


# Dynamic site

## PHP – variables – concatenation (merging of character strings)

```
$output = "Ein kurzer Text";
```

```
$anotherOutput = "noch ein kurzer Text";
```



```
echo $output." zusammen mit ".$anotherOutput;
```



# Dynamic site

## PHP – client display

Ein kurzer Text zusammen mit noch ein kurzer Text



# Dynamic site

## PHP - Variables - Integer

```
/* Integer */  
echo "<br><br>";  
echo "/* Integer */";  
$output = 1;  
echo "<br>";  
echo $output;
```

\$output

=

1

;

=> Label (for a variable)

=> variable designation

=> assignment

=> Integers

=> Ends the statement



# Dynamic site

## PHP – client display

```
/* Integer */  
1
```




# Dynamic site

## PHP - Variables - Concatenation (merging of character string and number)

```
$anotherOutput = "noch ein kurzer Text";
```

```
$output = 1;
```

```
/* String + Integer */  
echo "<br><br>";  
echo "/* String + Integer */";  
echo "<br>";  
echo $output." zusammen mit ".$anotherOutput;
```



# Dynamic site

## PHP – client display

```
/* String + Integer */  
1 zusammen mit noch ein kurzer Text
```



# Dynamic site

## PHP – Variables – Integers - Calculations

```
/*Rechnung*/
echo "<br><br>";
echo "/*Rechnung*/";
echo "<br>";
$four = 4;
$six = 6;
$plus = "+";
$equal = "=";
```

```
$result = $four + $six;
echo $four.$plus.$six.$equal.$result;
```

Arithmetic operators:

+	=	addition
-	=	subtraction
*	=	multiplication
/	=	division

**\$result**

=

+

=> result variable

=> assignment

=> Addition operator



# Dynamic site

## PHP – Variables – Integers - Calculations

```
/*Rechnung*/  
echo "<br><br>";  
echo "/*Rechnung*/";  
echo "<br>";  
$four = 4;  
$six = 6;  
$plus = "+";  
$equal = "=";  
  
$result = $four + $six;  
echo $four.$plus.$six.$equal.$result;
```



Output of the invoice including the result





# Dynamic site

## PHP – client display

```
/*Rechnung*/  
4+6=10
```



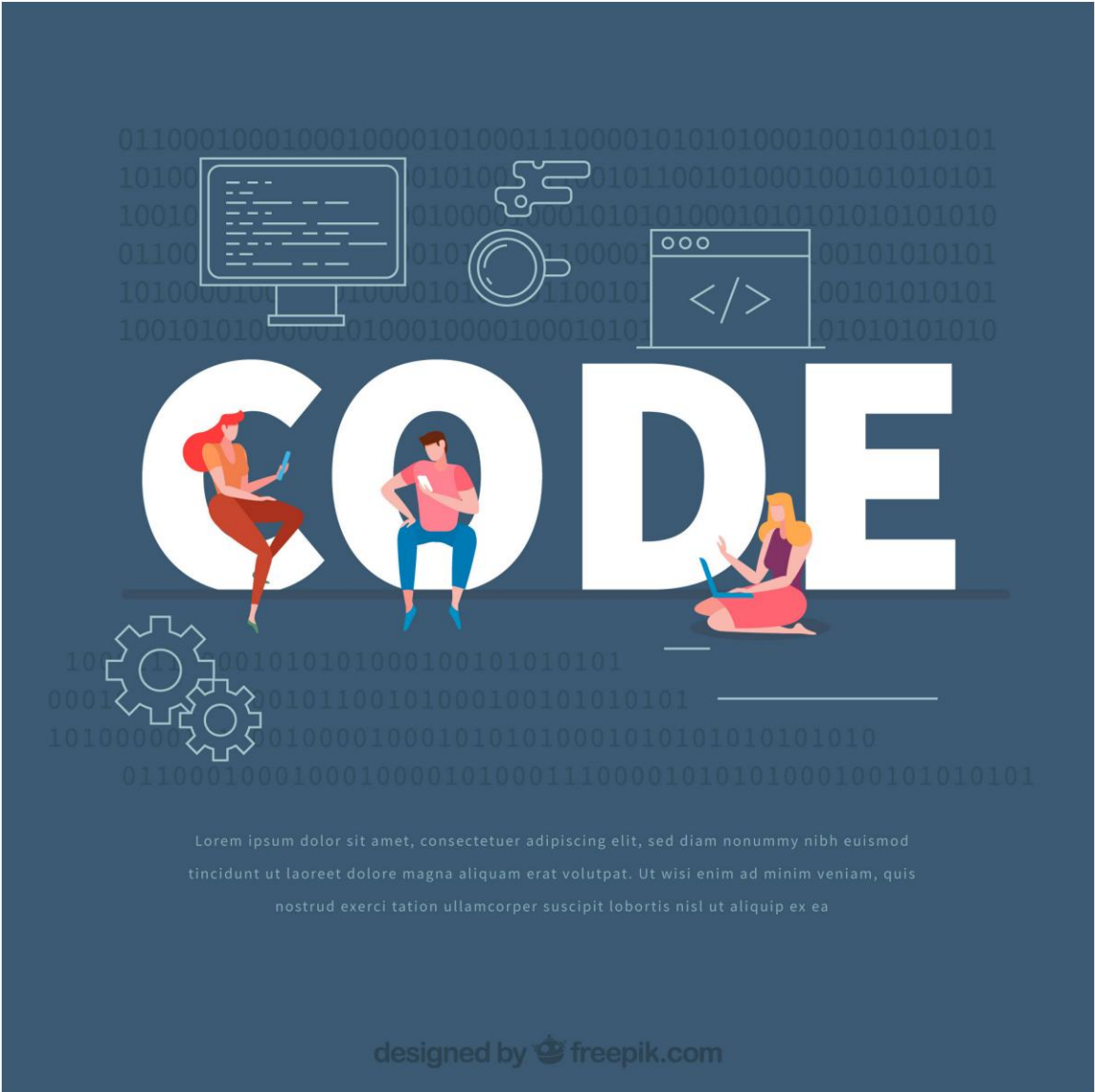
# Dynamic Websites – Why PHP?

## good documentation



- Good documentation
  - Entry ([w3schools, php.net](https://www.w3schools.com/php/))
- For countless problems already exist countless solutions
- [Blogs](#) / [StackOverflow](#) / [Reddit](#) Etc.





# Dynamic site

## PHP - HTML form

Requirement:

Create a form to input numbers that add together and return a result on the same page

`/* Eingabe Felder zum Addieren */`

addieren



`/* Eingabe Felder zum Addieren */`

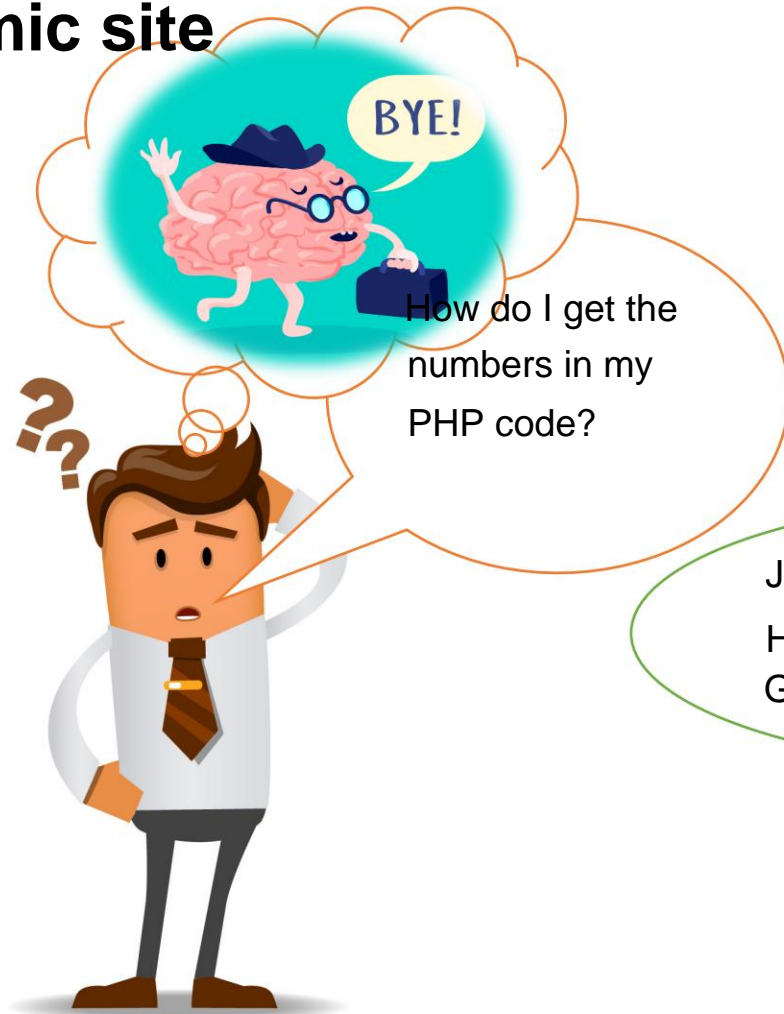
addieren

Das Ergebnis lautet:

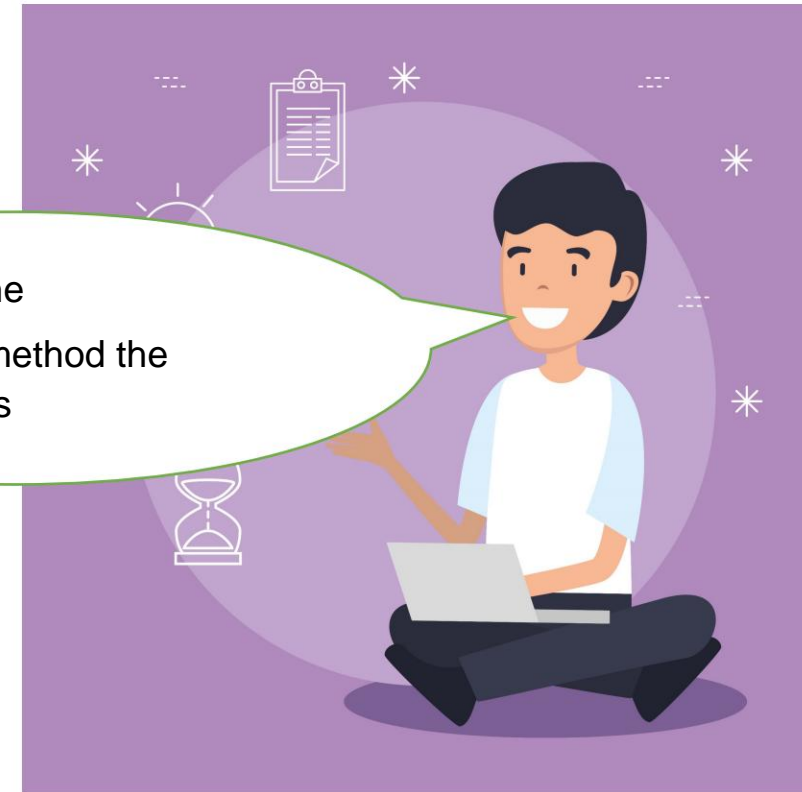
4



# Dynamic site



Just use from the HTTP request method the GET parameters



# Dynamic site

## PHP – HTML form design

```
/* Eingabe Felder zum Addieren */  
<br><br>  
<form action="" method="GET" id="form">  
    <input type="text" name="number1" value=""/>  
    <br><br>  
    <input type="text" name="number2" value=""/>  
    <br><br>  
    <input type="submit" value="addieren" style="background:lightgrey;width:100px;height:30px" />  
    <br><br>  
</form>
```


/\* Eingabe Felder zum Addieren \*/


# Dynamic site

## PHP - HTML Form - Redirect

Without forwarding (just left empty)



```
/* Eingabe Felder zum Addieren */  
<br><br>  
<form action="" method="GET" id="form">  
    <input type="text" name="number1" value=""/>  
    <br><br>  
    <input type="text" name="number2" value=""/>  
    <br><br>  
    <input type="submit" value="addieren" style="background:lightgrey;width:100px;height:30px" />  
    <br><br>  
</form>
```




# Dynamic site

## PHP - HTML form - Hyper Text Transfer Protocol

### Set HTTP Request Method

Here **GET**



```
/* Eingabe Felder zum Addieren */  
<br><br>  
<form action="" method="GET" id="form">  
    <input type="text" name="number1" value=""/>  
    <br><br>  
    <input type="text" name="number2" value=""/>  
    <br><br>  
    <input type="submit" value="addieren" style="background:lightgrey;width:100px;height:30px" />  
    <br><br>  
</form>
```





# Dynamic site

## PHP - HTML form - HTTP request method GET

Determine **GET** parameters

And what good is that now?



```
/* Eingabe Felder zum Addieren */  
<br><br>  
<form action="" method="GET">  
  <input type="text" name="number1" value=""/>  
  <br><br>  
  <input type="text" name="number2" value=""/>  
  <br><br>  
  <input type="submit" value="addieren" style="background:lightgrey;width:100px;height:30px" />  
  <br><br>  
</form>
```

# Dynamic site

## PHP - HTML form - HTTP request method GET

/\* Eingabe Felder zum Addieren \*/

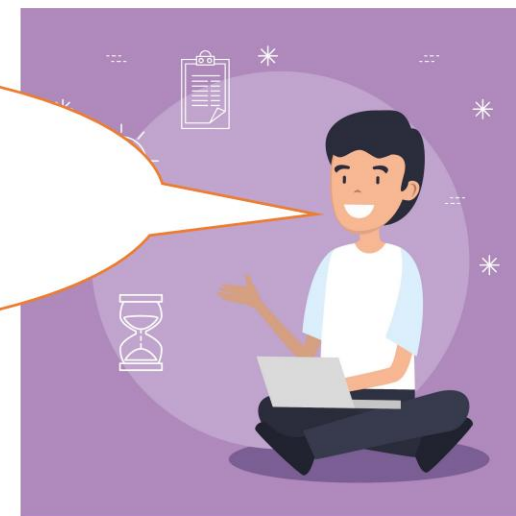
addieren

/\* Eingabe Felder zum Addieren \*/

addieren

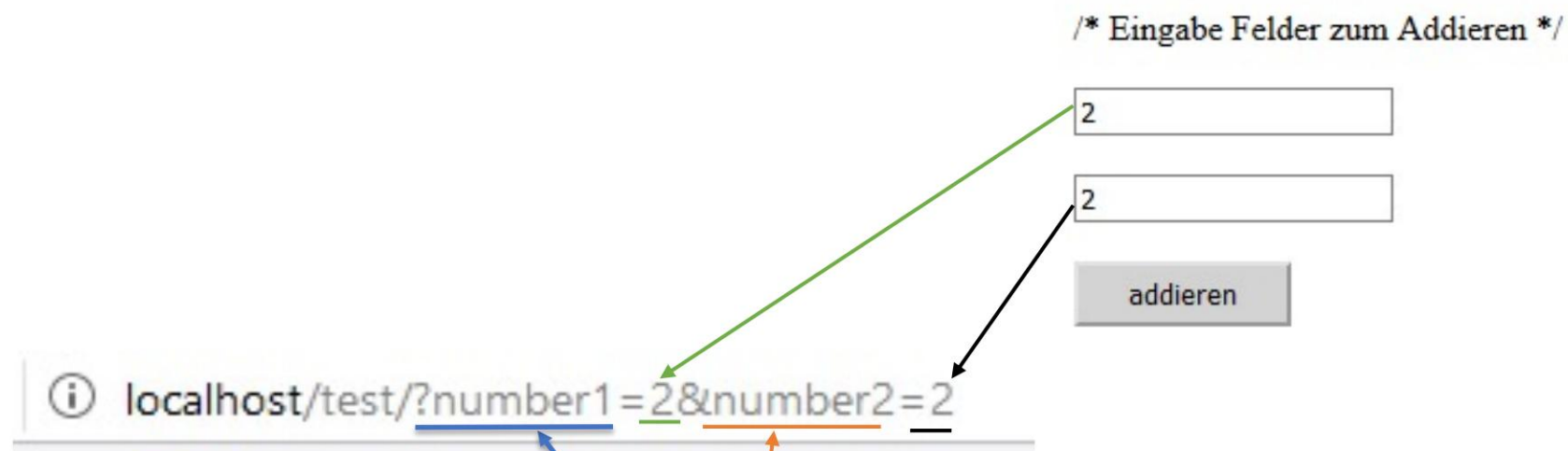
Press "add" and look at  
your browser's address bar

GET parameters



# Dynamic site

## PHP - HTML form - HTTP request method GET



```
/* Eingabe Felder zum Addieren */  
<br><br>  
<form action="" method="GET" id="form">  
  <input type="text" name="number1" value=""/>  
  <br><br>  
  <input type="text" name="number2" value=""/>  
  <br><br>  
  <input type="submit" value="addieren" style="background:lightgrey;width:100px;height:30px" />  
  <br><br>  
</form>
```



# Dynamic site

## PHP - HTML form - HTTP request method GET

Parameters are appended to the URL with ?

localhost/test/?number1=2&number2=2

Parameters are connected with &

What do you do with it now?



# Dynamic site

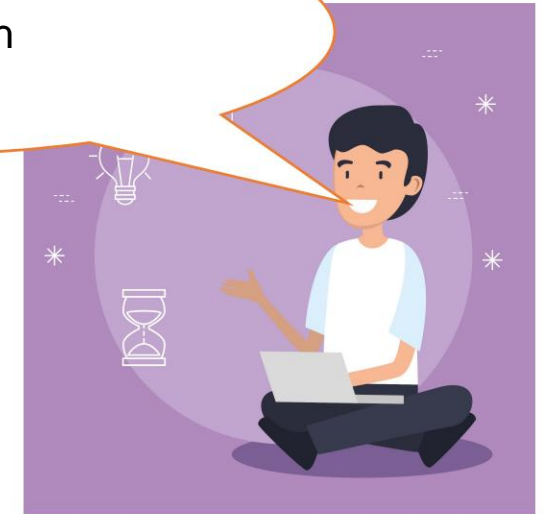
## PHP – HTTP request method GET

PHP can read and reuse the GET parameters with `$_GET['variable']`.

① localhost/test/?number1=2&number2=2

```
$number1 = $_GET['number1'];  
$number2 = $_GET['number2'];
```

```
/* Eingabe Felder zum Addieren */  
<br><br>  
<form action="" method="GET" id="form">  
  <input type="text" name="number1" value=""/>  
  <br><br>  
  <input type="text" name="number2" value=""/>  
  <br><br>  
  <input type="submit" value="addieren" style="background:lightgrey;width:100px;height:30px" />  
  <br><br>  
</form>
```



# Dynamic site

## PHP – HTTP request method GET

A message is displayed on the website when the page is accessed



```
$number1 = $_GET['number1'];  
$number2 = $_GET['number2'];
```

/\* Eingabe Felder zum Addieren \*/

addieren

**Notice:** Undefined index: number1 in C:\xampp\htdocs\test\content\vars.php on line 67

**Notice:** Undefined index: number2 in C:\xampp\htdocs\test\content\vars.php on line 68





# Dynamic site

## PHP – HTTP request method GET

Request parameters must  
be assigned to a  
Variables are checked (so that  
errors do not occur)



```
isset($_GET['number1'])
```

`isset` => checks if a variable exists and if it is not NULL  
=> returns a boolean =>  
[more on that](#)



# Dynamic site

## PHP – HTTP request method GET

```
if(isset($_GET['number1']) && isset($_GET['number2'])) {
```

Condition\*

Check if a  
value was  
set

request parameters

linking  
several  
Conditions\*

Check if a  
value was  
set

request parameters

\* More details on **if's** will come in a later lesson





# Dynamic site

## PHP – PHP + Invoice

```
<?php

if(isset ($_GET['number1']) && isset ($_GET['number2'])) {
    $number1 = $_GET['number1'];
    $number2 = $_GET['number2'];

    $result = $number1 + $number2;

    echo "Das Ergebnis lautet:";
    echo "<br>".$result;
}

?>
```



# Dynamic site

## PHP – HTML form + PHP

```
/* Eingabe Felder zum Addieren */  
<br><br>  
<form action="" method="GET" id="form">  
  <input type="text" name="number1" value=""/>  
  <br><br>  
  <input type="text" name="number2" value=""/>  
  <br><br>  
  
  <input type="submit" value="addieren" style="background:lightgrey;width:100px;height:30px" />  
  <br><br>  
</form>  
<?php  
  
if(isset ($_GET['number1']) && isset ($_GET['number2'])) {  
  $number1 = $_GET['number1'];  
  $number2 = $_GET['number2'];  
  
  $result = $number1 + $number2;  
  
  echo "Das Ergebnis lautet:";  
  echo "<br>".$result;  
}  
  
?>
```



designed by  freepik.com



# Dynamic Websites - What can PHP do?



# Dynamic Websites - What can PHP do?

## command line programming

```
File Edit View Search Terminal Help
SERVER["SCRIPT_FILENAME"] => /var/www/html/infophp.php
SERVER["PATH_TRANSLATED"] => /var/www/html/infophp.php
SERVER["DOCUMENT_ROOT"] =>
SERVER["REQUEST_TIME_FLOAT"] => 1436341155.45
SERVER["REQUEST_TIME"] => 1436341155
SERVER["argv"] => Array
(
    [0] => /var/www/html/infophp.php
)
SERVER["argc"] => 1

PHP License
This program is free software; you can redistribute it and/or modify
it under the terms of the PHP License as published by the PHP Group
and included in the distribution in the file: LICENSE

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

If you did not receive a copy of the PHP license, or have any
questions about PHP licensing, please contact license@php.net.
avi@deb:~$
```

- No server/browser • Only PHP parser required

- Exemplary use • Cron jobs\* • Simple text processing

(tokenization, Part of speech tagging)

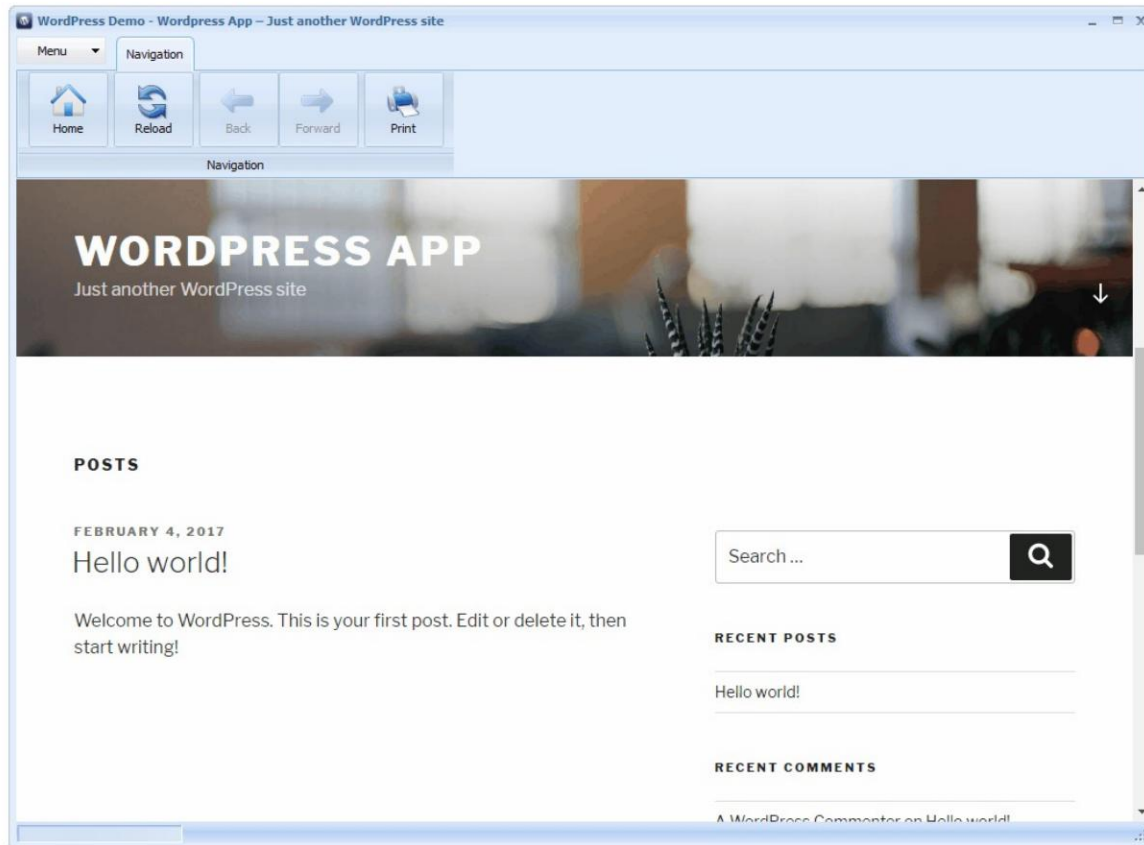
\*Certain tasks to be defined  
Appointments are executed automatically



Source: <https://www.tecmint.com/run-php-codes-from-linux-commandline/>

# Dynamic Websites - What can PHP do?

## Desktop – applications (rich client)



- Cross-platform
- More advanced
- (not the best alternative for desktop applications)

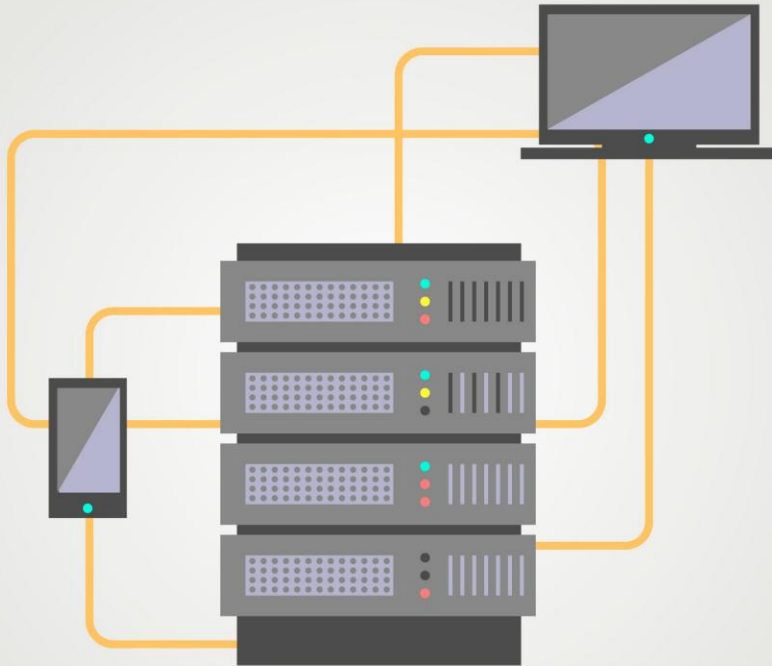
[\(http://gtk.php.net/\)](http://gtk.php.net/)

Source: <https://www.exeoutput.com/samples/wordpress-desktop-app>



# Dynamic Websites - What can PHP do?

## Server side programming



designed by  freepik

- Represents the usage focus of PHP
- You will need: 1.  
PHP parser (usually built-in) 2. a web  
server  
3. Web browser

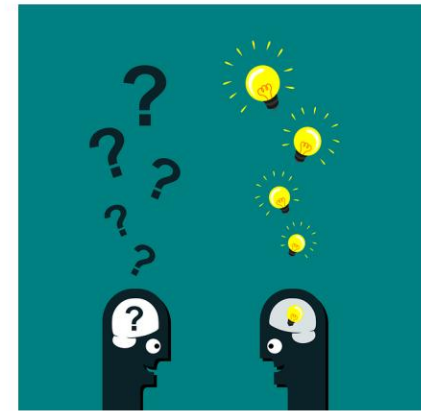


# Dynamic site

## What can PHP do in detail?

- Procedural programming (e.g. imperative approach)
- Object oriented programming
- Communication with various services
- Connection of many different databases possible • databases
- Opening of network sockets (->clients)
- Dynamic generation of content (e.g. HTML, Pdf files, images etc.)
- Etc.

Which in turn use various protocols  
(LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM)



finish

# Dynamic websites

## PHP I

