

# CSS Basics

CSS



**SoftUni Team**  
Technical Trainers



**Software University**

<http://softuni.bg>

# Table of Contents

1. What is CSS?
2. CSS in HTML document
3. Selectors
4. Block elements
5. Box model
6. Fonts
7. Comments in CSS



# Questions?

sli.do

**#TECH-FUND**



# CSS

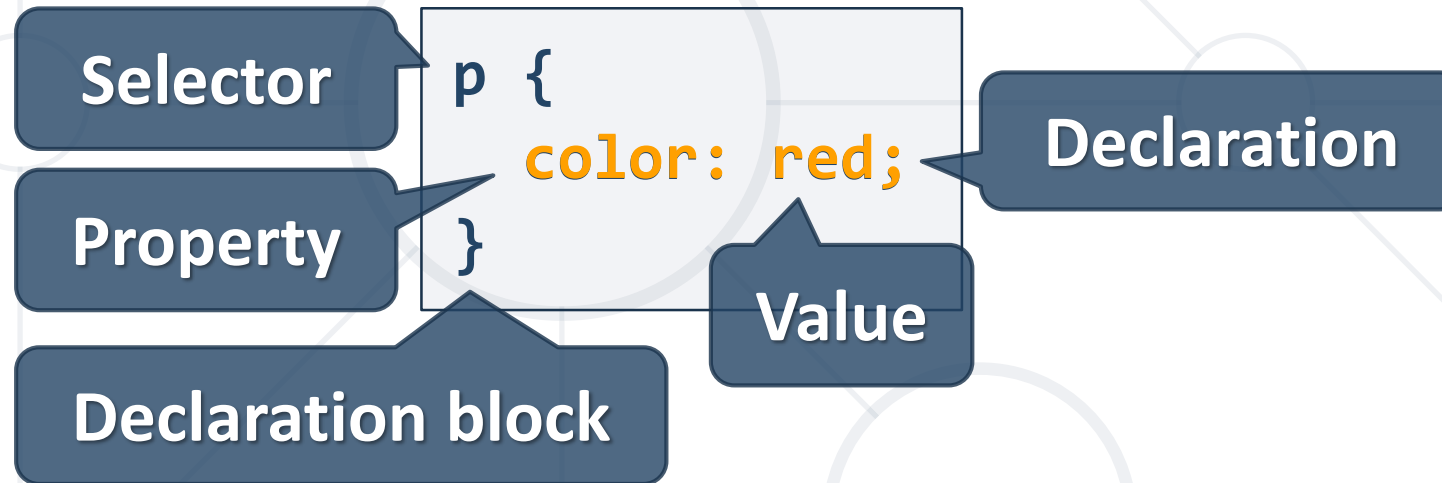
## Cascading Style Sheets

# Definition

- CSS (**C**ascading **S**tyle **S**heets) is the code used to **design** the HTML documents.
- An addition to the HTML.
- Designed to enable the separation of presentation and content, including layout, colors, and fonts.
- Not a programming language.
- Text file with **.css** extension.



- A CSS rule-set consists of a **selector** and a **declaration block**.



- The **selector** points to the HTML element you want to style.
- The **declaration block** contains one or more **declarations** separated by semicolons.


**HTML (+) CSS**



# **CSS and HTML**

## **How they communicate**

# CSS connection with HTML

- 
- When a browser reads a **style sheet**, it will **format** the HTML document according to the information in the **style sheet**.
  - There are three ways to insert CSS (style sheet):
    - Inline
    - Internal
    - External



- An **inline style** may be used to apply a unique style for a single element.
- To use inline styles, add the **style attribute** to the relevant element.
- The style attribute can contain any CSS **property**.

Style attribute

Style rules separated with ";"

```
<h1 style="color:blue;margin-left:30px;">This is a heading</h1>
```

property

value

# Problem: Heading stage upgrade

- Use your HTML code from the previous lecture
- Make your headings (**h1**) **inline** with **color blue**

**Lists exercise**

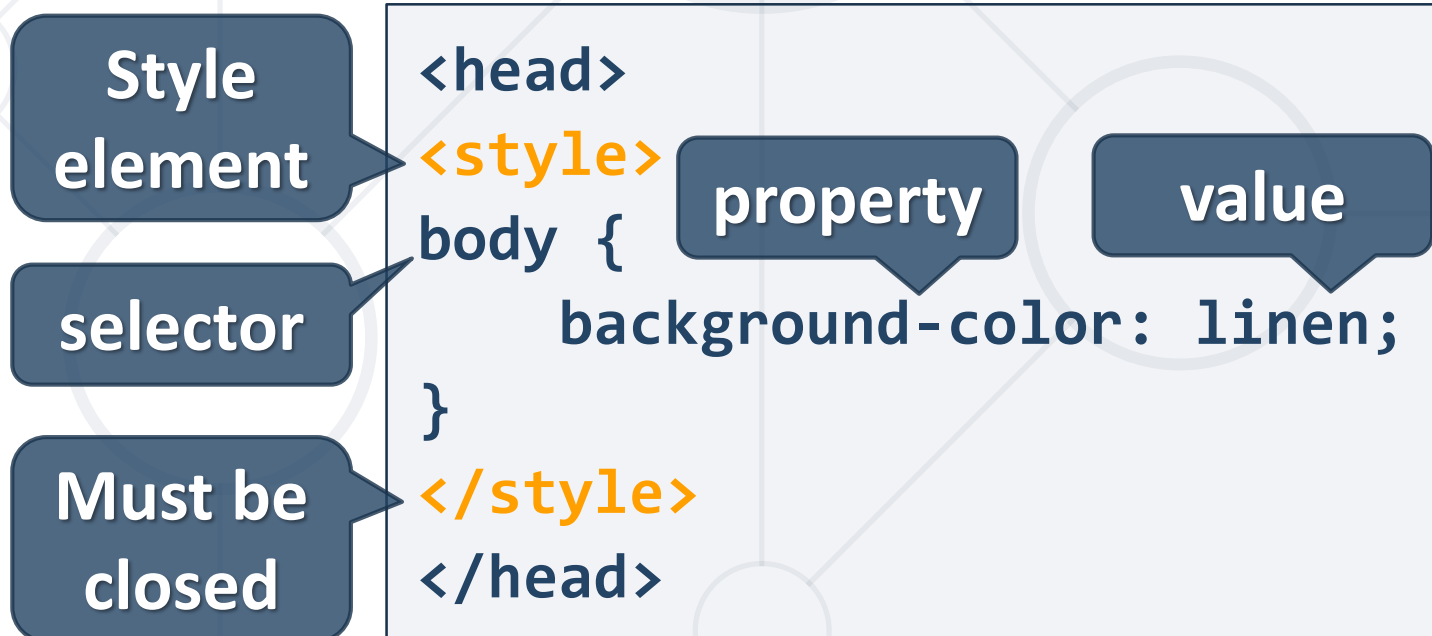
**Tables exercise**

**Forms exercise**

- Solution:

```
<h1 style="color: blue">Lists exercise</h1>  
<h1 style="color: blue">Tables exercise</h1>  
<h1 style="color: blue">Forms exercise</h1>
```

- An internal style sheet may be used if one single page has a unique style.
- Internal styles are defined within the **<style>** element, inside the **<head>** section of an HTML page.



# Problem: Paragraph stage upgrade

- Use your HTML code from the previous lecture
- Make all **paragraphs** with **internal style** with **color darkviolet**

## Lists exercise

The lists contain at least two elements. The most common types are ordered and unordered lists.

## Tables exercise

Tables is a structured set of data made up of rows and columns.

## Forms exercise

An HTML Form is made of one or more widgets. Those widgets can be text fields (single line or multiline), select boxes, buttons, checkboxes, or radio buttons.

# Solution: Paragraph stage upgrade

- Make Internal style in the HTML document and style all **<p>**

```
<head>
  <meta charset="UTF-8">
  <title>CSS-Lab</title>
  <style>
    p{
      color: darkviolet;
    }
  </style>
</head>
</html>
```

- You can change the look of an entire website by changing just **one file!**
- Time saver when you want to have the same element in all of your pages.
- Each page must include a **reference** to the **external style sheet file** inside the **<link>** element. The **<link>** element goes inside the **<head>** section.

# External style example

## using-css.html

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet"
type="text/css" href="styles.css">
  </head>
  <body id="content">
    <p>This is a <span class="special">
special beer</span> for <span class=
"special">special drinkers</span>.</p>
  </body>
</html>
```

## styles.css

```
.special {
  font-style: italic;
  font-weight: bold;
  color: blue;
}

#content {
  background: #EEE;
}

p {
  font-size: 24pt;
}
```

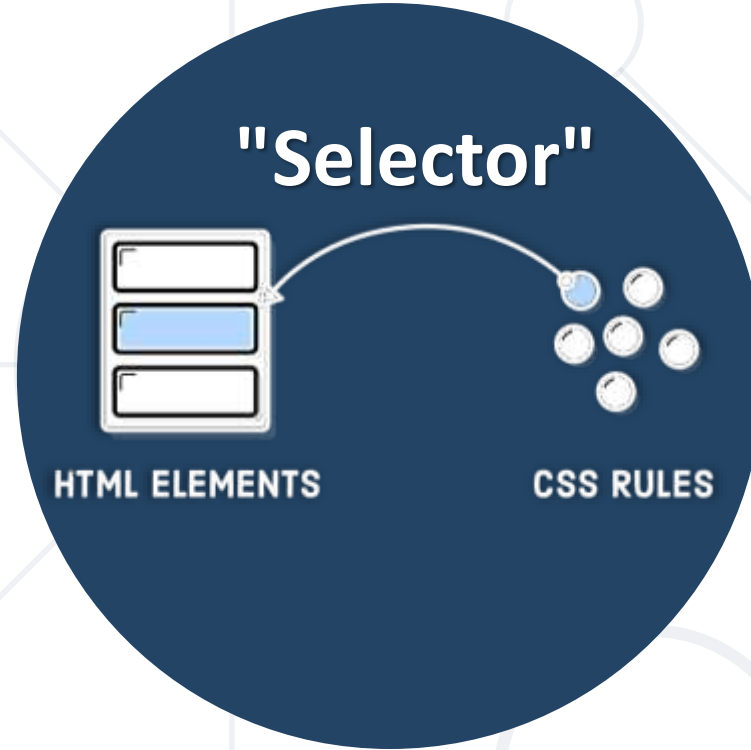
# Problem: Code refactor

- Delete all the styles you have made
- Open the styles from the HTML lecture
- Link the CSS file into your HTML file
- Add in the CSS file, the styles you have deleted

```
<head>
  <meta charset="UTF-8">
  <title>CSS-Lab</title>
  <link rel="stylesheet"
type="text/css" href="style.css"/>
</head>
```

```
h1{
  text-align: center;
  color: blue;
}
p{
  // TODO }
```





# CSS Selectors

Class, Id, Tag, "\*"

# Selectors

- In CSS, **selectors** are patterns used to select the element(s) you want to style.
- Types:
  - **.class** – selects a group of elements with the specified class
  - **#id** – selects a unique element by ID
  - **tag** – selects all specified tags
  - **\*** – selects everything



# Example

## using-css.html

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet"
type="text/css" href="styles.css">
  </head>
  <body id="content">
    <p>This is a <span class="special">
special beer</span> for <span class=
"special">special drinkers</span>.</p>
  </body>
</html>
```



## styles.css

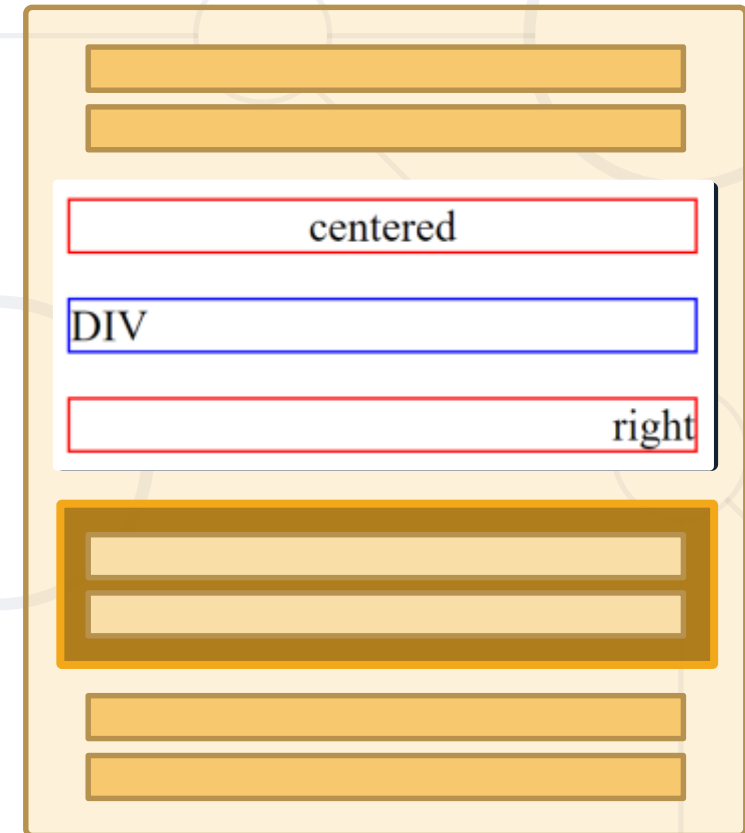
```
#content {
  background: #EEE;
}
.special {
  font-style: italic;
  font-weight: bold;
  color: blue;
}
p {
  font-size: 24pt;
}
* {
  border: 0;
}
```



# **Block elements**

## **Block, Inline, Inline-Block**

- A **block-level** element always starts on a **new line** and takes up the full width available.
- **<div>** and **<p>** are **block** elements
  - Fill the entire container width
  - Stack vertically one after another



# Problem: List/Table/Form stage upgrade

- Use your HTML code from the previous lecture
- Make all list elements (`<li>`) with **solid border** in **yellow color**

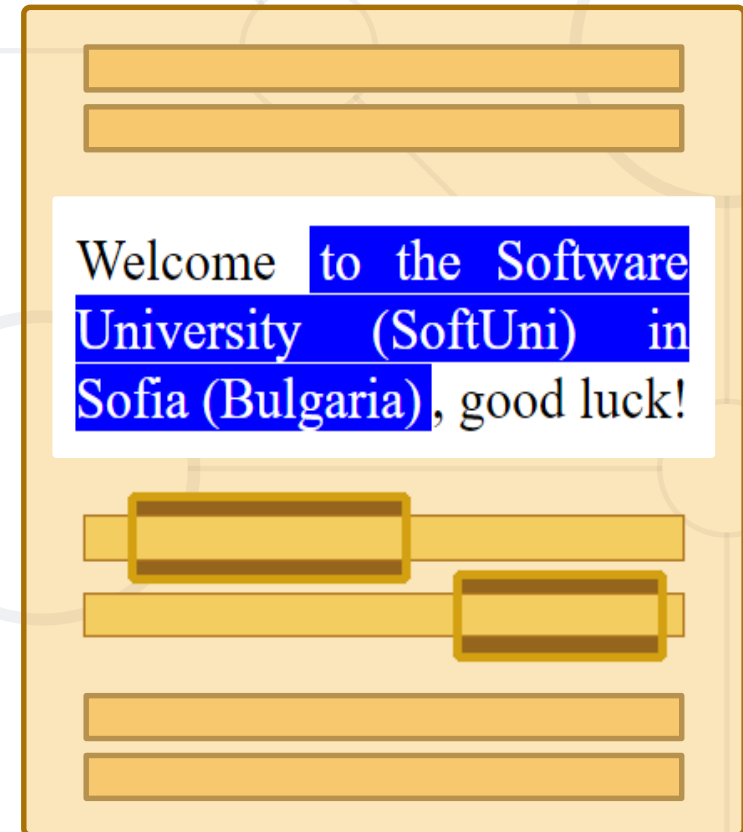
## Lists exercise

The lists contain at least two elements. The most common types are ordered and unordered lists.

### Example

1. Apple
  - Fruit
  - It's Healthy
2. Cucumber
  - Vegetable
  - The perfect salad ingredient
3. Bonster
  - Energy drink
  - Contains coffee

- An **inline** element does **not start on a new line** and only takes up as much width as necessary.
- **<span>** is **inline** element
  - Its shape is not always rectangular
  - Can be split across multiple lines



# Problem: List/Table/Form stage upgrade

- Use your HTML code from the previous lecture
- Make all your **spans** with **dotted border** in **darkcyan color**

## Lists exercise

The **lists** contain at least two elements.  
The most common types are ordered and unordered **lists**.

## Tables exercise

**Tables** is a structured set of data made up of rows and columns.

## Forms exercise

An HTML **Form** is made of one or more widgets. Those widgets can be text fields (single line or multiline), select boxes,



# Inline-Block elements

- **Inline-block** elements are similar to inline elements, except they can have padding and margins added on all four sides.
- Rectangles arranged one after another
- Just **like words** in a **sentence**

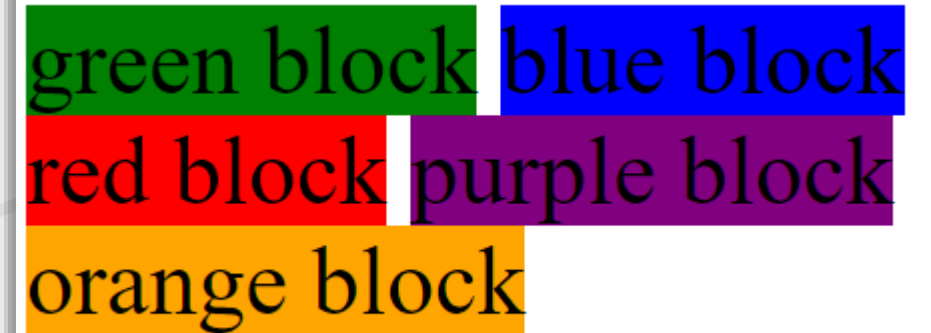


# Problem: Color Blocks

- Use the following HTML

```
<body>  
<div id="green">green block</div>  
<div id="blue">blue block</div>  
<div id="red">red block</div>  
<div id="purple">purple block</div>  
<div id="orange">orange block</div>  
</body>
```

- Example:

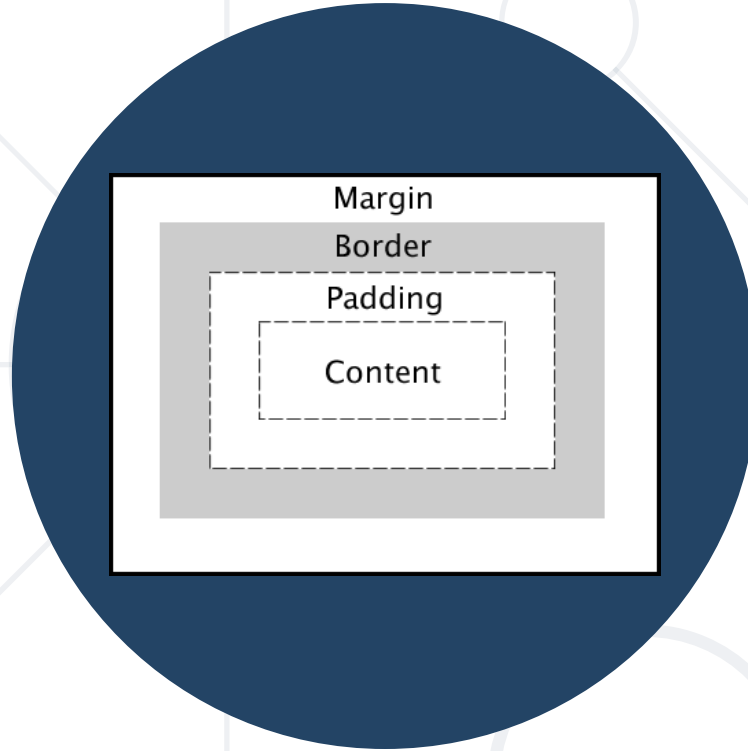


green block blue block  
red block purple block  
orange block

# Solution: Color Blocks

- Make external CSS file, link it in your HTML **<head>**
- Make all **<div>** with **display: inline-block**
- Call every **<div>** by its **id** name and set its **background-color** to its name's corresponding color

```
div{  
    display: inline-block;  
}  
#green{  
    background-color: green;  
} // TODO make the rest
```

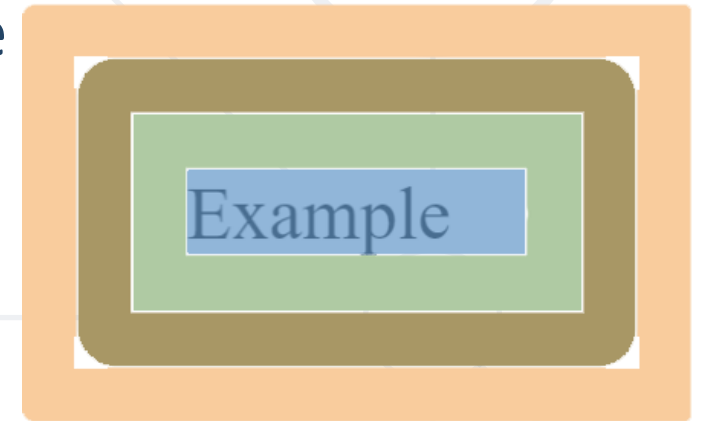


# Box Model

## Margins, Border, Padding, Content

# Box model

- The CSS box model is essentially a box that **wraps** around every HTML element. It consists of:
  - **Margins** - Area outside the border. It is transparent.
  - **Borders** - Border that goes around the padding and content
  - **Padding** - Area around the content. It is transparent.
  - **Content** - Where text and images appear.



top

right

bottom

left

```
<div class="margin" style="margin: 20px 20px 20px 20px">
  <div class="border" style="border: solid red">
    <div class="padding" style="padding: 10px 10px">
      <div>My content is here</div>
    </div>
  </div>
</div>
```

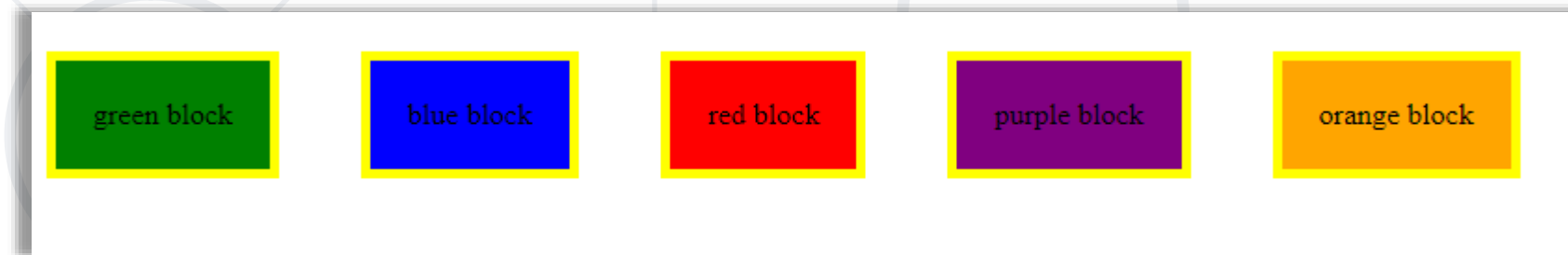
Top and  
bottom

Left and  
right

- The CSS border properties allow you to specify the **style**, **width**, and **color** of an element's border.

# Problem: Color Blocks upgrade

- Use the HTML and CSS from the previous problem
- Make all `<div>` **padding 20px** on each side, refresh the browser
- Set **solid, thick border** with **yellow color**, refresh the browser
- Now set **margin** with **20px** for each side, refresh the browser



# Solution: Color Blocks upgrade

- We have the `<div>` display and color already
- Set the **padding** and **margin**. In this case we have **20px** on **each side**, so we can just set them to a **single number** and it will be applied for **all sides**
- If you forget to tell what **kind of border** you want (in this case "solid"), it will **not** be displayed at all

```
div{  
  display: inline-block;  
  padding: 20px;  
  border: solid thick yellow;  
  margin: 20px;  
}
```

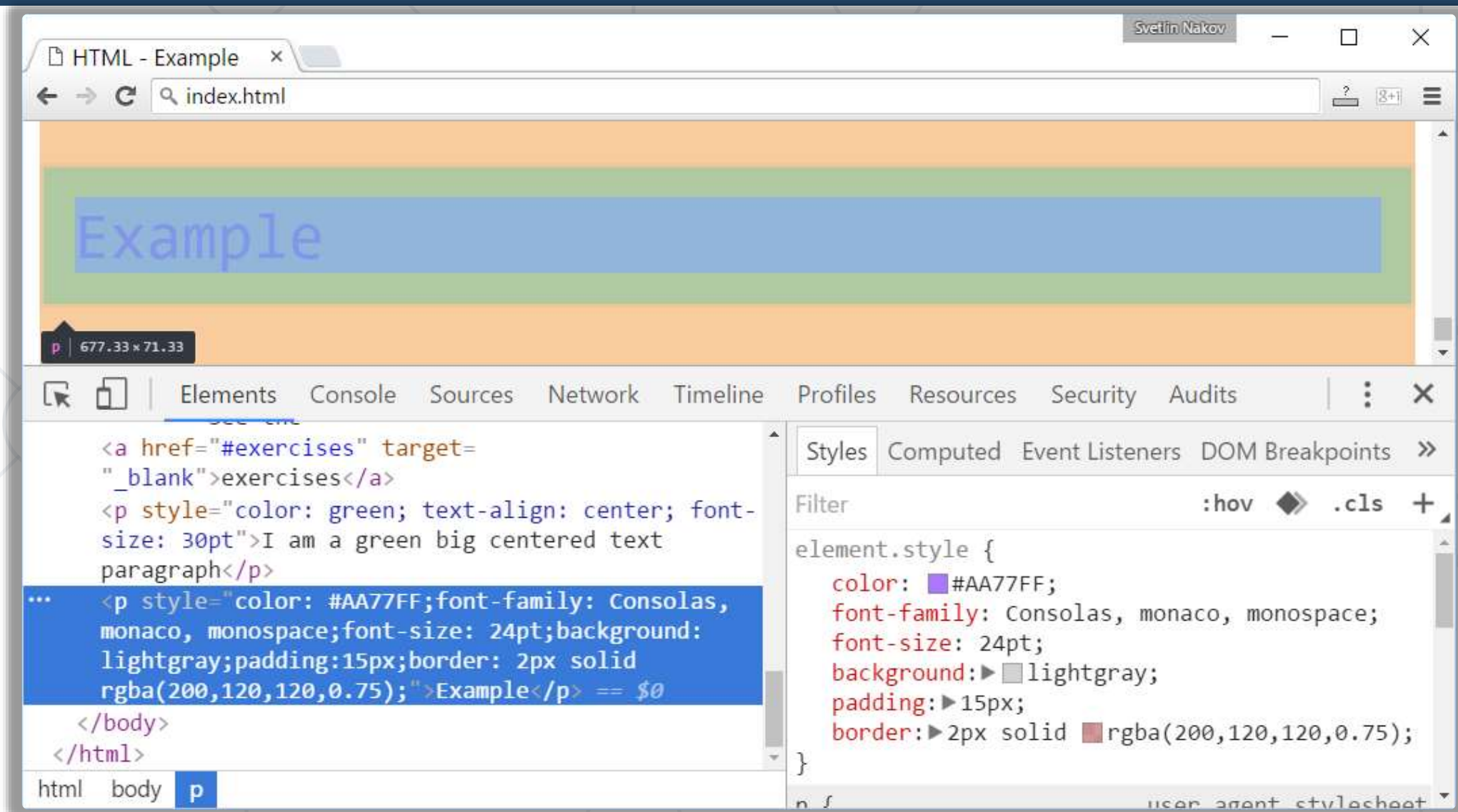


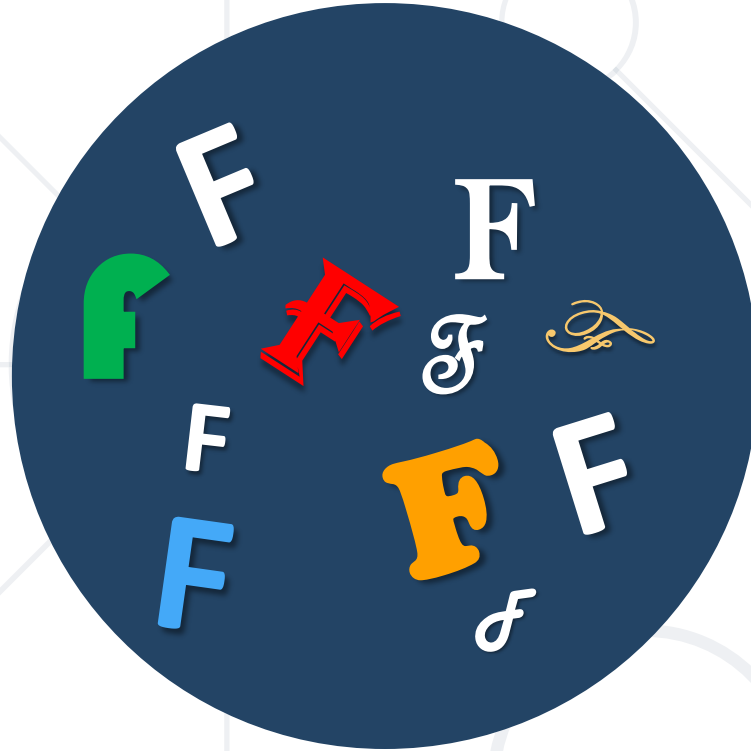
 F12 Ctrl+Shift+I Ctrl+Shift+K

# Dev tools

- We will view the following more **important tabs** in dev tools:
  - **Elements** – show our HTML structure
  - **Styles** – show our CSS code for the current page
- We can **change** our HTML and CSS code by double clicking the element.
- By **hovering** over the elements we can see their margins, padding and content.

# Google Chrome dev tools example





# Fonts

## Font-family, size, colors

- In CSS, there are two types of font family **names**:
  - **generic family** - a group of font families with a similar look
  - **font family** - a specific font family

Generic family	Font family	Description
Serif	Times New Roman Georgia	Have small lines at the ends on some characters
Sans-serif	Arial Verdana	Do not have the lines at the ends of characters
Monospace	Courier New Lucida Console	All characters have the same width

- The font family of a text is set with the font-family **property**.

```
p {  
  font-family: "Times New Roman", Times, serif;  
}
```



- The font-family property should hold **several font names** as a "fallback" system.
- End with a **generic family**, to let the browser pick a similar font in the generic family, if no other fonts are available.

# Problem: Cool Fonts

- Use the code from the previous problem
- Go to <https://fonts.google.com/>
- Pick a font of your preference
- Click on the "+" in the top-right corner
- Click here
- Copy the link in your HTML **<head>**
- Copy the font family in your **<div>**



1 Family Selected

STANDARD @IMPORT

```
<link href="https://fonts.googleapis.com/css?family=Indie+Flower" rel="stylesheet">
```

Specify in CSS

Use the following CSS rules to specify these families:

```
font-family: 'Indie Flower', cursive;
```

# Solution: Cool Fonts

- HTML file

```
<head>
  <meta charset="UTF-8">
  <title>CSS-Lab</title>
  <link
href="https://fonts.googleapis.com
/css?family=Indie+Flower"
rel="stylesheet">
  <link rel="stylesheet"
type="text/css" href="style.css"/>
</head>
```

- CSS file

```
div{
  display: inline-block;
  color: white;
  padding: 20px;
  border: solid thick yellow;
  margin: 20px;
  font-family: 'Indie
Flower', cursive;
}
```

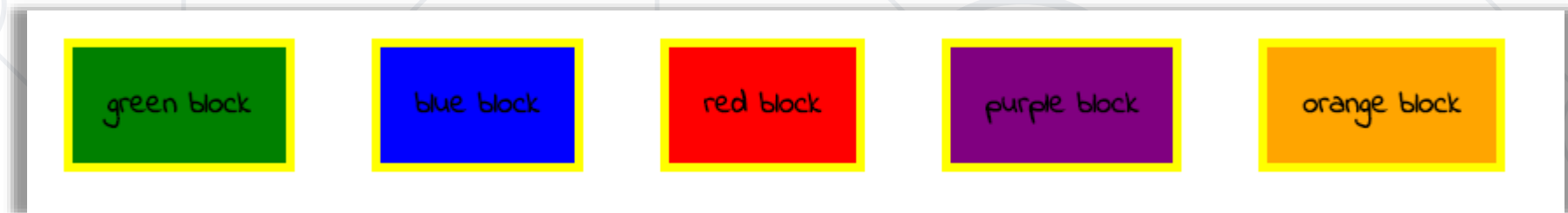


- The font-size property sets the **size** of the **text**.
- The font-size value can be an **absolute**, or **relative** size.

Absolute size	Relative size
Sets the text to a specified size	Sets the size relative to surrounding elements
The user cannot change the text size in all browsers	The user can change the text size in browsers
Useful when the physical size of the output is known	

# Problem: Cool Fonts upgrade

- Use the code from the previous problem
- Set **large font-size**
- Set **bold font-weight**

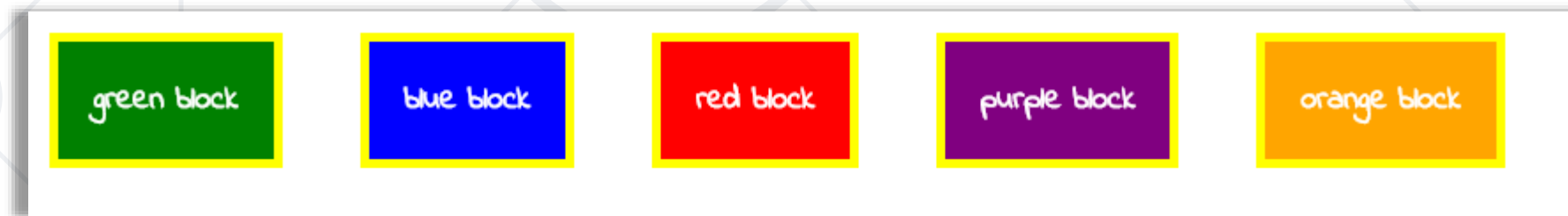



- The **color property** is used to set the color of the text.
- The color is specified by:
  - a color name - like "**red**"
  - a HEX value - like "**#ff0000**"
  - an RGB value - like "**rgb(255,0,0)**"

```
h1 {  
    color: red;  
    color: #FF0000;  
    color: rgb(255, 0, 0);  
}
```

# Problem: Cool Fonts upgrade

- Use the code from the previous problem
- In your `<div>` set **white color**



A dark blue circle containing the CSS comment syntax `/**/` in white. The background features a light gray geometric pattern of lines and circles.

`/**/`

**Comments in CSS**

# Writing comments

- The text inside the `/* */` marks is CSS **comment**. This allows you to enter notes into CSS that will not be interpreted.
- The **first** instance of `*/` that follows an instance of `/*` closes the comment. Can be multiline.
- Example:

```
body{  
    background-color: blue; /* maybe lighter color? */  
}
```



- What is CSS?
- CSS in HTML
- Selectors
- Fonts
- Box Model
- Block Elements



# Questions?



**SoftUni**



**Software  
University**



**SoftUni  
Svetlina**



**SoftUni  
Creative**



**SoftUni  
Digital**



**SoftUni  
Foundation**



**SoftUni  
Kids**



# SoftUni Diamond Partners



**XS**software



**SBTech**  
*we know sports*



telenor



**SoftwareGroup**  
*doing it right*

**NETPEAK**



**SmartIT**



**Postbank**

Решения за твоето утре

**SUPER  
HOSTING  
.BG**

**INDEAVR**

Serving the high achievers



**INFRAGISTICS®**

**LIEBHERR**



aeternity



**codexio**

# SoftUni Organizational Partners



# Trainings @ Software University (SoftUni)

- Software University – High-Quality Education and Employment Opportunities
  - [softuni.bg](http://softuni.bg)
- Software University Foundation
  - <http://softuni.foundation/>
- Software University @ Facebook
  - [facebook.com/SoftwareUniversity](https://facebook.com/SoftwareUniversity)
- Software University Forums
  - [forum.softuni.bg](http://forum.softuni.bg)



- This course (slides, examples, demos, videos, homework, etc.) is licensed under the "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International" license

