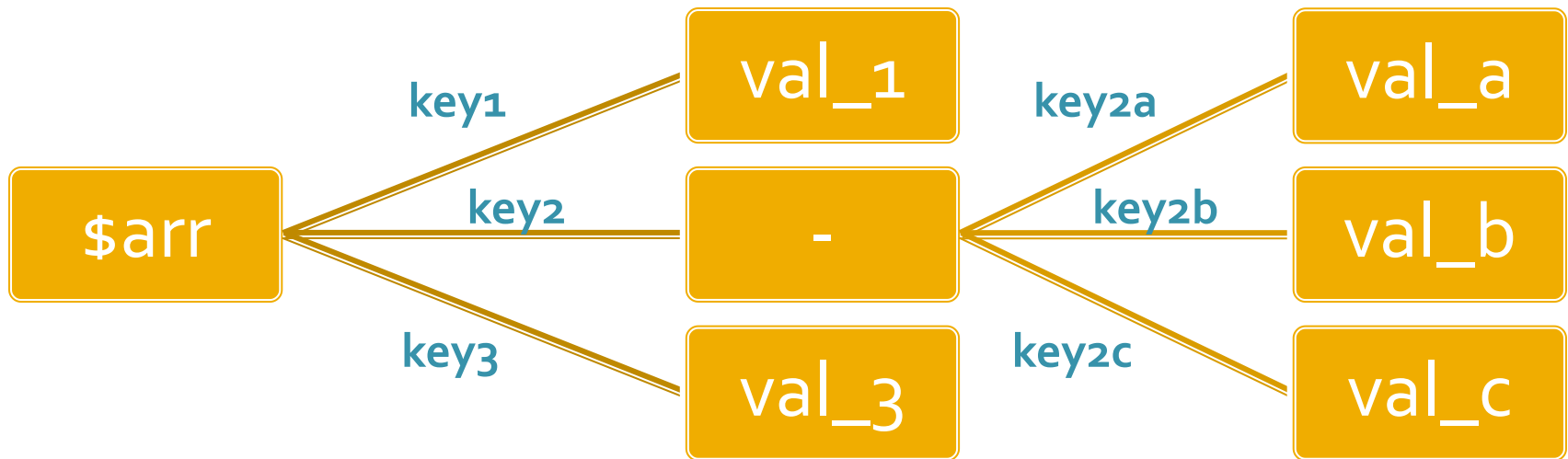
Array = Logical tree

- `$arr = array(key1 => value1, key2 => value2, key3 => value3)`



Array = Logical tree

- In particular, one or more of the values can in turn be an array, leading to **branching** of the tree
- `$arr = array(key1 => val_1, key2 => array(key2a => val_a, key2b => val_b, key2c => val_c), key3 => val_3)`



Arrays in PHP

```
$matr = array(1, 2, 3, 4, 5);
```

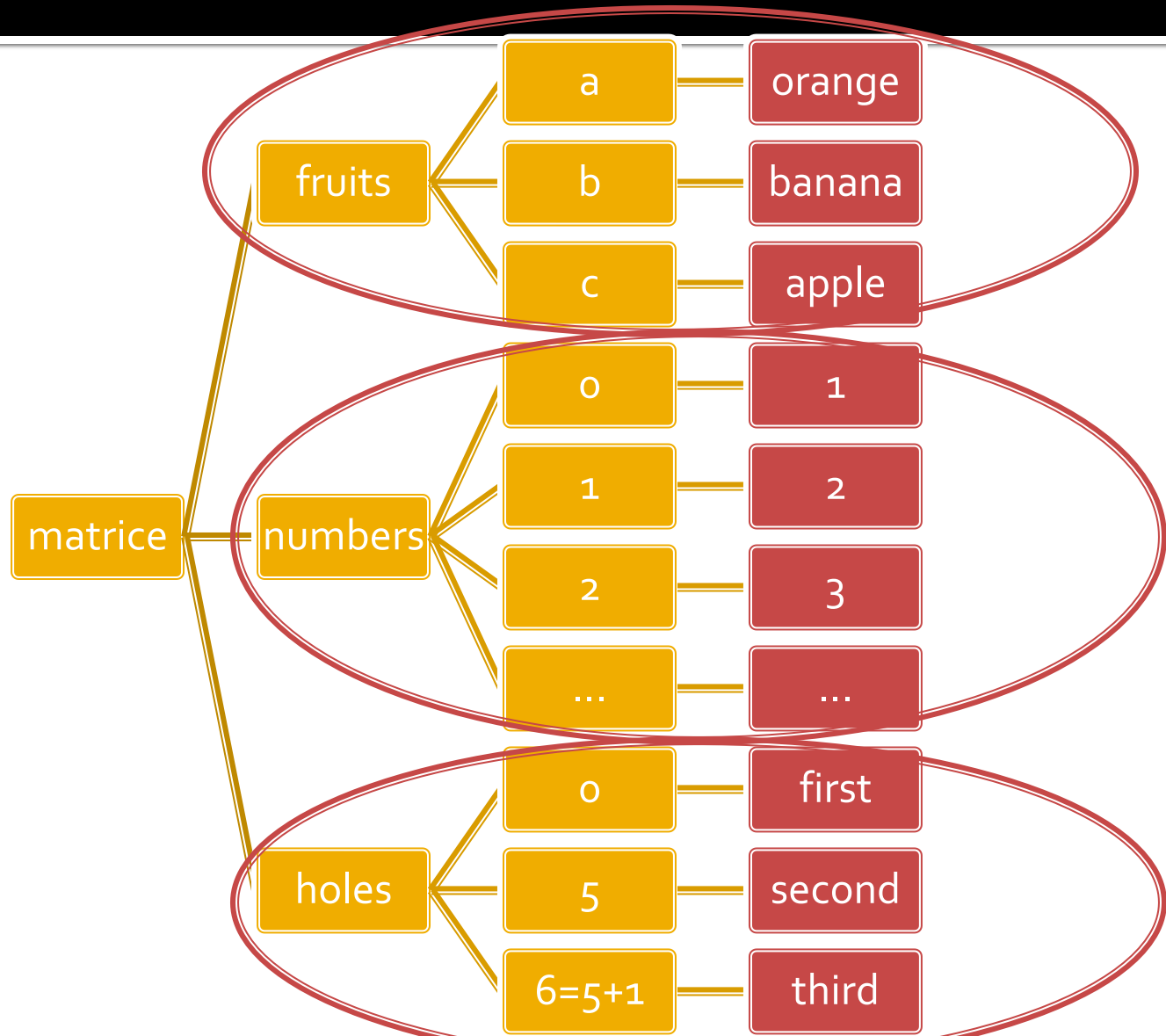
```
$matr[0]=1  
$matr[1]=2  
$matr[2]=3  
$matr[3]=4  
$matr[4]=5
```

```
$matr = array('a' => 1, 'b' => 2, 3, 4, 5);
```

```
$matr['a']=1  
$matr['b']=2  
$matr[0]=3  
$matr[1]=4  
$matr[2]=5
```

```
$matrice= array (  
    "fruits" => array("a" => "orange", "b" => "banana", "c" => "apple"),  
    "numbers" => array(1, 2, 3, 4, 5, 6),  
    "holes"  => array("first", 5 => "second", "third")  
);
```

Arrays in PHP



View array content (debug)

```
echo "<pre>";  
print_r ($matr);  
echo "</pre>";
```

```
$matr= array (  
"fruits" =>  
array("a" => "orange", "b" => "banana", "c" => "apple",  
"ultim"),  
"numbers" =>  
array(1, 2, 3, 4, 5, 6),  
"holes" =>  
array("first", 5 => "second", "third")  
);  
echo $matr;  
echo "<pre>";  
print_r ($matr);  
echo "</pre>";
```

```
Array  
  
Array  
(  
    [fruits] => Array  
        (  
            [a] => orange  
            [b] => banana  
            [c] => apple  
            [0] => ultim  
        )  
    [numbers] => Array  
        (  
            [0] => 1  
            [1] => 2  
            [2] => 3  
            [3] => 4  
            [4] => 5  
            [5] => 6  
        )  
    [holes] => Array  
        (  
            [0] => first  
            [5] => second  
            [6] => third  
        )  
)
```

Foreach loop

- `foreach (array_expression as $key => $value) statement`
- `foreach (array_expression as $value) statement`
- foreach construct is used to loop through each key/value pair in an array
- On each iteration assign the current element's key to the local variable `$key` and the value of the current element is assigned to the local variable `$value` (scope: statement)
- `foreach()` works with a **copy** of the array, you cannot change the original array inside the statement
 - ```
foreach ($arr as $key => $value) {
 $value = 'other value'; //doesn't work
 $arr[$key] = 'other value'; //works
}
```

# Example – foreach

```
$matr = array (
 "fruits" => array("a" => "orange", "b" => "banana", "c" => "apple", "ultim"),
 "numbers" => "in loc de numere",
 "holes" => "in loc de ce era"
);
foreach ($matr as $scheie => $continut)
 echo "matr[".$scheie."]=".$continut."
";
```

```
matr[fruits]=Array
matr[numbers]=in loc de numere
matr[holes]=in loc de ce era
```

# PHP Global Variables - Superglobals

# PHP Global Variables - Superglobals

- PHP Global Variables - Superglobals (predefined variables)
  - are always accessible, regardless of scope
  - Examples:
    - **\$\_SERVER** — Server and execution environment information
    - **\$\_GET** — HTTP GET variables
    - **\$\_POST** — HTTP POST variables
    - **\$\_FILES** — HTTP File Upload variables
    - **\$\_REQUEST** — HTTP Request variables
    - **\$\_SESSION** — Session variables
    - **\$\_ENV** — Environment variables
    - **\$\_COOKIE** — HTTP Cookies

# Forms in HTML

- required for the user to send data to the server
- `<form>...</form>`
- Specific attributes:
  - action: address of the script which receives the data
    - `<form action="<?php echo $_SERVER['PHP_SELF'];?>">`
    - `<form action="actiuni/file2.html">`
  - method: method for transmitting data: post or get
    - `<form method="post" action= ... >`

# Methods

- **post** : data is transmitted as a block (inside the body of the HTTP request)
- **get** : appends form-data into the URL :  
`results.php?prob=81&an=2009`
- **get** must be used only for “idempotent” data,
  - no collateral effects
  - no change in server status (databases, etc)
- we can emulate a form (**get**) by writing links appropriately

# Form elements

- inside `<form>...</form>`
  - input
  - select/option
  - textarea
  - button
- **all** elements will have a name
  - attribute: **name**=""
  - the name will be directly present in the URI generated by get, or will be a variable sent by post
- data is effectively sent by a **type="submit"** button

# Examples

```
<input name="textfield" type="text" value="ceva" />
```

```
<input name="Ok" type="submit" value="Trimite" />
```

```
<label><input name="check" type="checkbox" value="5" checked />check1</label>
```

```
<label><input name="RG1" type="radio" value="a" checked="checked" />but1</label>
```

```
<label><input name="RG1" type="radio" value="b" />but2</label>
```

```
<input name="hid" type="hidden" value="6" />
```

ceva

Trimite



check1



but1



but2

# TEXTAREA/SELECT

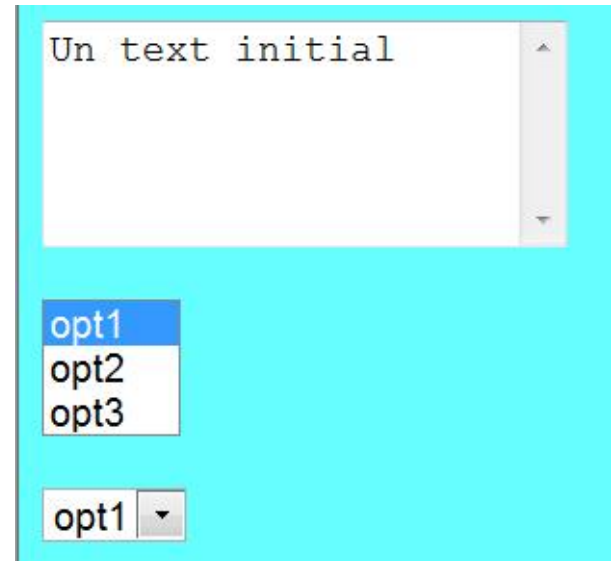
```
<textarea name="textarea" cols="20" rows="5">Un text
initial</textarea>


```

```
<select name="select." size="3">
 <option value="1" selected="selected">opt1</option>
 <option value="2">opt2</option>
 <option value="3">opt3</option>
</select>


```

```
<select name="select...">
 <option value="1" selected="selected">opt1</option>
 <option value="2">opt2</option>
 <option value="3">opt3</option>
</select>
```



Un text initial

opt1  
opt2  
opt3

opt1 ▼

# Getting user submitted data

- When a user submits the data by clicking on "Submit", the form data is found in the file specified in the **action** attribute of the <form> tag in one of the superglobal variables:
  - \$\_POST – method="post"
  - \$\_GET – method="get"
  - \$\_REQUEST – both methods
- the superglobal variables are **arrays** with **string keys** controlled by the **name** attribute of the input element
  - <input type="text" name="**books\_quant**" size="3" maxlength="3" />
  - \$\_POST['**books\_quant**'] contains the user input in the receiving script

# Organizing \$\_POST

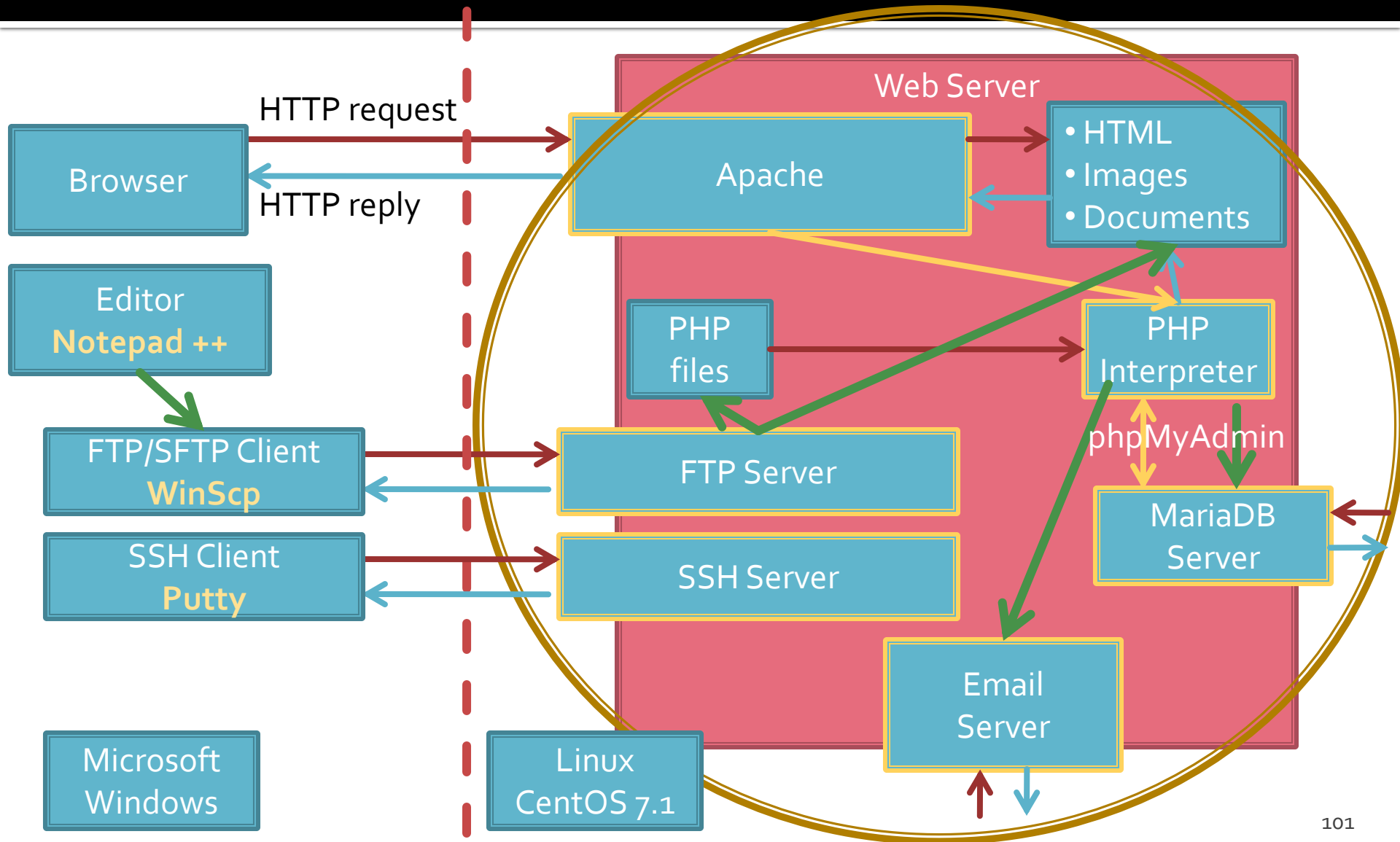
- **name** attributes in the form inputs become **keys** in the superglobal array `$_POST`
  - `<input type="text" name="books_quant" size="3" maxlength="3" />`
  - `$_POST['books_quant']` contains the user input
- creating **name** "array like", we can control branching of `$_POST` grouping input elements in the form as required
  - `<input type="text" name="quant[books]" size="3" maxlength="3" />`
  - `$_POST['quant']['books']` contains the user input

# Documentation

- <https://www.php.net/>
- [http://rf-opto.etti.tuiasi.ro/master\\_it.php](http://rf-opto.etti.tuiasi.ro/master_it.php)

# Reference server and applications

# Using LAMP



# Using LAMP

- differences from a Windows computer
  - system commands difficult
    - command prompt, SSH, Putty
  - files submitted by FTP
    - Copy/Paste unavailable
  - administration of MySql server:
    - using PhpMyAdmin (preloaded)
    - using other software on the host computer (eg. MySQL Workbench)

# Using LAMP – Advantages

- Advantages
  - Available applications have newer versions (**2020 ~**)
    - CentOS/7.1, Apache/2.4.6, PHP/5.4.16, MariaDB/5.5.44, PhpMyAdmin/4.4.15
    - Ubuntu/20.04, Apache/ 2.4.41, PHP/ 7.4.3, MariaDB/ 10.3.31, PhpMyAdmin/4.9.5
    - Debian/12.5, Apache/ 2.4.57, PHP/ 8.2.7, MariaDB/ 10.11.6, PhpMyAdmin/5.2.1
  - Available applications very similar with real life hosting solutions
    - SSH
    - FTP
    - Email
      - for full functionality the network interface for the VM must be changed  
**Host-only** -> **Bridged**

# Reference Server

- rf-opto.etti.tuiasi.ro > Master > Web Design

## Project/Design

[VMware Workstation Player](#) (link, 0 Bytes, en, )

[Ubuntu VM for VMWare](#) (link, 0 Bytes, en, )

[Ubuntu Setup](#) (pdf, 1.83 MB, en, )

[Centos VM for VMWare](#) (link, 0 Bytes, en, )

[Centos Setup](#) (pdf, 2.54 MB, en, )

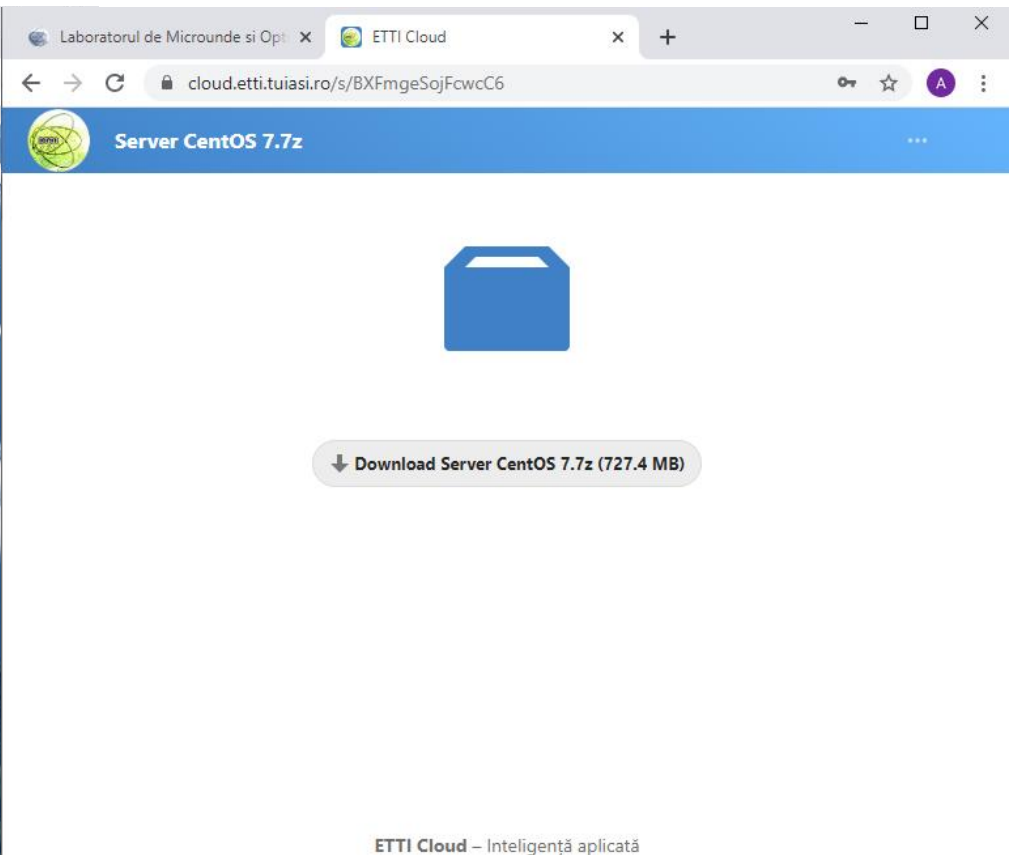
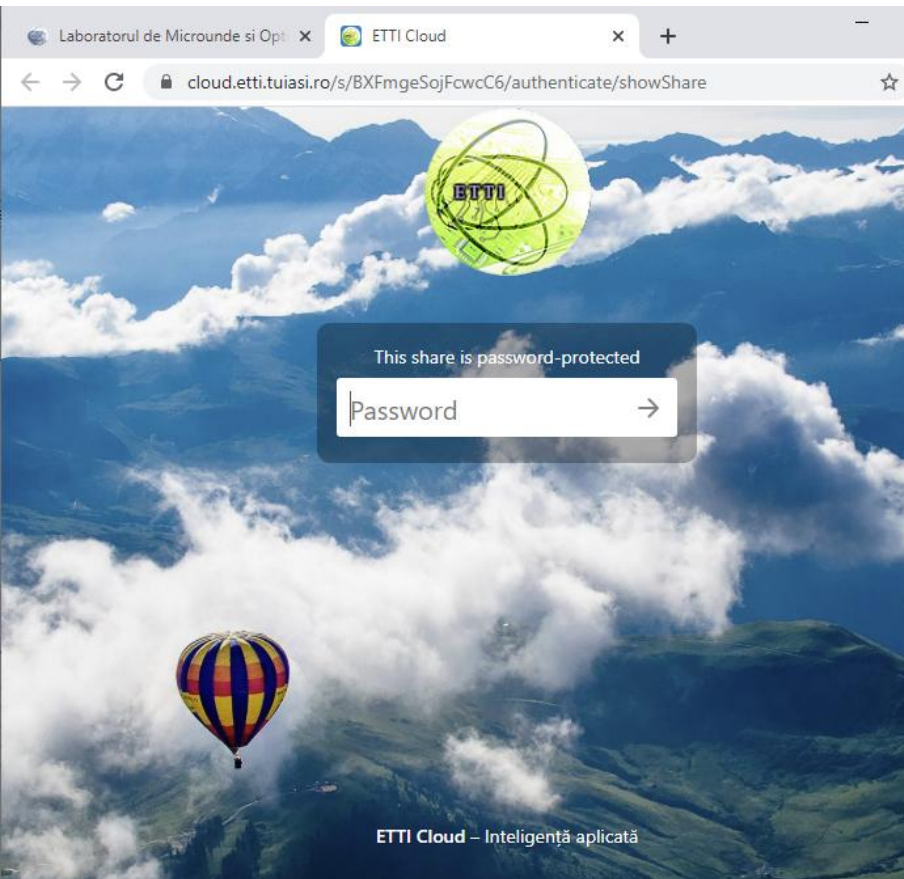
## Examen

[Online Exam manual](#) (pdf, 2.56 MB, en, )

[Manual examen on-line](#) (pdf, 2.65 MB, ro, )

# Reference Server

## ■ Cloud ETTI: RF-opto3#



# Server referinta

- Masina virtuala
- VMware Workstation Player
  - Gratuit (non-comercial)
  - <https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html>
- Inlocuit de VMware Workstation Pro



## VMware Workstation Pro for PC

Build and test nearly any app with the world's leading desktop hypervisor app for Windows and Linux.

[DOWNLOAD NOW >](#)

# Server referinta

- Masina virtuala
- VMware Workstation Pro (Broadcom)



## VMware Workstation Pro for PC

Build and test nearly any app with the world's leading desktop hypervisor app for Windows and Linux.

[DOWNLOAD NOW >](#)



Broadcom Inc. Customer sign-in

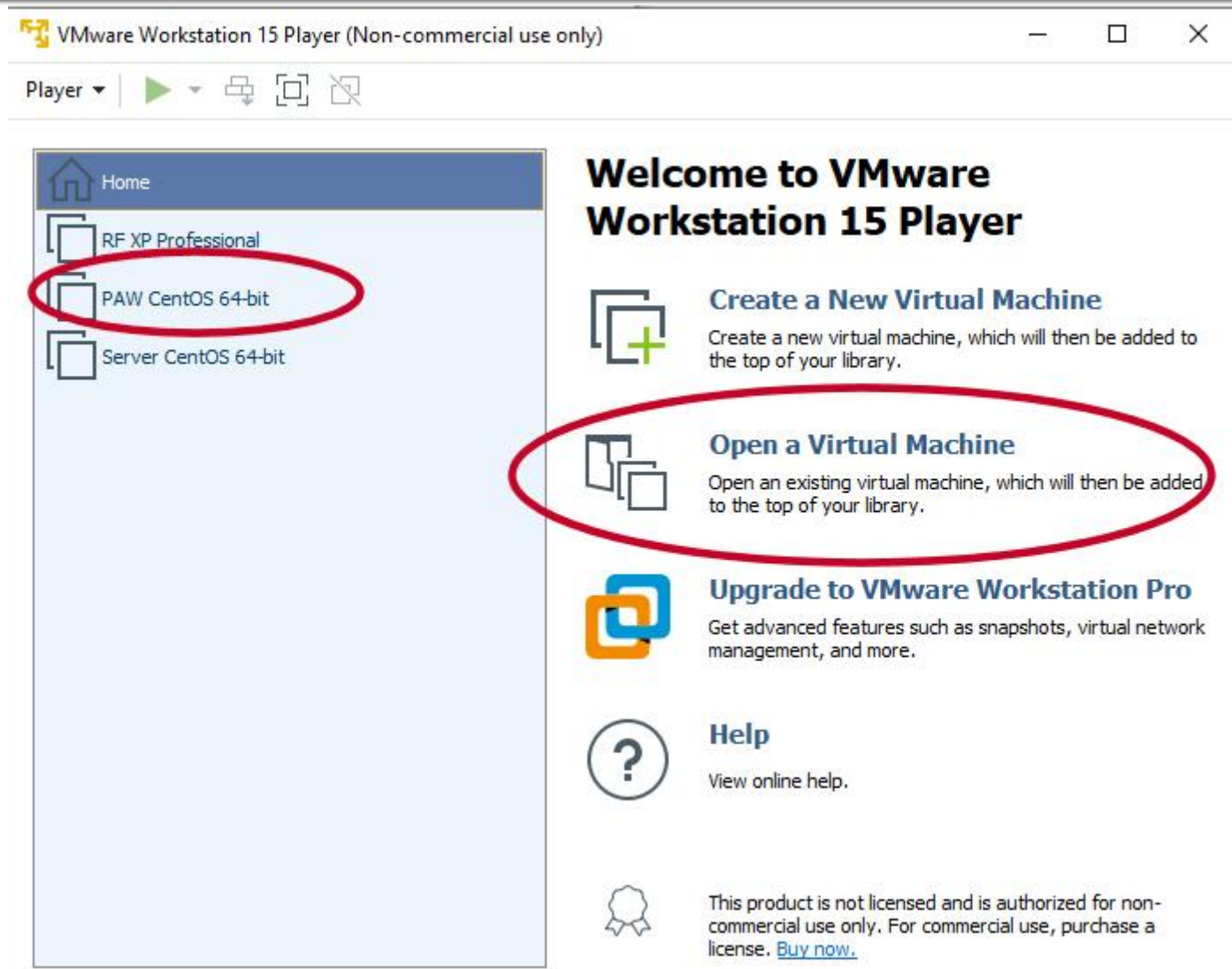
**Username**

Enter your username

☐ Remember me

Next

# Reference Server

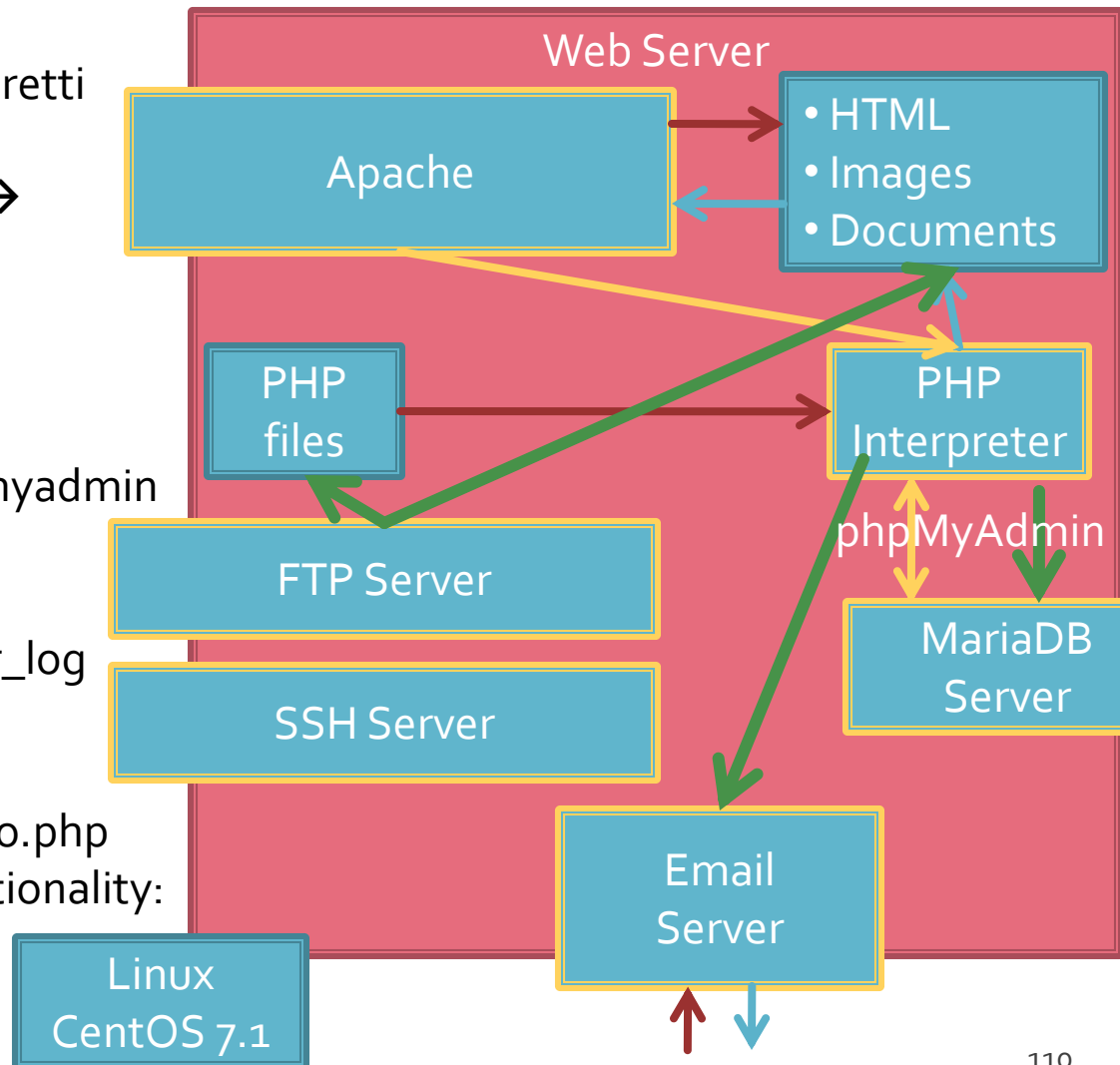


# Possible problems

- Current VMWare Player runs **only** on **64bit** operating systems Windows/Linux
  - for 32bit operating systems previous (**certified originals**) can be made available on rf-opto
- The host computer **must** enable **Hardware Virtualization**
  - Hardware Virtualization is enabled in BIOS, depending on the PC manufacturer: Processor, Chipset, Northbridge
  - Options name: VT-x, AMD-V, Vanderpool, Hyper-V, SVM, Intel Virtualization Technology. if available: Intel VT-d, AMD IOMMU
- VM archive requires **7zip** native to the target operating system

# Using LAMP

1. login → root:masterrc / paw:masteretti
2. ifconfig → 192.168.30.5
3. putty.exe → 192.168.30.5 → SSH → root:masterrc (remote login)
4. [other linux command line]
5. FTP → Winscp → SFTP → student:masterrc@192.168.30.5
6. MySql → http://192.168.30.5/phpmyadmin → root:masterrc / root:masteretti
7. Apache Error Log →
  - 7a. putty → nano /var/log/httpd/error\_log
  - 7b. http://192.168.30.5/logfile.php (nonstandard)
8. PHP info → http://192.168.30.5/info.php
9. if DHCP service stops Apache functionality:  
service httpd restart



# LAMP Reference Server 2023

- Linux, two variants
  - Centos 7.1
    - PHP 5.4.16
    - MariaDB 5.5.44
    - Apache 2.4.6
    - **root**/student:masterrc
  - Ubuntu 20.04 (**recommended**)
    - PHP 7.4.3
    - MariaDB 10.3.31
    - Apache 2.4.41
    - **paw**/student:masteretti
    - correction **paw FTP access**:
      - sudo usermod -a -G upload paw
      - sudo chmod -R 775 /var/www

# LAMP Reference Server 2025

- Linux, three proposed servers
  - CentOS 7.1
  - Ubuntu 20.04
  - **Debian 12.5**

# LAMP Reference Server

- Centos 7.1
  - PHP 5.4.16
  - MariaDB 5.5.44 / root:masterrc
  - Apache 2.4.6
  - PhpMyAdmin/4.4.15
  - **root**/student:masterrc
  - Python 2.7.5
  - creat: Workstation Player 12.x (**12**)

# LAMP Reference Server

- Ubuntu 20.04
  - PHP 7.4.3
  - MariaDB 10.3.31 / root:masteretti
  - Apache 2.4.41
  - **paw**/student:masteretti
  - necesar suplimentar pentru **acces FTP user paw**:
    - `sudo usermod -a -G upload paw`
    - `sudo chmod -R 775 /var/www`
  - Python 3.8.10
  - creat: Workstation Player 15.x (**16**)

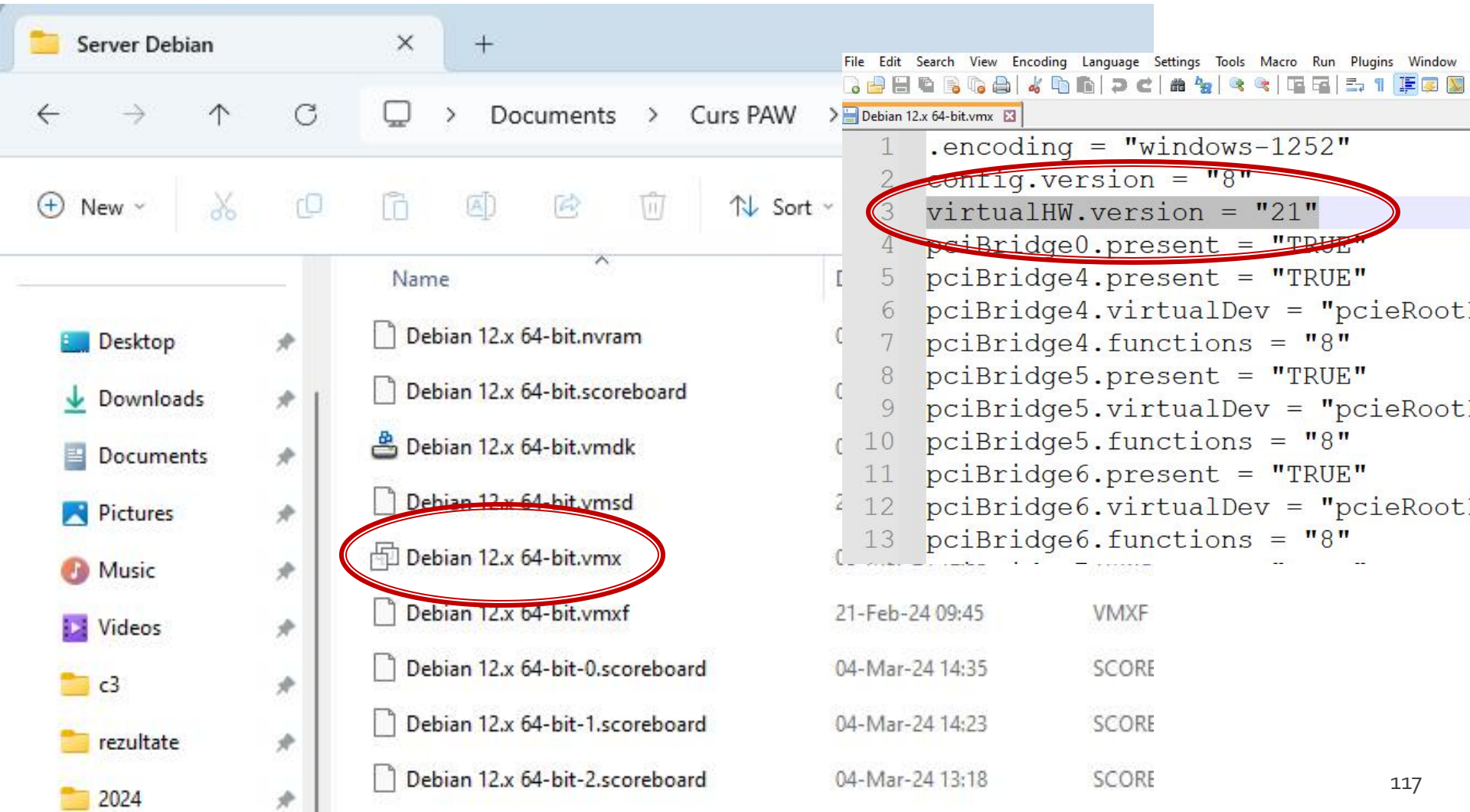
# LAMP Reference Server

- Debian 12.5
  - PHP 8.2.7
  - MariaDB 10.11.6 / root:masteretti
  - Apache 2.4.57
  - PhpMyAdmin/5.2.1 deb
  - **root**/paw/student:masteretti
  - Python 3.11.2
  - creat: Workstation Player 17.5 (**21**)

# Server referinta

- Pentru rularea unui server pe o versiune VMware Player anterioara:
  - se localizeaza fisierul "\*.vmx" a server-ului
  - se modifica virtualHW.version = "**21**" la o valoare mai mica (anterioara)
    - in 2.13 -> **18**

# LAMP Reference Server



# Support applications

- WinSCP (client FTP, gratuit)
  - <https://winscp.net/eng/download.php>
- Notepad ++ (editor, avansat, gratuit)
  - <https://notepad-plus-plus.org/downloads/>
- Putty (remote access)
  - <https://www.putty.org/>
- MySQL Workbench (gratuit, cont Oracle)
  - <https://www.mysql.com/products/workbench/>
- Visual Studio Code (gratuit, Microsoft)
  - <https://code.visualstudio.com/download>

# Aplicatii suport

- Variante portabile

## Laboratory

[Laborator 1](#) (pdf, 1.48 MB, ro, )

[Server Win2000 pentru VMWare Player - lab 1 \(cloud\)](#) (link, 0 Bytes, en, )

[Accesorii laborator \(x32 - partial\)](#) (zip, 28.92 MB, en, )

[Accesorii laborator \(x64 - complet\)](#) (zip, 133.58 MB, en, )

## Project/Design

[VMware Workstation Player](#) (link, 0 Bytes, en, )

[Server CentOS pentru VMWare Player \(cloud\)](#) (link, 0 Bytes, en, )

[Instalare Centos](#) (pdf, 2.54 MB, en, )

[Server Ubuntu pentru VMWare Player \(cloud\)](#) (link, 0 Bytes, en, )

[Instalare Ubuntu](#) (pdf, 1.83 MB, en, )

# IP address

- login, ifconfig
- Ctrl + Alt + mouse

PAW CentOS 64-bit - VMware Workstation 15 Player (Non-commercial use only)

Player ▾ || ▾ ⏏ ⏏ ⏏ ⏏

```
CentOS Linux 7 (Core)
Kernel 3.10.0-229.20.1.el7.x86_64 on an x86_64

tmpaw login: root
Password:
Last login: Wed Jun 17 05:35:16 from 192.168.0.106
[root@tmpaw ~]# ifconfig
```

PAW CentOS 64-bit - VMware Workstation 15 Player (Non-commercial use only)

Player ▾ || ▾ ⏏ ⏏ ⏏ ⏏

```
CentOS Linux 7 (Core)
Kernel 3.10.0-229.20.1.el7.x86_64 on an x86_64

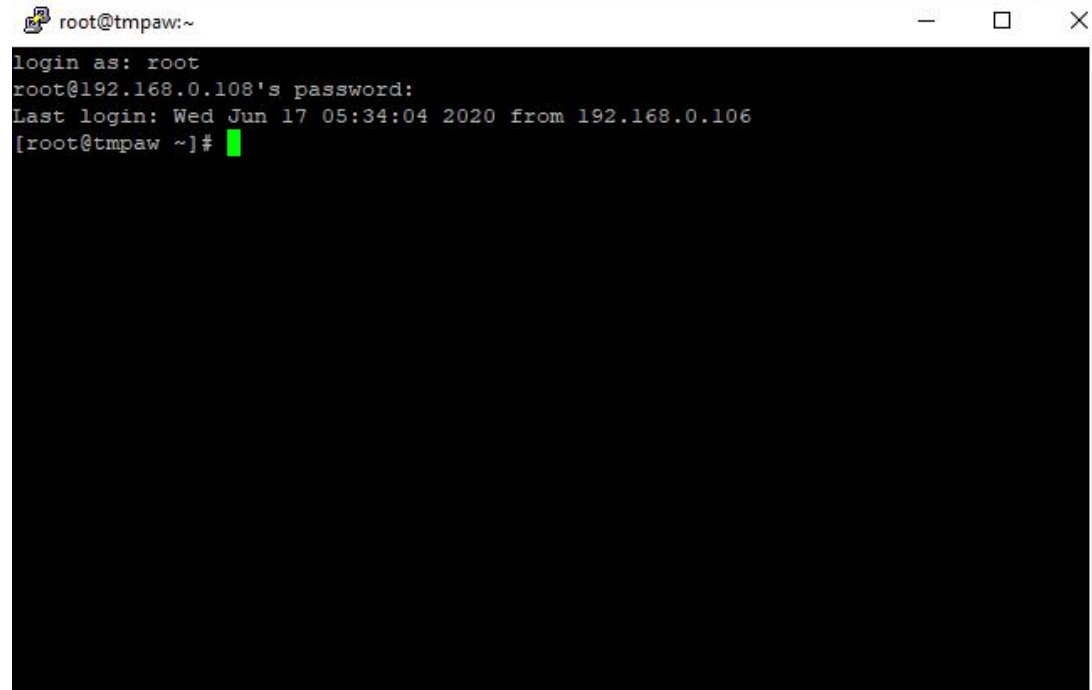
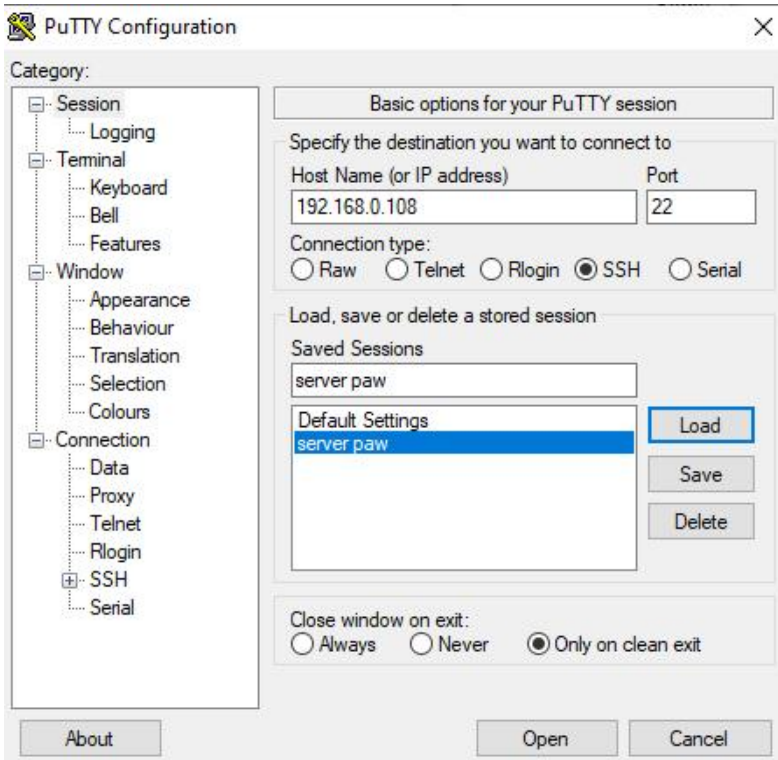
tmpaw login: root
Password:
Last login: Wed Jun 17 05:35:16 from 192.168.0.106
[root@tmpaw ~]# ifconfig
eno16777736: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
 inet 192.168.0.108 netmask 255.255.255.0 broadcast 192.168.0.255
 inet6 fe80::250:56ff:fe3e:1693 prefixlen 64 scopeid 0x20<link>
 ether 08:00:56:3e:16:93 txqueuelen 1000 (Ethernet)
 RX packets 104 bytes 12814 (12.5 KiB)
 RX errors 0 dropped 0 overruns 0 frame 0
 TX packets 99 bytes 11847 (11.5 KiB)
 TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
 inet 127.0.0.1 netmask 255.0.0.0
 inet6 ::1 prefixlen 128 scopeid 0x10<host>
 loop txqueuelen 0 (Local Loopback)
 RX packets 16 bytes 1774 (1.7 KiB)
 RX errors 0 dropped 0 overruns 0 frame 0
 TX packets 16 bytes 1774 (1.7 KiB)
 TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@tmpaw ~]# _
```

# Putty

- putty.exe
- avoids mouse capture (CentOS), copy/paste etc.



# WinSCP

- FTP client
- upload files

Session

File protocol:  
SFTP

Host name: 192.168.0.108 Port number: 22

User name: student Password: .....

Edit Advanced...

Login Close Help

html - student@192.168.0.108 - WinSCP

File Commands Mark Session View Help

Address /var/www/html

Find Files Download Edit Properties New Synchronize

Transfer Settings Default

student@192.168.0.108 x New Session

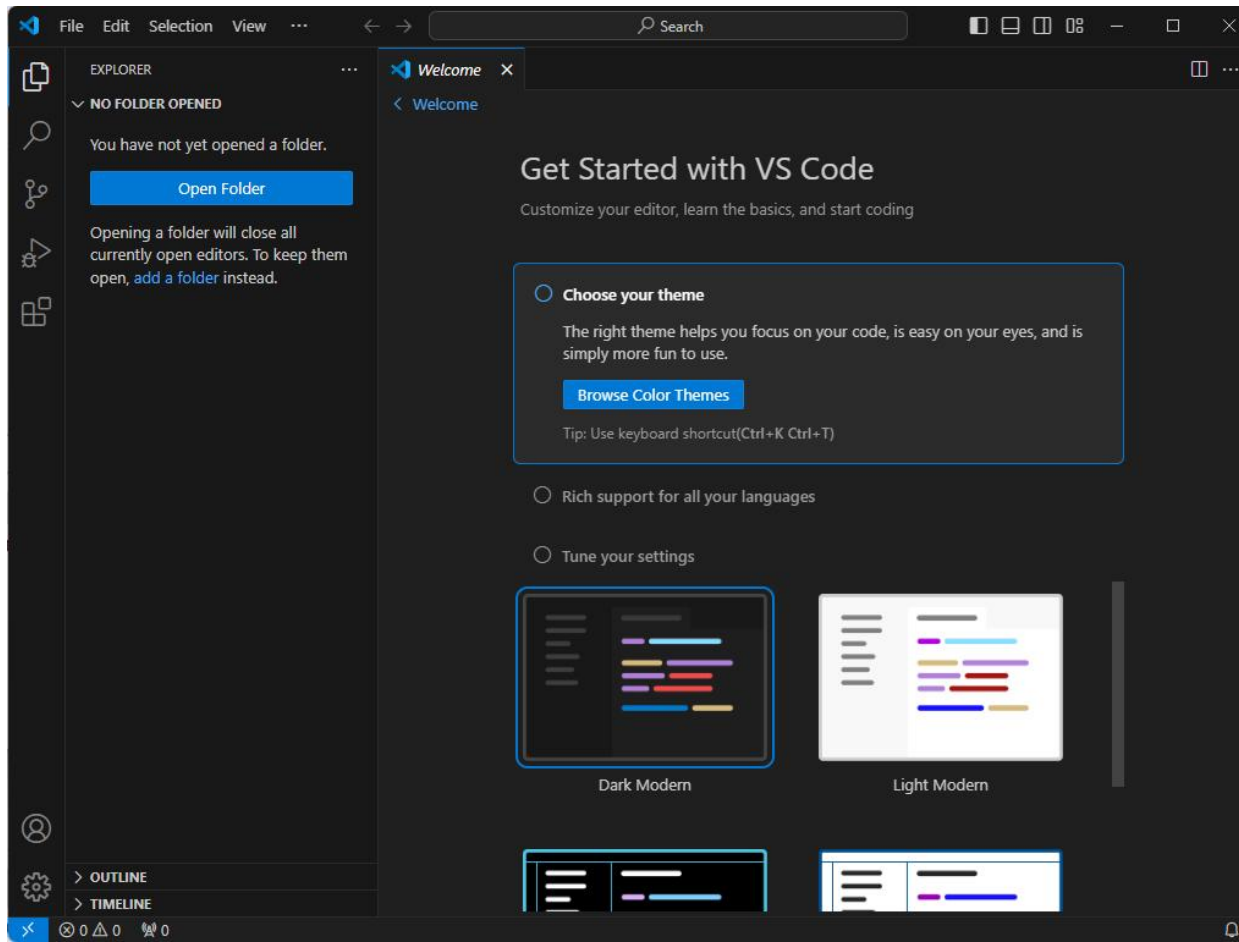
Name	Size	Changed	Rights
ap.log	1 KB	2/29/2016 11:28:50 AM	rw-rw-r
info.php	1 KB	9/30/2009 3:23:00 PM	rw-rw-r
logfile.php	4 KB	12/6/2015 12:05:08 PM	rw-rw-r
test.php	2 KB	2/29/2016 12:04:12 PM	rw-rw-r

0 B of 5.09 KB in 0 of 4

SFTP-3 1, 21:06:30

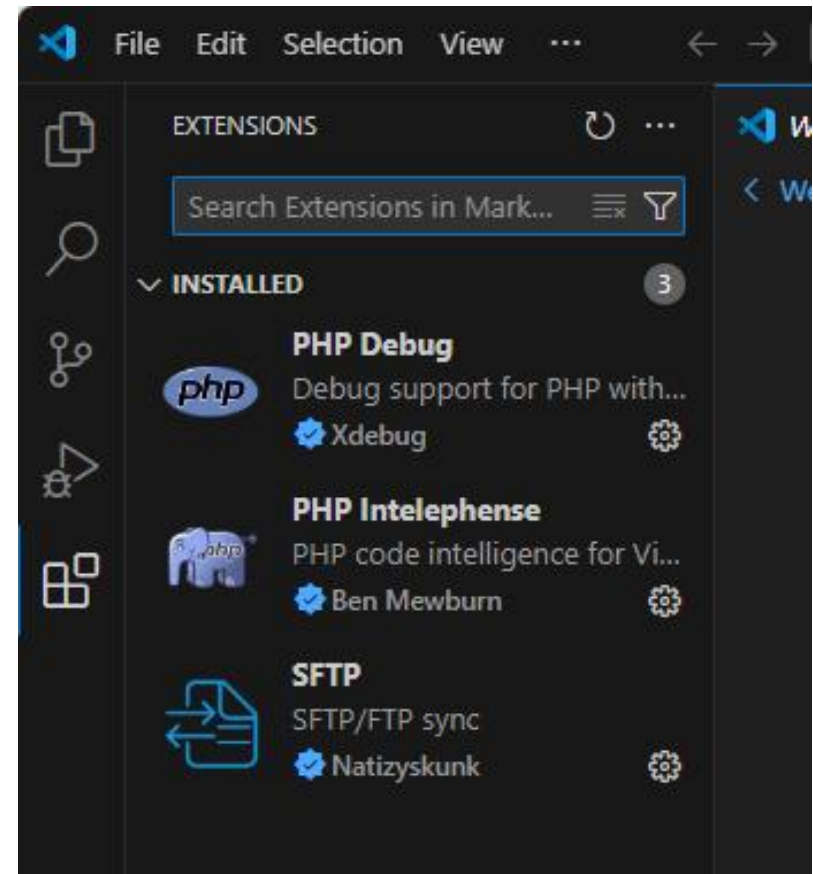
# Visual Studio Code

## ■ 1.87 Portable

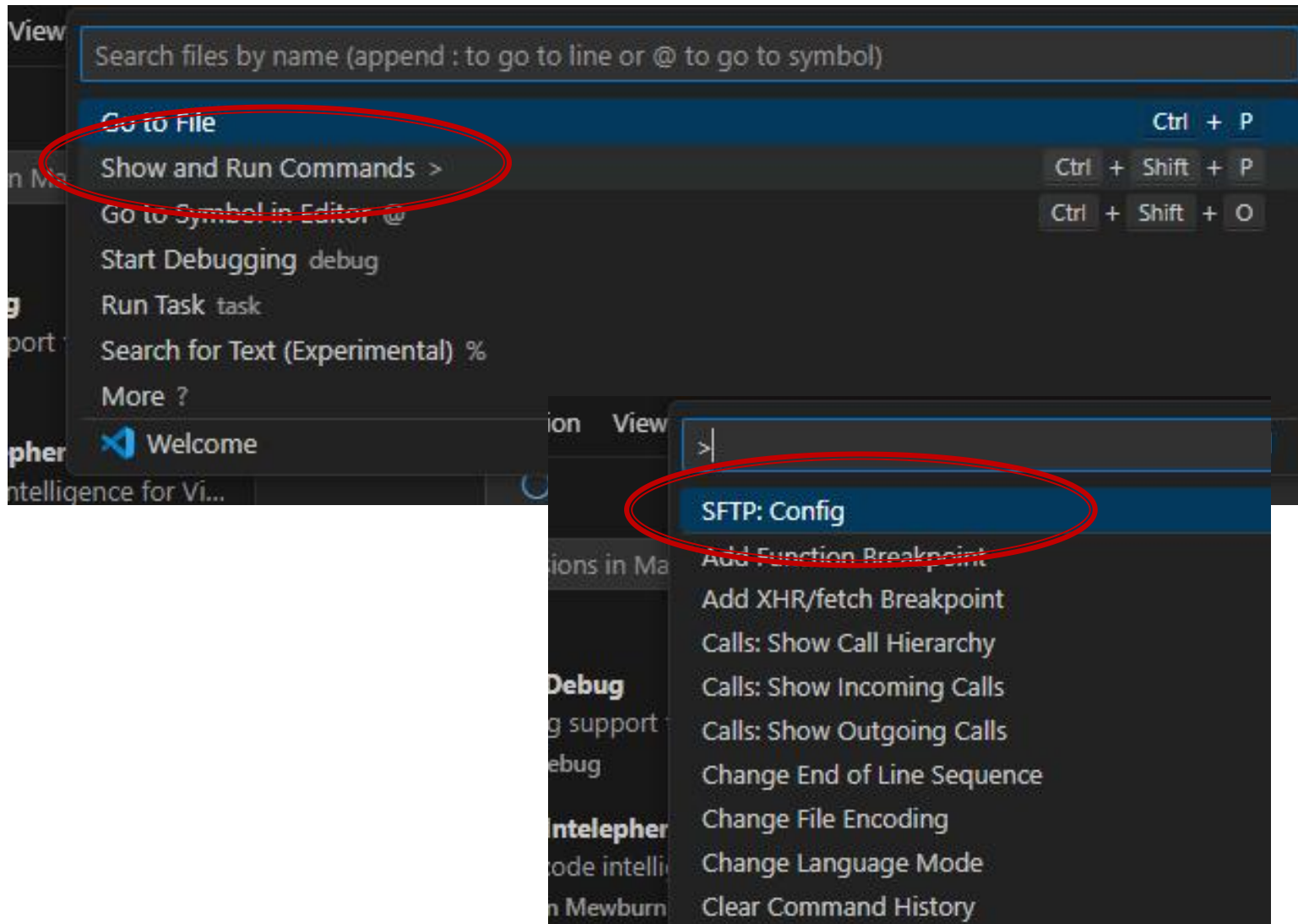


# Visual Studio Code

- Installed extensions
  - PHP Intelephense
    - PHP 8 -> Debian
  - PHP Debug (inactive for now)
  - SFTP – automatic save on server



# Visual Studio Code

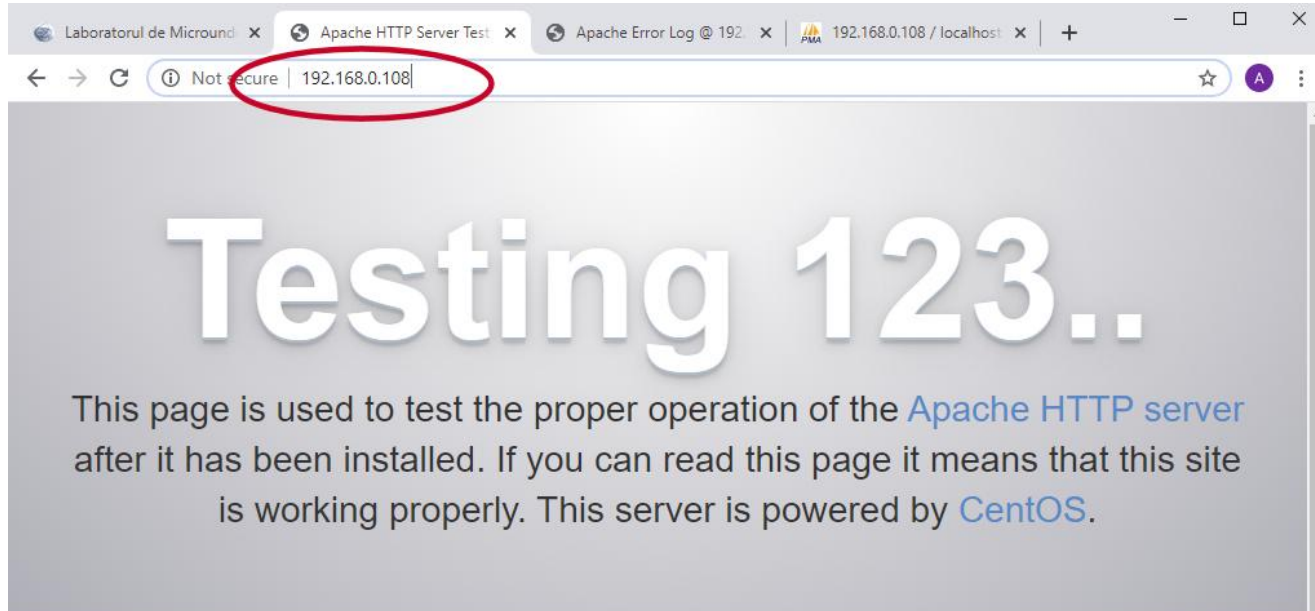


# Visual Studio Code

```
{ } sftp.json x
.vscode > { } sftp.json > ...
1 {
2 "name": "My Server",
3 "host": "localhost",
4 "protocol": "sftp",
5 "port": 22,
6 "username": "username",
7 "remotePath": "/",
8 "uploadOnSave": false,
9 "useTempFile": false,
10 "openSsh": false
11 }
12
```

```
{ } sftp.json •
.vscode > { } sftp.json > ...
1 {
2 "name": "Debian Server",
3 "host": "192.168.30.5",
4 "protocol": "sftp",
5 "port": 22,
6 "username": "student",
7 "remotePath": "/var/www/html/",
8 "uploadOnSave": true,
9 "useTempFile": false,
10 "openSsh": false
11 }
12
```

# Browser



## Just visiting?

The website you just visited is either experiencing problems or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting `www.example.com`, you should send e-mail to `"webmaster@example.com"`.

## Are you the Administrator?

You should add your website content to the directory `/var/www/html/`.

To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

## Promoting Apache and CentOS

You are free to use the images below on Apache and CentOS Linux powered HTTP servers. Thanks for using Apache and CentOS!



# Server MySQL/MariaDB

The screenshot displays the phpMyAdmin web interface in a browser window. The address bar shows the URL `192.168.0.108/phpmyadmin/...PMAURL-5:index.php?db=&table=&server=1&target=&token=f7dda12d42a1...`, with the domain part circled in red. The interface is organized into several sections:

- General Settings:** Includes options to change the password and set the server connection collation to `utf8mb4_unicode_ci`.
- Appearance Settings:** Allows setting the language to `English`, the theme to `pmahomme`, and the font size to `82%`. A `More settings` link is also present.
- Database server:** Lists server details: Localhost via UNIX socket, MariaDB server type, version 5.5.44-MariaDB, protocol version 10, user `root@localhost`, and UTF-8 Unicode (utf8) charset.
- Web server:** Lists web server details: Apache/2.4.6 (CentOS) with OpenSSL/1.0.1e-fips, mod\_fcgid/2.3.9, PHP/5.4.16, mod\_python/3.5.0, Python/2.7.5, database client version libmysql - 5.5.44-MariaDB, PHP extension mysql, and PHP version 5.4.16.
- phpMyAdmin:** Provides links for version information (4.4.15.1), documentation, wiki, official homepage, contribute, get support, and list of changes.

The left sidebar shows a tree view of databases: `New`, `information_schema`, `mysql`, `performance_schema`, `tmpaw`, and `world`.

# Contact

- Laboratorul de microunde si optoelectronica
- <https://rf-opto.etti.tuiasi.ro>
- [rdamian@etti.tuiasi.ro](mailto:rdamian@etti.tuiasi.ro)