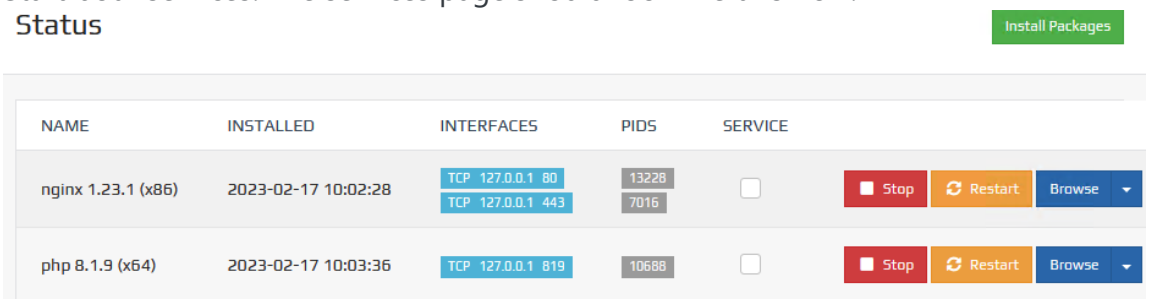
Lab I: Basic Usage of PHP

PART 1: Configure and test your local web server

Part1:

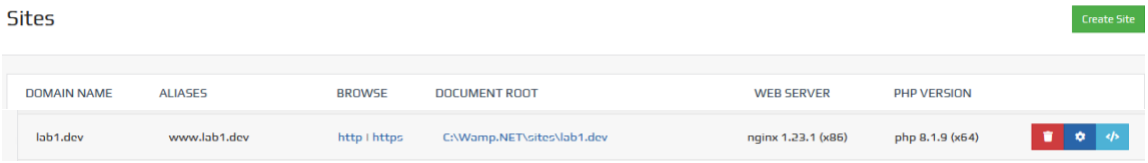
Alternative A: Wamp.Net on Windows

1. Download and install the latest version of Wamp.Net (https://wamp.net)
2. Launch Wamp.Net management panel
3. Using "Install Packages", add PHP 8.1.9 and Nginx 1.23.1. Leave the default parameter values of the said services, **except**, set PHP Time Zone to Europe/Riga
4. Start both services. The services page should look like this now:



5. Go to "Sites section" of the management panel.
6. Create a new site for Lab1 with the following properties:
   a Domain name: lab1.dev
   b Domain aliases www.lab1.dev
   c Web server: nginx 1.23.1 (x86)
   d PHP version: php 8.1.9(x64)

The sites collection should look like this



7. C:\Wamp.NET\sites\lab1.dev is now your *web root*.
8. In Visual Studio Code, open your web root folder. Create a file 'test.php' in the root folder. Set its contents to this:

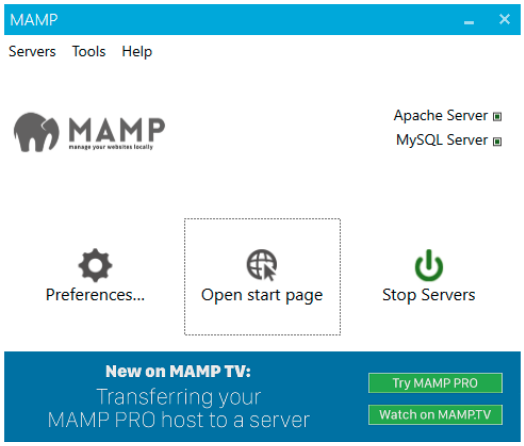
```
<?php echo "Hello from php"; ?>
```

9. Using your web browser, visit the address https://lab1.dev/test.php and make sure you see the text "Hello from php" only.

Part1:

Alternative B: MAMP on Windows

1. Download and install the latest version of MAMP for Windows (https://www.mamp.info/en/downloads/). Choose c:\MAMP as installation directory.
2. Start MAMP management application
3. Check the Preferences according to docs: https://documentation.mamp.info/en/MAMP-Windows/Preferences/index.html
   a. Apache and Nginx port should be 80
   b. Pick PHP version 8.1 or newer
   c. Choose Nginx web server for this lab
   d. Set web server root to c:\MAMP\htdocs
4. Check the status of "Start servers/stop servers" button; it should say "Stop Servers" when it's running



5. Open the web root directory in your computer: c:\MAMP\htdocs.
6. Create a sub-directory "lab1"; you will use it in this lab assignment; Keep open this directory during the lab.
7. In the directory "lab1", create a PHP file named "test.php"
8. In Visual Studio code, edit the file "test.php"

```
<?php
echo "Hello from PHP";
?>
```

9. Using your web browser, visit the address https://localhost/lab1/test.php and make sure you see the text "Hello from php" only.

Part1:

Alternative C: Ubuntu Linux 22.04 native or in Windows Subsystem for Linux

1. Install apache webserver, following the guidelines at https://www.digitalocean.com/community/tutorials/how-to-install-nginx-on-ubuntu-22-04

2. Follow guidelines at https://www.digitalocean.com/community/tutorials/how-to-install-php-8-1-and-set-up-a-local-development-environment-on-ubuntu-22-04 to install PHP 8.1, which will be useful in next lab assignments.

3. Your web root is /var/www/html by default

4. Create a sub-folder “lab1” in it and store all the files in this lab assignment in that folder.

5. In Visual Studio code, edit the file "test.php"

```
<?php
echo "Hello from PHP";
?>
```

6. Using your web browser, visit the address https://localhost/lab1/test.php and make sure you see the text “Hello from php” only.

Lab I: Basic Usage of PHP

PART 2: PHP basics

1. Using Visual Studio Code, create a file "part2.php" un your lab1 folder (the same location where you just created "test.php")
2. Within that file, do the following:
3. Change the content type of your script output to text/plain (using header function)
4. Using loop constructs, make the script print out the following output on screen (you might want to have an array that contains Latvian letters in correct order), i.e., ["a", "ā", "b", "c"…"ģ"];

```
Part 2, subtask1
a
aā
aāb
aābc
aābcč
aābcčd
aābcčde
aābcčdef
aābcčdefg
aābcčdefgģ
```

3. Write a PHP function that takes a natural number, say *year*, as the input and then returns the date of first Monday of that year (for instance, this year, it was January 2nd)

   The function is expected to return a Unix timestamp (e.g. what is returned from mktime() function).

   In the main body of the script after the function definition, create some tests that run your function, E.g., pick *year* as 2022, and then print the result on the screen (use date() for formatting the output).

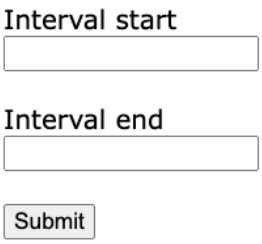
4. Use a loop to print function results for multiple years:

   The first Monday of year 2023 is 2nd January.
   The first Monday of year 2030 is 7th January.
   The first Monday of year 2016 is 4th January.
   The first Monday of year 2017 is 2nd January.

Lab I: Basic Usage of PHP

PART 3: Forms

1. Create a new HTML file under the directory "lab1", name the file "part3_input.html"
2. Create the following form in your HTML page. You can use POST or GET method.



3. When clicking submit button, both values should be submitted to a PHP file, say "part3_result.php".
4. Create the PHP file "part3_result.php". Within the script, read both numbers, and parse them as integers.
5. If the data is wrong (not integer, start > end, negative numbers…) output an error message and stop execution of script.
6. Otherwise, think of the two numbers as year numbers. For the whole year range, print a list of first Mondays of the year.
7. Test your program with different inputs.