

# Assignment - Refactoring and organizing the e-commerce project

## Exercise

## Task description

In this task you will connect all the pages of the e-commerce site, organize the files, and prepare the project for further development in JavaScript.

You will use the previously created clean structure, connect the pages through correct links in the navigation, separate styles and scripts into separate files, and achieve a consistent layout across all pages.

In the end, when you click on any element on the page, everything will work! This means that all links are connected correctly, the paths are accurate, and your project is ready for further development.

## Objective

The objective of this task is to improve the e-commerce site so that:

- HTML files should be in the main folder, and styles, scripts, and images should be in separate folders (css/, js/, and images/).
- External styles and scriptures are correctly linked in relative ways.
- All pages are connected by navigation;
- Remove unnecessary duplicates and inline styles.
- The site has the same look and feel on all pages.

Tas

ks

1. Go back to your e-commerce site from the previous course and reorganize the structure so that the files are placed as follows:

```

project/                                     |      index.html
|                                           |
|                                           |      about.html
| contact.html                             |      (and all the other pages you have) |      css/
|                                           |      style.css
|                                           |
|                                           |      js/

```



- Move all images to the images/ folder.
- Move the CSS files to the css/ folder.
- Create the js/main.js file (it may be empty for now) in the js/ folder.

2. In each HTML file, import the external CSS file. If you have multiple CSS files, make sure that the order of the CSS files is correct.

3. Move the styles inline to style.css (e.g., `.btn-primary { ... }`) and remove the `style="..."` attribute from HTML.

4. Correct paths to images if, after rearranging, they are no longer displayed correctly.

5. Linking pages (a `href`): In the navigation of each page put relative links and wherever there is the element `a`. Make sure the links work on all pages.

6. Consistency:

- Check that the header, footer, and basic typography look the same on all pages.
- Remove duplicate CSS; use Bootstrap (e.g., `.container`, `.section`, `.btn`).

Tips for successfully performing the task

Relative paths are essential: it's not the same if you're writing

```
<link rel="stylesheet" href="css/style.css">
```

or

```
<link rel="stylesheet" href="style.css">
```

- Always add another to it `<img>` and a meaningful title.
- Use consistent file names.
- If you have repetitive HTML (header/footer), make sure it's the same on all pages.
- Leave comments where they can help you understand the code.
- Before submitting, do the following checks:

- ❖ Technical verification (paths, structure, 404 errors, browser console, valid links).
- ❖ UI consistency (same header/footer, typography, colors).
- ❖ Quality of refactoring (elimination of inline styles, no duplicate CSS).
- ❖ Code organization (clear classes, comments, tidy CSS/JS).

## **Evaluation process**

The course coordinator will analyze the paper and provide you with feedback within a maximum of 24 hours. If you receive comments and suggestions, you will only have one chance to edit and resubmit the task.

## **Evaluation criteria**

The evaluation is made according to the following key criteria:

- Project structure and file organization Are the folders (css, js, and images) organized correctly? Are the file names clear and relevant?
- Correctness of paths and links Do all links (navigation and others with href) work? Are all images and CSS/JS files loading correctly, without errors?
- External CSS and code refactoring Have inline styles been removed where possible? Are newly defined classes and Bootstrap classes used?
- Design consistency Are the header, footer, typography, and basic styles consistent across all pages?
- The quality and readability of the code Is HTML semantic and orderly? Is CSS organized with clear classes and comments?

## **Sending the task**

When you're done with the assignment, upload your paper in zip format to the same page where the assignment description is located.

Save the task file as: prenume\_numa\_assignment01.zip

The maximum file size that can be sent in this way is 50 MB. If your file exceeds this limit, use the WeTransfer website.

## **How to use WeTransfer**

1. Open the [wetransfer.com](https://wetransfer.com) site.

2. Click on "Add your files" and select the files you want to send—it is mandatory to send the archived file in a zip.
3. Add your email address.
4. As a title, enter the same name as the file with the task.
5. In the message you must mention your name and surname and the name of the course.
6. The shelf life for downloading the file is 3 days.
7. Click on the three dots and choose the "Create link" option.
8. Then click on "Get a link."
9. Enter the verification code you received at your email address.
10. You send the generated link to the course coordinator on the assignment submission page.