Co-funded by the
Erasmus+ Programme
of the European Union

List of Task sheets from Module D: Web Development

#	Handout	#	Task sheet
D1	HTML/CSS/ JavaScript	**D1.1**	Create your first webpage
		D1.2	Create your first webpage using a visual block editor
		D1.3	Find the errors
		D1.4	Create website for a specific target group
		D1.5	Javascript - Make your webpage interactive with javascript
		D1.6	Javascript - Create your first interactive e-commerce
		D1.7	Create a working contact form
D2	Wordpress	**D2.1**	Connect wordpress on wamp server
		D2.2	Create pages and posts on wordpress
		D2.3	Edit PHP files

Co-funded by the
Erasmus+ Programme
of the European Union

Module: Web development

Topic: HTML/CSS

Task sheet D1.1: Create your first webpage

Time: 2:00 hour(s)

General description:

Students will create a web page of choice using a visual editor and remake the same page with HTML and CSS code. The students will learn the basics of HTML and CSS code and will create their first simple web page using these both languages.

Learning objective(s):

- Understanding how a web page is developed using, basically, HTML and CSS code.
- Learn the use of the basic HTML tags, like , <p>, , etc.
- Learn the use of basic CSS rules, and how to apply them to change the style of the web page.
- Create a personal webpage using both HTML and CSS code.
- Acquire experience in order to be able to read and understand HTML and CSS code.

Material required:

- Computer or laptop
- Internet connection
- online website builder, for example Google Sites (visual)
- online text editor, for example w3.school

Description of the activity:

1. To start this lesson the trainer will make the students think of a personal webpage that they can create, for example, it can be a personal webpage with their CV, or a webpage with some kind of interest for them, like a recipes webpage.
2. Once they have decided on a theme or topic, the students can create the webpage they picked using some online instrument like Google Sites. Give them around 30 minutes to build a web page with certain requirements. For example, images, titles in different sizes, different colours, and other things you would like the students to learn in HTML or CSS code. Mention that it is not important that the website is finished or perfect but should contain all the necessary requirements within the given timeframe.
3. Once they have finished, give a short introduction to web programming and to HTML and CSS languages, showing them the possibility to create the same webpage but

using these languages instead. Introduce W3Schools as an online editor and guideline.

4. Ask students to recreate the same webpage they have made before but this time creating it with HTML tags and CSS rules, using an online editor or a local editor installed on their computer. Students can find all the rules and tags in the different sections of the W3Schools web page and apply them directly through the online editor.
5. When finished, ask the students to present their websites to the group. This is a good moment for the trainer to understand the level of the students (for the next session) and which elements of the coding languages are not yet mastered by the students. It can also be helpful to the students, giving new ideas of the possibilities of web development.

How to adapt to different learners:

If some students already know how to write HTML and CSS code, code challenges can be assigned to them. You can write the challenges in different cards, to give to a group, individual or pair after finishing the activity, and after each finished challenge. For example:

- Use CSS to apply an image as a background to a element.
- Use CSS to change a property to the same tag through the whole webpage
- Restructure the CSS code in order to apply it on the head of the document and not inline, inside the HTML tag
- Add “Classes” or “IDs” in order to apply specific styling to specific elements

Additional information:

- HTML reference guide
- W3Schools - Guide for every HTML element and CSS rule, and examples for each one of them
- Khan Academy: useful resources and videos on coding HTML & CSS, in many different languages

Co-funded by the
Erasmus+ Programme
of the European Union

Module: Web development

Topic: HTML/CSS

Task sheet D1.2: Create your first web page using a visual block editor

Time: 3:00 to 4:00 hour(s)

General description:

Participants will create a web page of choice using a platform that simplifies the process thanks to the use of blocks, being able to use HTML and CSS code. The participants will learn the basics of HTML and CSS code and will create their first simple web page using both these languages, helping the participants that are starting with these languages through the use of already defined 'blocks' and a visual interface.

Learning objective(s):

- Understanding how a web page is developed using, basically, HTML and CSS code.
- Learn the use of the basic HTML tags, like <p>, <img>, <a>, etc.
- Learn the use of basic CSS rules, and how to apply them to change the style of the web page.
- Create a personal webpage using both HTML and CSS code.
- Acquire experience in order to be able to read and understand HTML and CSS code.

Material required:

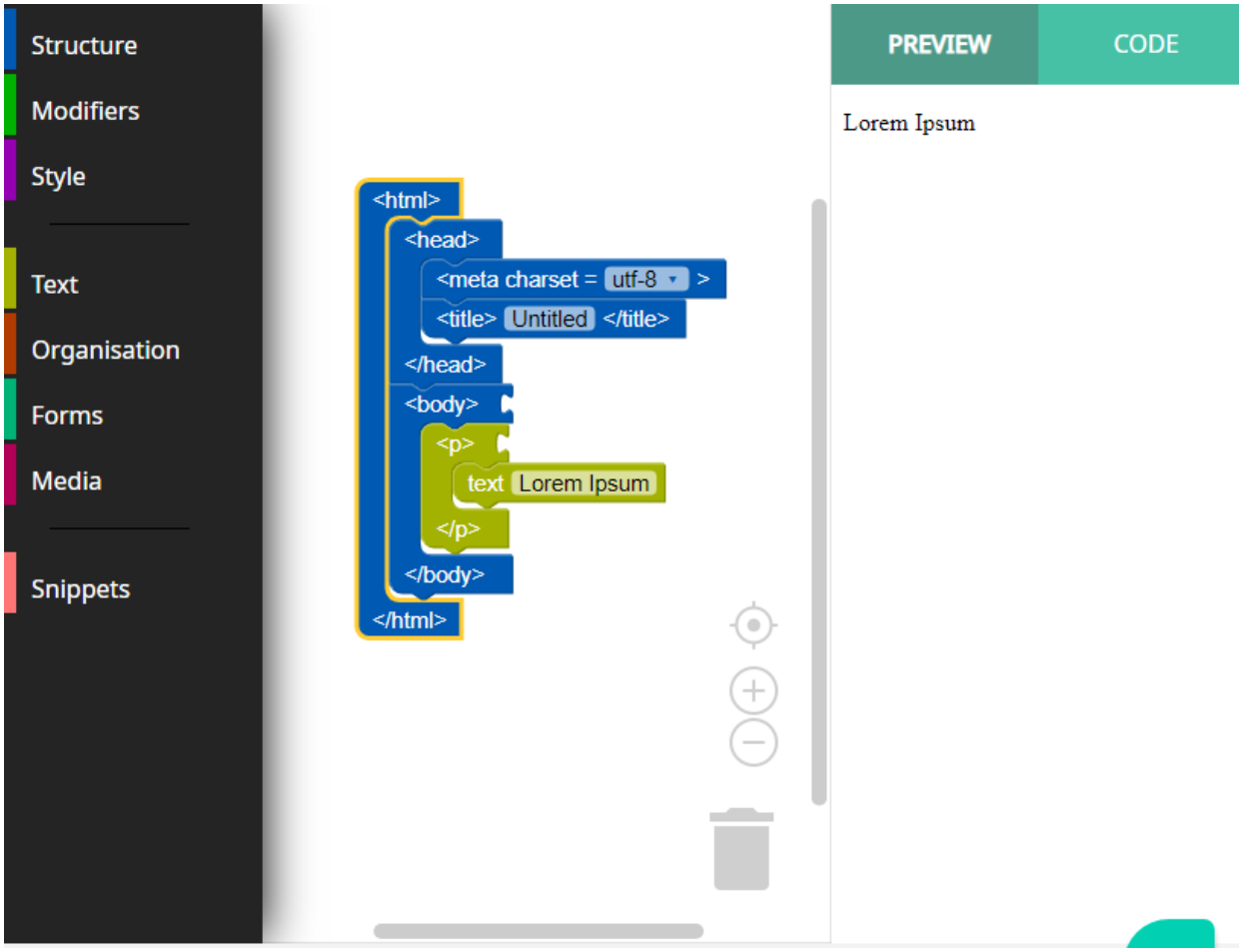
- Computer or laptop
- Internet connection
- online website builder, for example Codedragon.org

Description of the activity:

1. To start this lesson the trainer will make the participants think of a personal webpage that they can create, for example, it can be a personal webpage with their CV, or a webpage with some kind of interest for them, like a recipe's webpage. If the participants are of an intermediate level of digital skills, you can use Google Sites to create a simple web page. Just make sure to include at least: titles, text, image, button, link, and any other element you want participants to learn. If the participants are not able to think of a web page example on their own the trainer can show them an example.
2. After this, the trainer will introduce the platform Codedragon.org, will guide them on creating an account on this platform and will show them how to start their own project. After the account creation the participants will go to https://codedragon.org/create,

introduce a title and a description, and they can start to build their new website by clicking on the “Code” button.

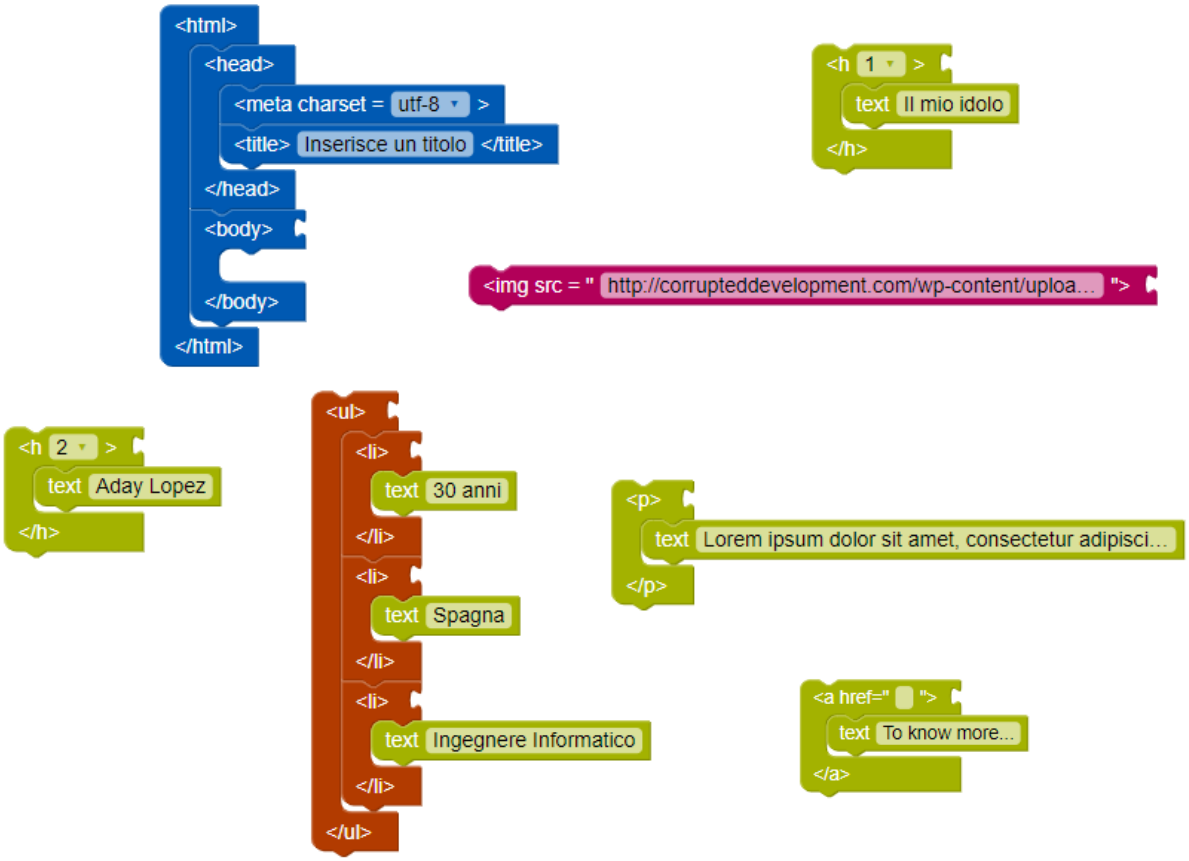
3. On the new interface that will be shown, they can start to add blocks to their website and a preview of their work will be shown on the right side of the screen. It is interesting to advise them to also check the real code they are creating clicking on the code button on the top right corner, in order to start to read and get comfortable with HTML and CSS code:



4. After showing how the platform works, ask the participants to start to work on their personal webpage or to recreate the example that the trainers have given them.

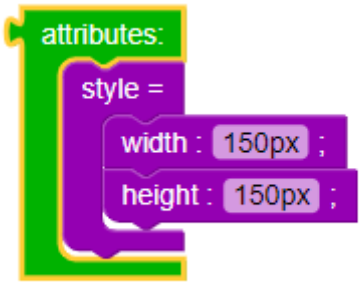
Challenge 1

Create a basic website using the following blocks:



Challenge 2

Try to add a little bit of style (CSS) changing the size of the image. You can use the following blocks:

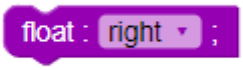


Challenge 3

Change the style of the main title (H1) using the following blocks, in this way you will change the font style, the text alignment, colour, size and background colour:

Challenge 4

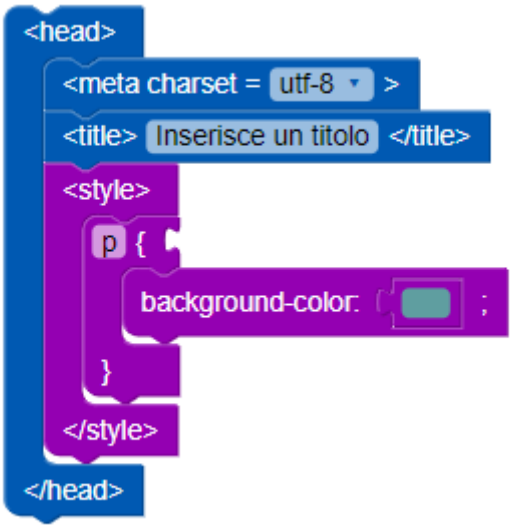
Try to add an image to the right of a text applying the following style to the image. You can try to change the value from ‘right’ to ‘left’ and see what changes:



Challenge 5

Add more than one paragraph to your website and try to change the style of all paragraphs at the same time. In order to do that, you will have to add your style on the ‘head’ part of your code.

To know more, look for “Internal CSS” in the following link:
https://www.w3schools.com/css/css_howto.asp

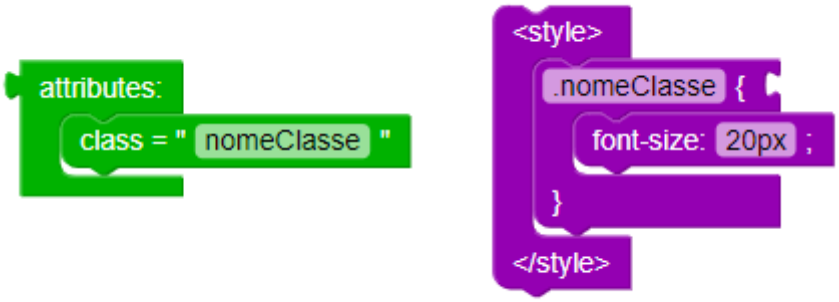


Challenge 6

One you have changed the style of all paragraphs, try just to change the style of one paragraph. You can use an 'ID' or a 'Class' in order to do so:

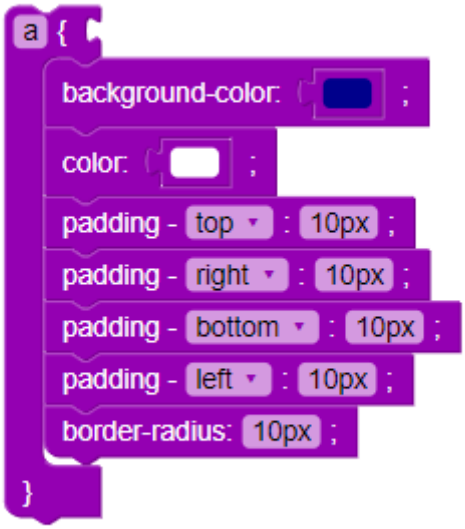
For example: https://www.w3schools.com/html/html_classes.asp

You will have to add a class or an id to your paragraph and use this selector on your style:



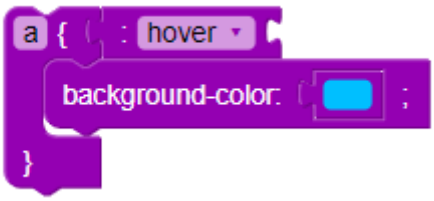
Challenge 7

Now, you can try to change the style of a link to make it more similar to a button. For example, you can add the following style, always in the head section of your code:



Challenge 8

Now you can try to use the selector :hover to change the style of your link just when the mouse is over it. Try to add the following style also in the head part of your code:



Challenge 9

Now we can start to work locally on our computer. Download the code and use a code editor to open it, and use your favourite browser to look the webpage you are creating.

5. When finished, ask the participants to present their websites to the group. This is a good moment for the trainer to understand the level of the participants (for the next session) and which elements of the coding languages are not yet mastered by the participants. It can also be helpful to the participants, giving new ideas of the possibilities of web development.

How to adapt to different learners:

If some participants are new to HTML and CSS code, have low computer writing skills, or low English skills, code challenges can be a solution. You can write the challenges in different cards, to give to a group, individual or pair after finishing the activity, and after each finished challenge. For examples, please check here (link to examples)

You can also split this lesson, and use one lesson to finish all the challenges, and another lesson to replicate an existing webpage of choice or made by the participants with a graphic website creation tool like Google Sites.

Additional information:

- HTML reference guide
- W3Schools - Guide for every HTML element and CSS rule, and examples for each one of them
- Codedragon.org – Platform that allows the creation of web pages using blocks

Co-funded by the
Erasmus+ Programme
of the European Union

Module: Web development

Topic: HTML/CSS

Task Sheet D1.3: Find the errors

Time: 01:30 hour(s)

General description:

The students will be presented with an already created web page including HTML and CSS code. This webpage has a lot of syntax errors that the students will have to solve with the help of online resources and their previous knowledge of CSS and HTML languages.

Learning objective(s):

- Learn how to read an already existing HTML and CSS code.
- Improve the skills analyzing and understanding HTML and CSS code, and understand that is the objective of each rule and tag.
- Improve the skill on debugging, finding where the errors are and learning how to solve them.
- Learn how to consult external material in order to improve the developing skills.

Material required:

- Computer or laptop
- Internet connection
- Text editor (online or offline): Sublime Text/Brackets/W3Schools online editor

Description of the activity:

1. The trainer will give to the students a webpage developed with HTML and CSS and with a lot of syntax errors.
2. The trainer will ask them to solve all the errors they can in a maximum amount of time (suggestion is about 45 minutes). Students will have to find out the errors by reading the code and trying to see what changes when changing the code. They can of course make use of online resources and tutorials when searching to understand the code.
3. (optional) At the end, each participant/group will present the errors that they found. The trainer also shows the page without errors and can go into more depth in errors that were difficult to find.

How to adapt to different learners:

The best solution is to present a webpage with a large amount of errors. The objective of this activity is not to fix all the errors but to improve the debugging or error fixing skills through the practice. The trainer should find a way to help those with less knowledge in order to balance the lesson and add more complex errors for those who are already more advanced.. The errors go from easy syntax errors, i.e. code that is not closed, to more sophisticated errors such as white text on white background.

Additional information:

- HTML reference guide
- W3Schools - Guide for every HTML element and CSS rule, and examples for each one of them
- Example code for the activity:

Code with errors:

```
<!DOCTYPE html>
<head>
  <title>All about Palermo</title>
</head>
<bodi style="margin: 0px;">
  <header stile="text-align: center; background:
url('https://siviaggia.files.wordpress.com/2019/09/palermo-1.jpg'); background-
position: center; background-size: cover; color: white;">
    <img style="width: 200px; margin-top: 30px; border: 7px solid white; border-radius:
20px;" href="https://gdsit.cdn-immedia.net/2015/05/Trinacria.png">
    <h1 style="margin: 0px; font-size: 70px; font-family: fantasy; text-shadow: 4px 4px
black; color: white; background: rgba(0,0,0,0.5); padding-top: 20px;">Palermo
  <header>
  <nav style="text-align= center;">
    <ul style="margin: 0px; font-size: 20px; padding: 10px; background: black">
      <li style="display: inline; padding: 0px 10px 0px 10px;"><a target="_blank"
style="color: black;"
href="https://www.google.com/maps/place/Palermo,+Province+of+Palermo/data=!4m2!3m1!1s0x
1319e8c9814ed099:0xa0b042c233bd880?sa=X&ved=2ahUKEwimqa6YotTlAhXS3KQKHeKNDXAQ8gEwAHoECA
oQAQ">Where</a></li>
      <la style="display: inline; padding: 0px 10px 0px 10px;"><a target="_blank"
style="color: black;"
href="https://en.wikipedia.org/wiki/History_of_Palermo">History</a></li>
    </ul>
  </nav>
  <main>
<center>
  <div style="max-width: 700px; padding: 20px;">
  <h2>Information</h2>
  <paragraph>The population of Palermo urban area is estimated by Eurostat to be
855,285, while its metropolitan area is the fifth most populated in Italy with around
1.2 million people. In the central area, the city has a population of around 676,000
people. The inhabitants are known as Palermitani or, poetically, panormiti. The
languages spoken by its inhabitants are the Italian language and the Palermitano
dialect of the Sicilian language. </p>
  </div>
  <div style="max-width: 700px; padding: 20px;">
  <h2>Cannoli</h2>
  <img style="width: 200px; float: left; margin:
10px;"src="https://www.giallozafferano.it/images/ricette/206/20605/foto_hd/hd650x433_wm
```

```
.jpg">
  <p>Cannoli (Italian pronunciation: [kanËˆnÉ”Ëli]; Sicilian: cannula) are Italian
pastries that originated on the island of Sicily and are today a staple of Sicilian
cuisine[1][2] as well as Italian-American cuisine. Cannoli consist of tube-shaped
shells of fried pastry dough, filled with a sweet, creamy filling usually containing
ricotta. They range in size from "cannulicchi", no bigger than a finger, to the fist-
sized proportions typically found south of Palermo, Sicily, in Piana degli Albanesi.[2]
In the mainland Italy they are commonly known as cannoli siciliani (Sicilian cannoli).
</p>
  <////div>
</center>
<main>
<footer style="background-color: black; text-align: center; color: white; font-size:
18px; padding: 20px;">
  Enjoy Palermo!</footer>
</bodi>
```

Code without errors:

```
<!DOCTYPE html>
<head>
  <title>All about Palermo</title>
</head>
<body style="margin: 0px;">
  <header style="text-align: center; background:
url('https://siviaggia.files.wordpress.com/2019/09/palermo-1.jpg'); background-
position: center; background-size: cover; color: white;">
    <img style="width: 200px; margin-top: 30px; border: 7px solid white; border-radius:
20px;" src="https://gdsit.cdn-immedia.net/2015/05/Trinacria.png">
    <h1 style="margin: 0px; font-size: 70px; font-family: fantasy; text-shadow: 4px 4px
black; color: white; background: rgba(0,0,0,0.5); padding-top: 20px;">Palermo</h1>
  </header>
  <nav style="text-align: center;">
    <ul style="margin: 0px; font-size: 20px; padding: 10px; background: black">
      <li style="display: inline; padding: 0px 10px 0px 10px;"><a target="_blank"
style="color: white;"
href="https://www.google.com/maps/place/Palermo,+Province+of+Palermo/data=!4m2!3m1!1s0x
1319e8c9814ed099:0xa0b042c233bd880?sa=X&ved=2ahUKEwimqa6YotTlAhXS3KQKHeKNDXAQ8gEwAHoECA
oQAQ">Where</a></li>
      <li style="display: inline; padding: 0px 10px 0px 10px;"><a target="_blank"
style="color: white;"
href="https://en.wikipedia.org/wiki/History_of_Palermo">History</a></li>
    </ul>
  </nav>
  <main>
<center>
  <div style="max-width: 700px; padding: 20px;">
  <h2>Information</h2>
  <p>The population of Palermo urban area is estimated by Eurostat to be 855,285, while
its metropolitan area is the fifth most populated in Italy with around 1.2 million
people. In the central area, the city has a population of around 676,000 people. The
inhabitants are known as Palermitani or, poetically, panormiti. The languages spoken by
```

```
its inhabitants are the Italian language and the Palermitano dialect of the Sicilian
language. </p>
  </div>
  <div style="max-width: 700px; padding: 20px;">
  <h2>Cannoli</h2>
  <img style="width: 200px; float: left; margin:
10px;"src="https://www.giallozafferano.it/images/ricette/206/20605/foto_hd/hd650x433_wm
.jpg">
  <p>Cannoli (Italian pronunciation: [kanËˆnÉ”Ëli]; Sicilian: cannula) are Italian
pastries that originated on the island of Sicily and are today a staple of Sicilian
cuisine[1][2] as well as Italian-American cuisine. Cannoli consist of tube-shaped
shells of fried pastry dough, filled with a sweet, creamy filling usually containing
ricotta. They range in size from "cannulicchi", no bigger than a finger, to the fist-
sized proportions typically found south of Palermo, Sicily, in Piana degli Albanesi.[2]
In the mainland Italy they are commonly known as cannoli siciliani (Sicilian cannoli).
</p>
  </div>
</center>
</main>
<footer style="background-color: black; text-align: center; color: white; font-size:
18px; padding: 20px;">Enjoy Palermo!</footer>
</body>
```

Co-funded by the
Erasmus+ Programme
of the European Union

Module: Web development

Topic: HTML and CSS

Task sheet D1.4: Create website for a specific target group (Design Thinking and Web development)

Time: 2:00 hour(s)

General description:

The students will think of a specific target group to create a website, they will study the whole Design Thinking process through which they will identify the needs of the target group, and will create a website following the target group's needs. You could also divide this activity and only do the design thinking part, or only the creating the website part with predefined particular needs. It is an activity to practice HTML and CSS skills.

Learning objective(s):

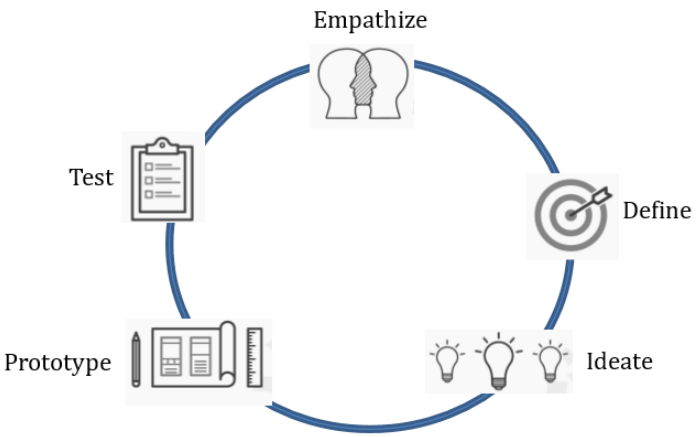
- Understand the Design Thinking cycle and how useful it is in every kind of project creation process.
- Start to think on how to create a product thinking on the target needs.
- Improve the HTML and CSS skills on creating a web page including
  - changing and adding colours and styles
  - inserting images and videos
- Learn how to create a webpage thinking on the user experience, thinking on what is interesting for the user, how to organize the content, etc.

Material required:

- Computer or laptop
- Internet connection
- Text editor (online or offline): Sublime Text/Brackets/W3Schools online editor

Description of the activity:

1. The teacher will present, if not done in previous lessons, the Design Thinking cycle:



2. The trainer will explain the different phases:
    - Empathize: Understand who is the target of your solution focused, thinking about their needs, their daily life, their experience, etc.
    - Define: What does my client need? What are the solutions to their possible problems?
    - Ideate: What are the possible solutions to my client problems?
    - Prototype: Create a first version of the solution to thinking on the previous step.
    - Test: Check the viability of the solution and the feedback of the client.

3. In this lesson we will go through the “Empathise”, “Define”, “Ideate” and “Prototype” steps.

   The trainer divides the class in groups (between 3 and 5 participants) and different target groups to each group. These target groups should have particular needs. For example, we used an existing code from a page with information for tourists in Palermo so therefore we identified target groups like eldery, children, tourists with disabilities, sport fanatics, etc.

4. Once each group has its target client, they have to start with the Empathize phase of the cycle. The trainer will show some questions and will give a short amount of time to answer them. Examples of questions are:
    - Who are they?
    - How old they are?
    - Where they live?
    - How much money do they have?
    - What are their problems?
    - What is important for them?

5. After this, we will go to the “Define” phase, picking one problem and defining it in a more extended and person-focused way. Again, some questions and a short amount of time will be given to the students. Examples of questions are:
    - What are the possible solutions to the problem?

6. Next phase is the “Ideate” phase, where the students will do a brainstorming with all the possible solutions oriented on a webpage. Is important in this part to let students know that there is no wrong solution, and the more solutions they have in this phase, the better. They will have a short amount of time to write all the solutions down.
    - Possible solutions: Social networks, maps, forums, accessibility of the page, content, etc.

7. For the final part, they will create a site that gives a solution to the problem they thought of.

8. In the final part of the class, each group will present the process which they went through on thinking on the problem, and the webpage they have created to respond to those needs.

How to adapt to different learners:

For the web page creation, you can give an already created base web page structure in which they can work on.

For the more advanced students, coding challenges can be created in order to make them use more advanced HTML and CSS code. For basic learning you can leave them free in what they want to create for the web page, for more advanced learners you can give certain requirements and limitations such as the insertion of videos, meta structures, style structures, etc.

Additional information:

- HTML reference guide
- W3Schools - Guide for every HTML element and CSS rule, and examples for each one of them

Co-funded by the
Erasmus+ Programme
of the European Union

Module: Web development

Topic: Javascript

Task sheet D1.5: Make your webpage interactive

Time: 2:00 hours

General description:

JavaScript is a programming language that allows the developer to change the content of the web page dynamically. In this task sheet students will learn how Javascript works through 5 very simple exercises:

1. The first one will show the student how the JavaScript is executed on the loading of the webpage and how it can change the webpage content.
2. Through the second exercise, the student will define a function and make use of an event to execute the JavaScript code after the user interaction and show a browser alert.
3. The third one, instead of using a browser alert, that is not very user friendly, will use the event to change directly the HTML content.
4. The fourth one will show the student the use of different events attached to a button, showing how to change the button appearance, creating, in this way, a more appealing and user responsive interface.
5. The last one, will show how to change the 'src' attribute of an image, showing in this way how the student can make a slider or an image gallery.

Learning objective(s):

- The student will understand that JavaScript allows the developer to change content dynamically and modify the HTML or CSS information on the client side, when the webpage is being shown to the user.
- The student will learn the structures of JavaScript code
- The student will learn how to change HTML/CSS code on the loading of the page
- The student will learn to create a function and attach it to an event.

Material required:

- Computer or laptop
- Internet connection
- Text editor (online or offline): Sublime Text/Brackets/W3Schools online editor

Description of the activity:

1. The trainer will make an introduction to JavaScript, making the student understand that JavaScript code allows the developer to change the content of a webpage on the client side without reloading the page.

2. The trainer will explain the <script> HTML tag, where the JavaScript code is introduced on the web page file.

3. In order to make the students understand that, if not indicated differently, the JavaScript code will be executed by the browser as the page loads, the trainer can propose to paste the following code on the Code Editor and make them load the webpage. The text inserted by the JavaScript will be shown directly afterwards the load of the page.

```
<html>
   <body>
      <script language = "javascript" type = "text/javascript">
         <!--
            document.write("Hello World!")
         //-->
      </script>
   </body>
</html>
```

4. After making this little exercise, the trainer will show the students how to make a function and attach it to an HTML button using the following code. This will show an alert message, something that explains how the code works but that is normally not used because it is not very user friendly (except very unusual cases, like debugging or warning messages).

```
<html>
   <head>
      <script type = "text/javascript">
         <!--
            function sayHello() {
               alert("Hello World")
            }
         //-->
      </script>
   </head>

   <body>
      <input type = "button" onclick = "sayHello()" value = "Say
Hello" />
   </body>
</html>
```

5. As a final exercise the trainer will explain how to create a function in JavaScript and how to look for an HTML object on the web page document using the JavaScript functions `document.getElementById` and `document.getElementByClass.`
   - **`document.getElementById("example"):`** Will look for the first element in the document with the **ID** "example".
   - **`document.getElementByClass("example"):`** Will look for the first element in the document with the **Class** "example".

   These two JavaScript methods are normally used to retrieve HTML elements on the document. After finding the element that we want to modify, a wide variety of attributes can be changed, for example:
   - The attribute **'innerHTML'** to change the content of the HTML element. For example:

```
<!DOCTYPE html>
<html>
<head>
<script>
function myFunction() {
  document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>
</head>
<body>

<h1>A Web Page</h1>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>

</body>
</html>
```

6. After this, the trainer will show how to use other event listeners on the button and how to change the style using JavaScript, for example, normally on web pages when the user passes the mouse over a button it changes style in order to make the page more visually attractive and more user friendly. The next code will change the style of the button using the `'onmouseover'` and `'onmouseout'` events on the HTML element.

```
<!DOCTYPE html>
<html>
<head>
<script>
function myFunction() {
  document.getElementById("demo").innerHTML = "Paragraph changed.";
}
function mouseOver() {
  document.getElementById("button").style.color = "white";
  document.getElementById("button").style.backgroundColor = "green";
  document.getElementById("button").style.fontWeight = "bold";
}
function mouseOut() {
  document.getElementById("button").style.color = "black";
  document.getElementById("button").style.backgroundColor = "#e7e7e7";
  document.getElementById("button").style.fontWeight = "normal";
}
</script>
</head>
<body>
<h1>A Web Page</h1>
<p id="demo">A Paragraph</p>
<button id="button" type="button" onclick="myFunction()"
onmouseover="mouseOver()" onmouseout="mouseOut()">Try it</button>
</body>
</html>
```

7. In the last example the trainer will show how to change the 'src' attribute of an image element, showing how to change the image when clicking a button and creating in this way a dynamic photo gallery.

```
<!DOCTYPE html>
<html>
<head>
<script>
function changeImage() {
    var url = document.getElementById('imageToChange').src;
    if (url == 'https://images.freeimages.com/images/small-
previews/e7c/recipe-1538714.jpg') {
        document.getElementById('imageToChange').src =
'https://images.freeimages.com/images/small-previews/b9e/recipe-for-
cookbook-3-1483847.jpg';
    } else {
        document.getElementById('imageToChange').src =
'https://images.freeimages.com/images/small-previews/e7c/recipe-
1538714.jpg';
    }
}
function mouseOver() {
  document.getElementById("button").style.color = "white";
  document.getElementById("button").style.backgroundColor = "green";
  document.getElementById("button").style.fontWeight = "bold";
}
function mouseOut() {
  document.getElementById("button").style.color = "black";
  document.getElementById("button").style.backgroundColor = "#e7e7e7";
  document.getElementById("button").style.fontWeight = "normal";
}
</script>
</head>
<body>
<h1>A Web Page</h1>
<p>Gallery:</p>
<img id="imageToChange"
src="https://images.freeimages.com/images/small-previews/e7c/recipe-
1538714.jpg" alt="Letter A">
<div>
<button id="button" type="button" onclick="changeImage()"
onmouseover="mouseOver()" onmouseout="mouseOut()">Change</button>
</div>
</body>
</html>
```

How to adapt to different learners:

- Depending on the level of the students the trainer can give the code to the students, or if they are more advanced on programming the students can try to write the exercises by themselves.
- On the final step, the challenge can be adapted depending on the skills of the students, a few examples of challenges can be:
    - Change the style of some HTML elements depending on different events
    - For more advanced users they can try to create a gallery with more than two pictures and a “Previous” and “Next” button.

Additional information:

JavaScript is a programming language that allows the developer to change the content of the web page dynamically, for example, showing the date if the user clicks a button.
The programs in this language are called *scripts*. They can be written right in a web page’s HTML and run automatically as the page loads or when an event is triggered (click on a button, scroll down the webpage, etc.).
The advantages of using JavaScript are:

- **Increased interactivity** − You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.
- **Richer interfaces** − You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.
- **Less server interaction** − You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.
- **Immediate feedback to the visitors** − They do not have to wait for a page reload to see if they have forgotten to enter something.

Other resources:

- JavaScript tutorial: w3schools
- Online Course on JavaScript: CodeAcademy

Co-funded by the
Erasmus+ Programme
of the European Union

Module: Web development

Topic: Javascript

Task sheet D1.6: Create your first interactive e-commerce

Time: 2:00 hours

General description:

JavaScript is a programming language that allows the developer to change the content of the web page dynamically without reloading the page. Thanks to this aspect, it is commonly used on the validation of the information introduced by the end user (i.e validation of numbers when filling in the age and not numbers) and to give interactivity to the web page. In this task sheet we will create a very easy e-commerce product page where some input fields will be validated.

Learning objective(s):

- The student will understand that JavaScript allows the developer to change content dynamically and modify the HTML or CSS information on the client side, when the webpage is being shown to the user.
- The student will learn the structures of JavaScript code
- The student will learn how to change HTML/CSS code on the loading of the page
- The student will learn to create a function and attach it to an event.

Material required:

- Computer or laptop
- Internet connection
- Text editor (online or offline): Sublime Text/Brackets/W3Schools online editor

Description of the activity:

1. The trainer will make an introduction to JavaScript, making the students understand that JavaScript code allows the developer to change the content of a webpage on the client side without reloading the page.

2. The trainer will explain the <script> HTML tag, where the JavaScript code is introduced on the web page file.

3. The trainer will propose the students to create a single product page for e-commerce. Depending on the level of the students they can create this page by themselves or they can use the code that you can find at the end of this task sheet.

4. Once the base for the product web page is created, the trainer will ask the student to add buttons in order to choose the colour of the product. This button will, through JavaScript, change the image that is shown on the web page to match the choice of the user. To do this the student will have to create two buttons in HTML and create a function on javascript that changes the image when the button is clicked. The following code is an example on how to do it.

```
…
<div class="color-choose">
      <span>
                <button onclick="changeImage(this)" class="choice-
      button" id="red-color-button">Red</button>
      </span>
      <span>
                <button onclick="changeImage(this)" class="choice-
      button" id="yellow-color-button">Yellow</button>
      </span>
</div>
…
<script language="javascript">
  function changeImage(element) {
       if(element.id == "red-color-button"){
         document.getElementById('product-
image').src='http://www.pngall.com/wp-content/uploads/2016/04/T-
Shirt-PNG-HD.png'
       } else {
         document.getElementById('product-
image').src='http://www.pngall.com/wp-content/uploads/2016/04/T-
Shirt-PNG-Pic.png'
       }
    }
</script>
```

5. After this, the students will add an input field to allow the webpage user to insert the quantity that he or she wants to buy. Depending on this quantity, the final price will change accordingly. To do this, the students have to create a new function that will be called upon when the information on the input is changed using the 'onchange' event.

   The students have to pay attention to the conversion from “string” to “float number” to get the individual price from the webpage and be able to multiply it.
   After this the students can also add to this function a code that will control that the value of the “Quantity” that the user has written is a natural number.
   Following you can find an example on how to approach this challenge.

```
    <!-- Product Pricing -->
     <div class="product-price">
       <div>Unit price: <span id="unit-price">19.99¤</span></div>
       <div>
         <span>Quantity:</span>
         <input onchange="showFullPrice()" id="product-quantity" value="1"/>
       </div>
       <div>Total price: <span id="total-price">19.99¤</span></div>
       <a href="#" class="cart-btn">Add to cart</a>
     </div>

  <script language="javascript">

     function showFullPrice(element)  {
       var quantity = parseFloat(document.getElementById('product-
quantity').value);
       if(!isNaN(quantity)) {
         var unitPrice = parseFloat(document.getElementById('unit-
price').innerHTML)
         var totalPrice = quantity*unitPrice;
         document.getElementById('total-price').innerHTML = totalPrice + '¤';
       }
     }
  </script>
```

The most important elements to understand this codes are the following functions:

- **isNaN():** The isNaN() function determines whether a value is an illegal number (Not-a-Number). This function returns true if the value equates to NaN. Otherwise it returns false.
- **parseFloat():** The parseFloat() function parses a string and returns a floating point number.
- **innerHtml:** In this example we use this function either to access the content of an HTML element or also to change its content assigning a new value.

How to adapt to different learners:

- Depending on the level of the students the trainer can give the code to the students, or if they are more advanced on programming the students can try to write the exercises by themselves.
- The student can try to give more interactivity to the webpage, for example, another field for a discount code that if inserted correctly will apply a percentage discount to the total price.

Additional information:

JavaScript is a programming language that allows the developer to change the content of the web page dynamically, for example, showing the date if the user clicks a button.
The programs in this language are called *scripts*. They can be written right in a web page's HTML and run automatically as the page loads or when an event is triggered (click on a button, scroll down the webpage, etc.).
The advantages of using JavaScript are:

- **Increased interactivity** − You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.
- **Richer interfaces** − You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.
- **Less server interaction** − You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.
- **Immediate feedback to the visitors** − They don't have to wait for a page reload to see if they have forgotten to enter something.

Other resources:

- JavaScript tutorial: w3schools
- Online Course on JavaScript: CodeAcademy
- Example code:

```
<!DOCTYPE html>
<html>
<head>
  <title></title>

<style>
  /* Basic Styling */
html, body {
  height: 100%;
  width: 100%;
```

```
  margin: 0;
  font-family: 'Roboto', sans-serif;
}

.row {
  max-width: 1200px;
  margin: 0 auto;
  padding: 15px;
  display: flex;
}

.column {
  flex: 50%;
}

/* Left Column */
.column img {
  max-width: 500px;
  transition: all 0.3s ease;
}

/* Product Description */
.product-description {
  border-bottom: 1px solid #E1E8EE;
  margin-bottom: 20px;
}
.product-description span {
  font-size: 12px;
  color: #358ED7;
  letter-spacing: 1px;
  text-transform: uppercase;
  text-decoration: none;
}
.product-description h1 {
  font-weight: 300;
  font-size: 52px;
  color: #43484D;
  letter-spacing: -2px;
}
.product-description p {
  font-size: 16px;
  font-weight: 300;
  color: #86939E;
```

```
  line-height: 24px;
}
.color-choose button {
  display: inline-block;
  background-color: white;
  border: 2px solid grey;
  border-radius: 6px;
  font-size: 16px;
  color: black;
  text-decoration: none;
  padding: 12px 30px;
  transition: all .5s;
  margin-bottom: 20px;
}

.color-choose button:hover {
  background-color: #aaaaaa;
}

/* Product Price */
.product-price {
  display: flex;
  align-items: center;
}

.product-price span {
  font-size: 26px;
  font-weight: 300;
  color: #43474D;
  margin-right: 20px;
}

.cart-btn {
  display: inline-block;
  background-color: #7DC855;
  border-radius: 6px;
  font-size: 16px;
  color: #FFFFFF;
  text-decoration: none;
  padding: 12px 30px;
  transition: all .5s;
}
.cart-btn:hover {
```

```
  background-color: #64af3d;
}
</style>

</head>
<body>

<main class="row">

  <!-- Left Column / Headphones Image -->
  <div class="column">
    <img       src="http://www.pngall.com/wp-content/uploads/2016/04/T-Shirt-PNG-HD.png"
alt="">
  </div>


  <!-- Right Column -->
  <div class="column">

    <!-- Product Description -->
    <div class="product-description">
      <span>T-Shirts</span>
      <h1>Very cool t-shirt</h1>
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor
in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint
occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est
laborum.</p>
    </div>

    <!-- Product Configuration -->
    <div class="product-configuration">

      <!-- Product Color -->
      <div class="product-color">
        <span>Color</span>

        <div class="color-choose">
          <div>
            <button id="red-color-button">Red</button>
          </div>
```

```
        </div>

      </div>

    </div>

    <!-- Product Pricing -->
    <div class="product-price">
      <span>19,99€</span>
      <a href="#" class="cart-btn">Add to cart</a>
    </div>
  </div>
</main>

</body>
</html>
```

Co-funded by the
Erasmus+ Programme
of the European Union

Module: Web development

Topic: HTML/CSS

Task sheet D1.7: Create a working contact form

Time: 2:30 hour(s)

General description:

We will create a contact form which will use a send action written in php in order to send the message.

Learning objective(s):

- Familiarize with the use and syntax of HTML and CSS
- An introduction to the PHP language and specifically in the send function
- Create a working contact form

Material required:

- Computer
- Internet connection

Necessary software:

- Notepad++

Description of the activity:

Participants will implement a very basic combination of HTML5 with CSS and PHP in order to create a contact form:

- Create the index.php file
- Create a section for the form (<form></form>) and add the desired fields
- Create a style.css file and link it in the index.php file
- Style the elements in the css file
- Add the send function just above the form tag

How to adapt to different learners:

- The trainers can help the learners to find which section of the code they should edit, according to the change they want to do.

According to the level of the participants, they can suggest different edits (eg. something simple, like adding some text or something more advanced – adding a new widget position)

Additional information:

Example code:

index.php

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>My Contact Form</title>
</head>
<body>
   <header class="main">
      <h1>My Contact Form</h1>
   </header>
   <section class="main">
      [Form code goes here]
   </section>
</body>
</html>
```

Form code:

```
<form>
      <label>Your Name:</label>
      <input name="name" placeholder="Goes Here">
      <label>Your Email:</label>
      <input name="email" type="email" placeholder="Goes Here">
      <label>Your Message:</label>
      <textarea name="message" placeholder="Goes Here"></textarea>
      <input id="submit" name="submit" type="submit" value="Submit">
</form>
```

style.css:

```
label {
    display:block;
    margin-top:50px;
    letter-spacing:1px;
}
/* This section centers our complete page */
.main {
    display:block;
    margin:0 auto;
    width:500px;
}
/* This section centers the form inside our web page*/
form {
    margin:0 auto;
    width:420px;
}
/* Applying styles to our text boxes */
input, textarea {
            width:400px;
            height:27px;
            background:#666666;
            border:2px solid #f6f6f6;
            padding:10px;
            margin-top:10px;
            font-size:0.7em;
            color:#ffffff;
}
textarea {
            height:200px;
            font-family:Arial;
}
#submit {
    width:85px;
    height:35px;
    background:url(submit.png);
    text-indent:-9999px;
    border:none;
    margin-top:20px;
    cursor:pointer;
}
```

send function:

```php
<?php
    $name = $_POST['name'];
    $email = $_POST['email'];
    $message = $_POST['message'];
    $from = 'From: My Contact Form';
    $to = 'salman@mywebsite.com';
    $subject = 'Wassup?';

    $body = "From: $name\n E-Mail: $email\n Message:\n $message";

    if ($_POST['submit']) {
       if (mail ($to, $subject, $body, $from)) {
       echo '<p>Message Sent Successfully!</p>';
       } else {
       echo '<p>Ah! Try again, please?</p>';
       }
    }
 ?>
```

Co-funded by the
Erasmus+ Programme
of the European Union

Module: Web development

Topic: Wordpress

Task sheet D2.1: Connect wordpress on wamp server

Time: 01:00 hour

General Description:

This task sheet will teach you how to connect the Wamp server and Wordpress on your computer. WordPress is not a stand-alone application and needs server software to run. WampServer provides the necessary server environment so you can install and run WordPress on your local host.

Learning objective:

- Learn how to connect the Wamp server and Wordpress.

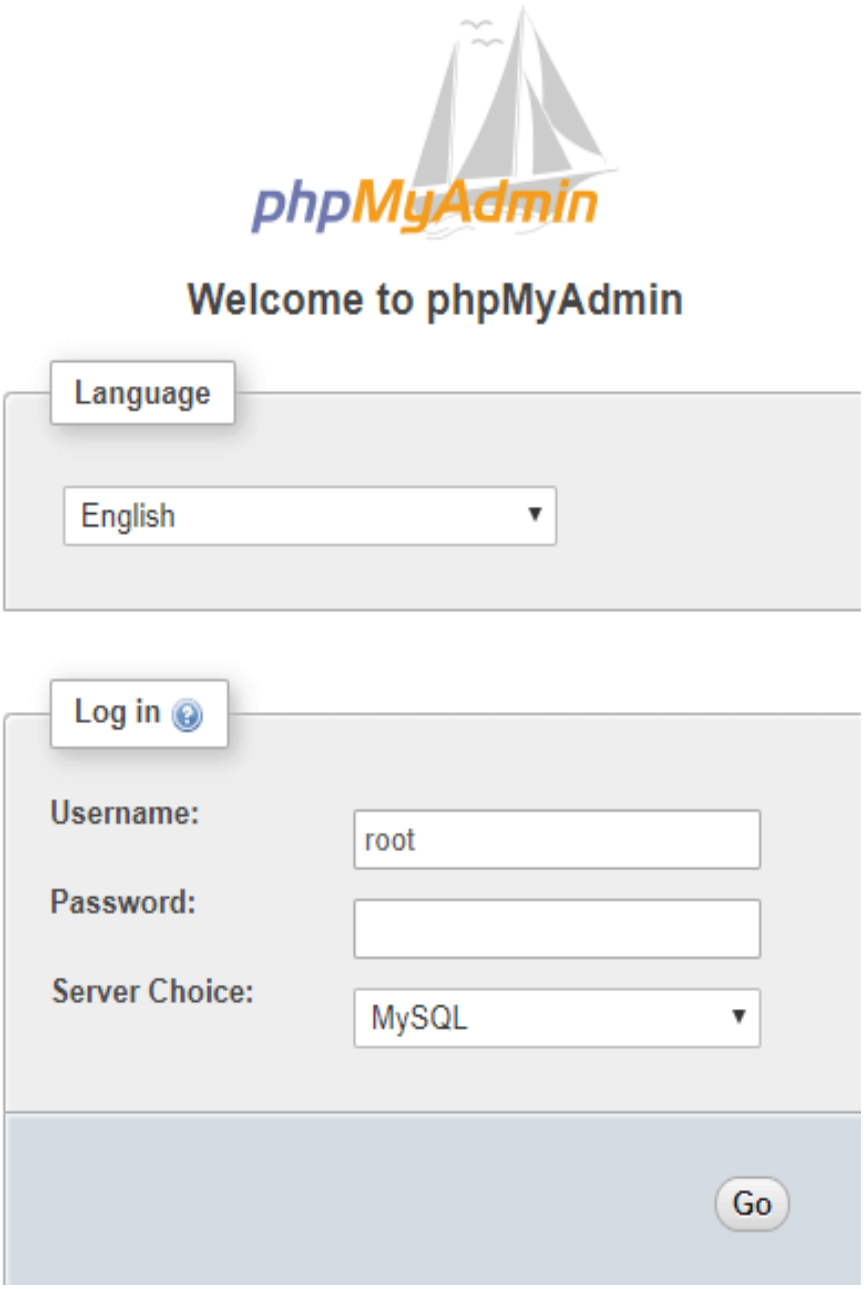
Material required:

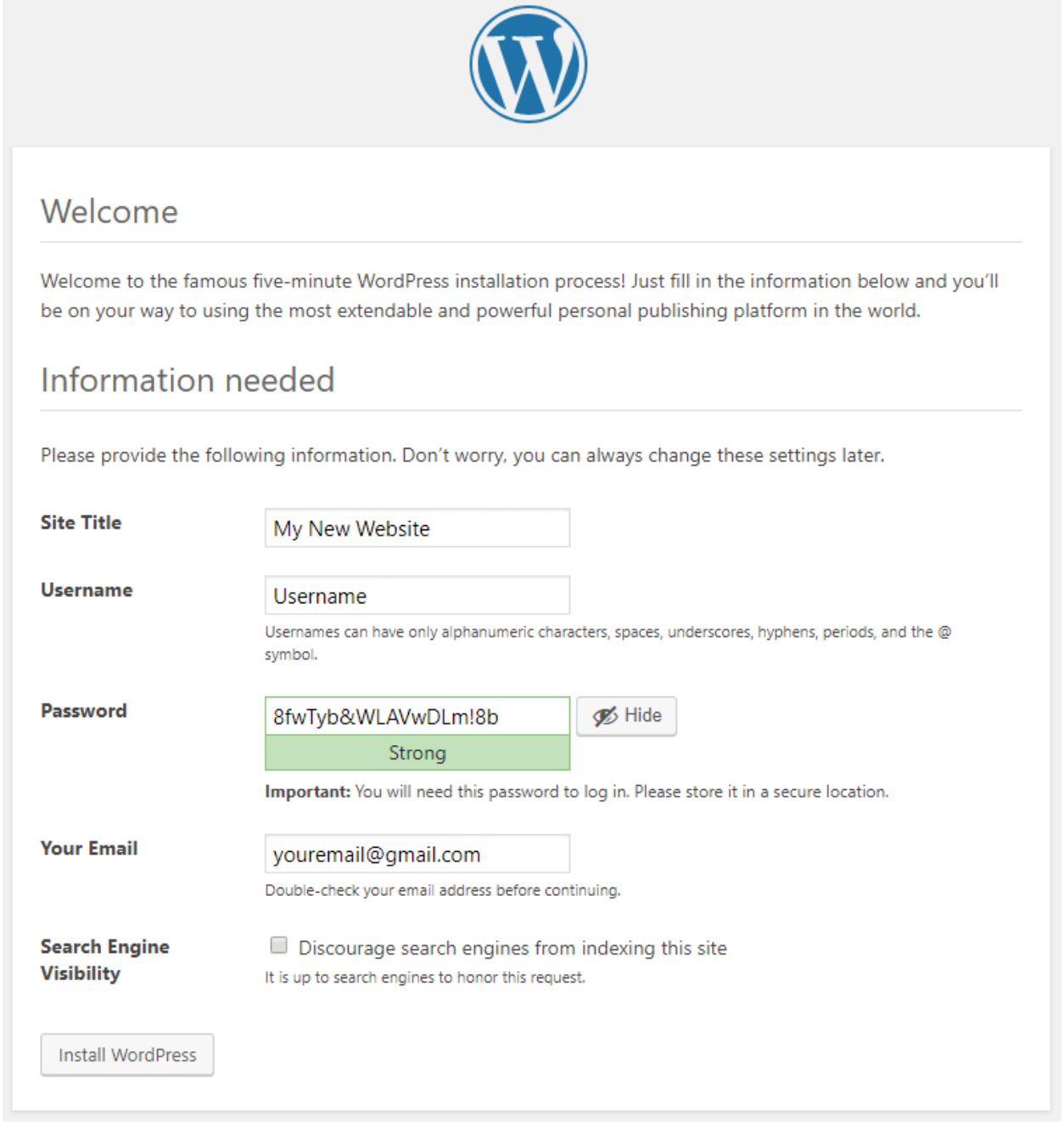
- Computer
- Internet connection

Description of the activity:

- Download and Install WampServer https://www.wampserver.com/en/
- Create a Database in phpMyAdmin
- Download and Install WordPress https://wordpress.org/
- Setup your WordPress Configuration.

Co-funded by the
Erasmus+ Programme
of the European Union





Welcome

Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Don't worry, you can always change these settings later.

Site Title My New Website

Username Username

Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

Password 8fwTyb&WLAVwDLm!8b Hide

Strong

Important: You will need this password to log in. Please store it in a secure location.

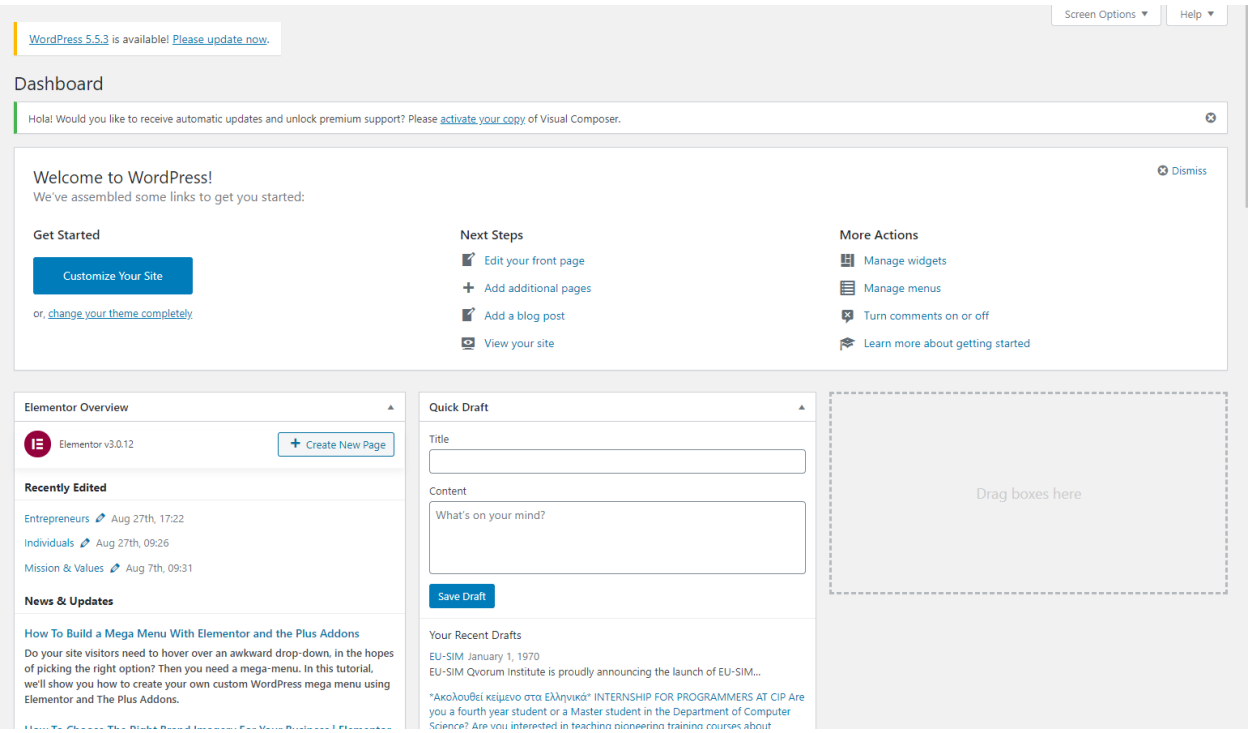
Your Email youremail@gmail.com

Double-check your email address before continuing.

Search Engine Visibility Discourage search engines from indexing this site

It is up to search engines to honor this request.

Install WordPress



How to adapt to different learners:

This task is the basic exercise to teach how to connect the Wamp server and Wordpress on your computer. The students can learn this procedure only with practice.

Additional information:

- Tutorial: "How to Install WordPress on Windows using WAMP Server" https://zuziko.com/tutorials/how-to-install-wordpress-on-windows-using-wamp-server/

Co-funded by the
Erasmus+ Programme
of the European Union

Module: Web development

Topic: Wordpress

Task sheet D2.2: Create pages and posts on wordpress

Time: 01:00 hour

General Description: This task sheet will teach you how to create pages, posts and media using wordpress. The purpose of creating pages is to host the blogs and their details. A blog contains a title, a content, an image and categories. A website can host many blogs/articles. Also, the media are a part of the articles and a way to attract the users to read the articles.

.

Learning objective:

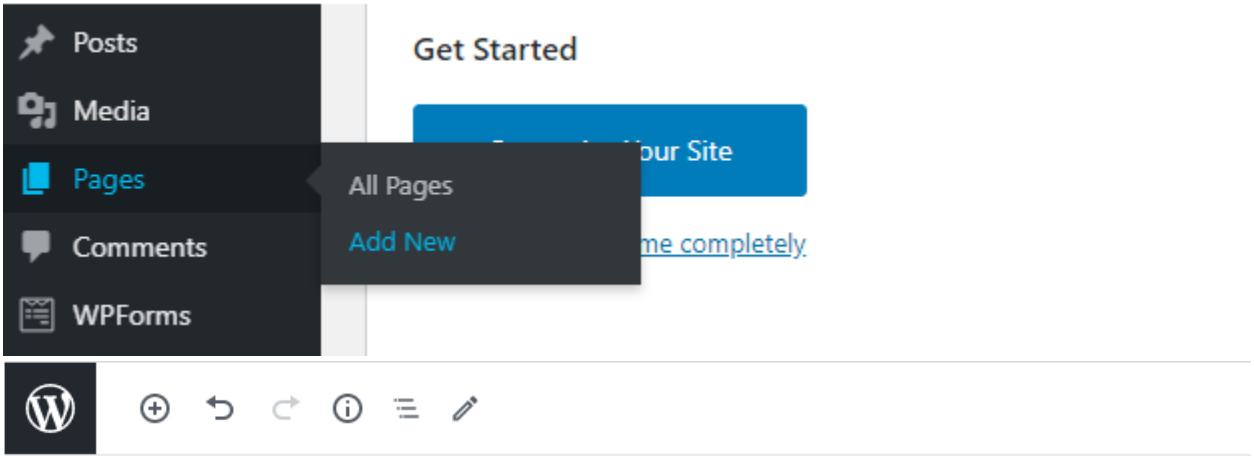
- Learn how to create pages in wordpress.
- Learn how to create posts.
- Learn how to add media.

Material required:

- Computer
- Internet connection

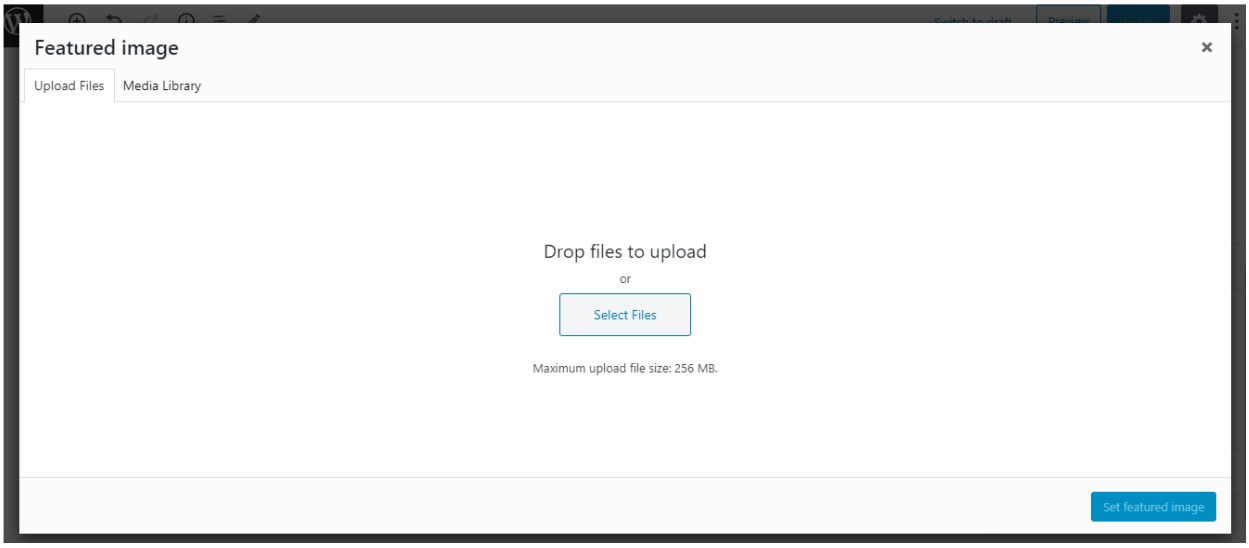
Description of the activity:

- On the left menu of Wordpress, click on the section “Pages” and then click on the button “Add new page”.
- First write the page title and automatically it will generate a permalink.
- In the next field you can write the description or you can add content.
- On the right menu click on the tab “Featured image” and you can add a picture or a video.
- The final step is to click on the button “Publish”.



Home

Start writing or type / to choose a block



How to adapt to different learners:

This task contains the basic steps to teach the students how to create pages, posts and media on wordpress.

For social platform: Buddypress plugin
https://premium.wpmudev.org/blog/buddypress-guide/

For E-Commerce: WooCommerce plugin:
https://docs.woocommerce.com/documentation/plugins/woocommerce/getting-started/

Changing wordpress theme:
https://www.wpbeginner.com/beginners-guide/how-to-install-a-wordpress-theme

Additional information:

- Tutorial: “How to Create a Separate Page for Blog Posts in WordPress” https://www.wpbeginner.com/wp-tutorials/how-to-create-a-separate-page-for-blog-posts-in-wordpress/

Co-funded by the
Erasmus+ Programme
of the European Union

Module: Web development

Topic: Wordpress

Task sheet D2.3: Edit PHP files

Time: 2:00 hour(s)

General description:

We will edit some of the core wordpress files in order to alter slightly the appearance of the header/footer website. In some cases, we may need to alter the size of the header or add some additional information on the footer. Editing those files is a quick way to do it, instead of searching and using the appropriate plugin.

Learning objective(s):

- Familiarize with the use and syntax of php
- Learn to edit core file of the CMS
- Familiarize with the file structure of wordpress

Material required:

- Computer
- Internet connection

Necessary software:

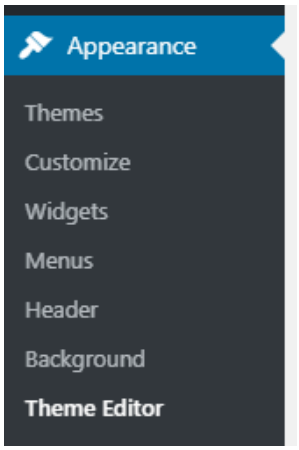
- Notepad++
- Wordpress installation

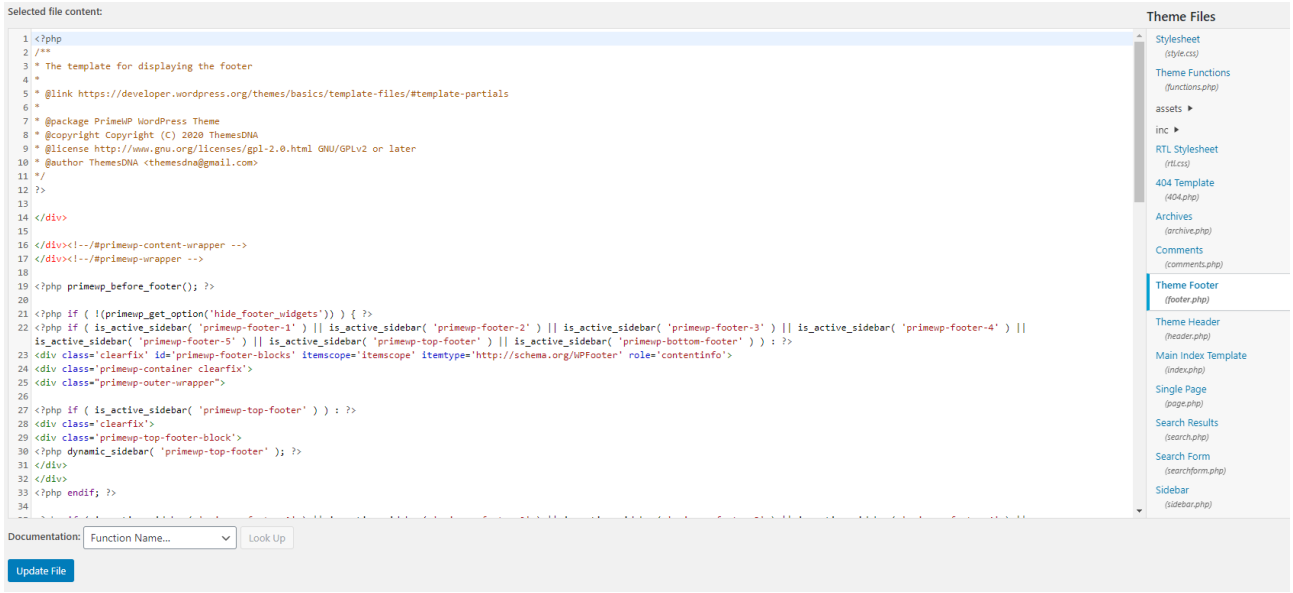
Description of the activity:

The participants navigate through wordpress to the file editor and they add some custom code. An example is adding a widget position on the header, by editing the functions.php, styling it by adding css code and eventually adding a widget in that position through wordpress.

How to adapt to different learners:

- The trainers can help the learners to find which section of the code they should edit, according to the change they want to do.
- According to the level of the participants, they can suggest different edits (eg. something simple, like adding some text or something more advanced – adding a new widget position)





This is what you should see, if you have navigated to the correct location to find the theme editor. In the main panel you can edit the code and in the panel on the right, there is a list with the theme files available for editing.

Additional information:

Location of the theme editor:

On the left menu of Wordpress, click on the section “Appearance” and then click on the button “Theme editor”.

- On the right pane of the page, there is a list with php files. For editing the header select Theme Header (header.php) or for editing the footer, select Theme Footer(footer.php)
- Browse through the code (visible in the center of the page), find the section you want and make the appropriate changes.
- Click the update file button, usually located on the bottom.

https://www.wpbeginner.com/wp-tutorials/how-to-edit-the-footer-in-wordpress/
https://www.wpbeginner.com/plugins/how-to-add-header-and-footer-code-in-wordpress
https://www.wpbeginner.com/wp-themes/how-to-add-a-wordpress-widget-to-your-website-header/