



Intro to Web

Basics of HTML and CSS

Maksud Aghayev aka Shadow maghayev@student.42.us.org
42 Staff pedago@42.fr

Summary: Introduction to Web World.

Contents

I	Foreword	2
II	General instructions	3
III	Mandatory part	4
III.1	Foundation and Style	4
III.2	Forms and Inputs	6
III.2.1	Form for block creation	6
III.2.2	Form for accepting user login and password	6
IV	Turn-in and peer-evaluation	8

Chapter I

Foreword

We live in a time when websites have become part of our everyday lives, replacing newspapers and books, and offering humans a whole new range of opportunities to express themselves. You probably visit at least your favorite websites every day, whether it's for shopping, sending messages, playing games, checking the news or looking at pictures of your friends. Since the 1990s we have been using websites to have fun, make a living, and talk to previously unknown people.

[YouTube Evolution 2005-2016](#)

[Evolution of Facebook 2004-2016](#)

[r/forgottenwebsites](#)

[Internet Archive Wayback Machine](#)

Have you ever wondered how to build your own piece of online real estate?
How hard could it be?

I invite you to take that journey with me.

Chapter II

General instructions

You must use HTML5 and CSS only!

There are 2 **Mandatory Parts**.

Mandatory Parts.

Your folders must be arranged as follows:

- For mandatory part 1, create a folder named **part1** in root of the repository. This folder has to have 1 .html file with your page code.
- For mandatory part 2, create a folder named **part2** in root of the repository. This folder has to have 2 .html files for each part of the section.
- Root folder must also contain the folders: **css**, **assets**.

CSS folder can contain from 0 to * .css files.

Assets folder will contain subfolders:

- **icons**. Here you can place any icons or images you would like to use.
- **background**
 - . Here you can store background image of your choice.

Chapter III

Mandatory part

III.1 Foundation and Style

In this section we will start by laying foundation of your web page.

General instructions for this section are:

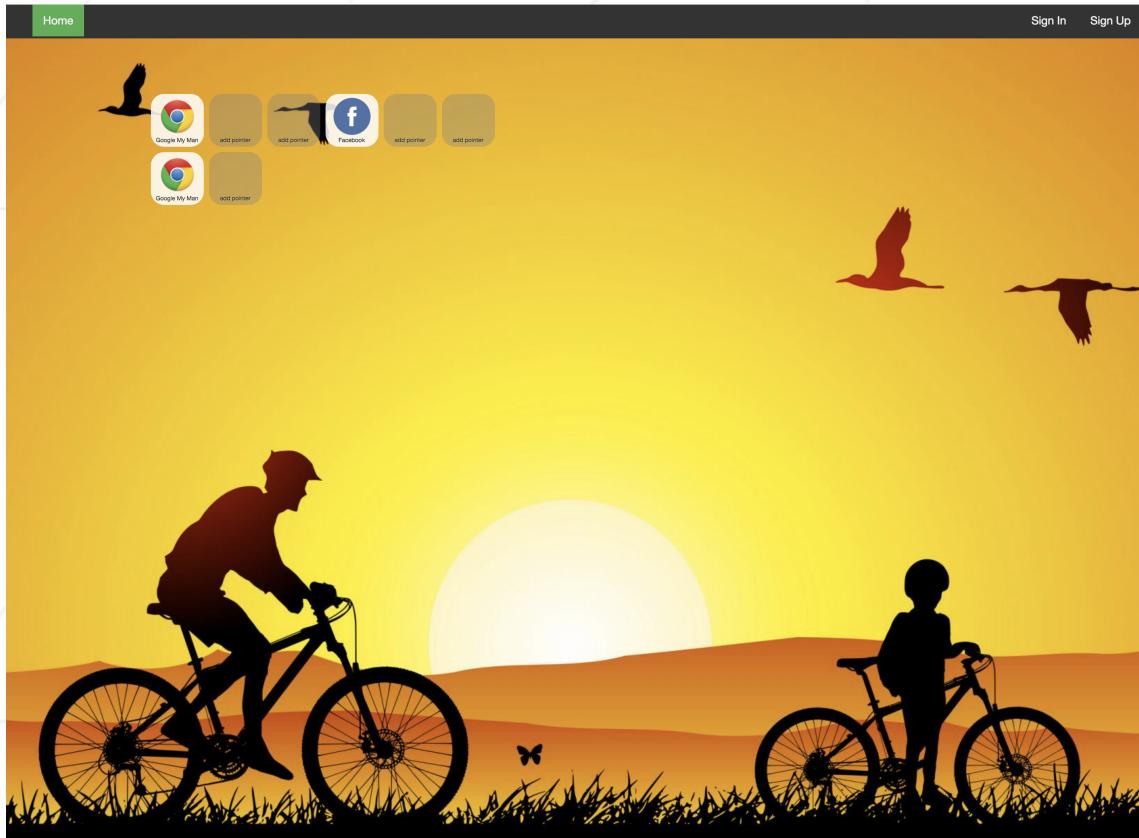
- You have to create a grid like page.
- You can style it however you like, the mandatory part is the correct structure of your page. E.g. having navigation bar, proper setting of padding and margin.
- Each grid block should not exceed 100x100 square.
- Grid blocks should be placed in logical order,
- Each grid item should have a tooltip when hovered, or should have a small text that will have a name of the item.
- As an idea for development you can build a grid where each block will link to real websites, or to specific images that you liked on the internet. Your imagination is a limit (and school norms :))



Great websites have one common thing, they all have favicon ;)



Think of how you plan to style your page. There are multiple ways to create styles. You can have them inline or as external file. Which way to use is totally up to you. Employ [Google](#) to let you know what are the best practises.



III.2 Forms and Inputs

In this section we will two forms and learn how input elements work. Also, as usual you have to style your form to the minimal pleasant appearance. Don't leave it in some default standard way. Imaginatiooonnn...

General instructions for this section are:

III.2.1 Form for block creation

Create a form that will adhere to the following guidelines:

- Form should be of method POST and action should be linked to nowhere.
- Form should have 5 **input fields**:
 - Input field which will hold string. Maximum characters in the field should be limited to 20.
 - Input field which will hold string. Maximum characters in the field should be limited to 50.
 - Input field which will hold URL. No cap on maximum length.
 - Input field for file selection.
 - Input field for submitting the form.

The form consists of five input fields:

- Title name: A text input field.
- Title tooltip: A text input field.
- URL path: A text input field.
- Choose File: A button with the text "Choose File" and "No file chosen".
- Send it out: A large green rectangular button.

III.2.2 Form for accepting user login and password

Create a form that will adhere to the following guidelines:

- Form should be of method POST and action should be linked to nowhere.
- Form should have 3 **input fields**:
 - Input field which will hold email. No cap on maximum length.
 - Input field which will hold password. No cap on maximum length.
 - Input field for submitting the form.

User email

Pswd

Please fill out this field.

Login



Take close look at required parameter of input field.



Think about forms the same way you would write information about someone. E.g.

- Name ...
- Age
- Image ..
- etc...

Chapter IV

Turn-in and peer-evaluation

Turn your work in using your `GiT` repository, as usual. Only work present on your repository will be graded in defense.

The next project in this series will build on the scaffolding here and use Javascript to make functional forms.