**Begin**

Number : D, E, C.

Print (Evro)

Input E

C = E \* 117

Print (Iznos)

Output C

**IF**

Begin

Ocena 1, ocean 2, ocean 3, prosek

Print (ocean 1)

Print (ocean 2)

Print (ocena 3)

Input ocena 1, ocena 2, ocena 3

Prosek = (ocena 1 + ocena 2 + ocena 3) / 3

If

Prosek > 60 than print Successful

Else

Print failed

END If

End.

**Begin**

Var Mile, kilometer, result

Print (upisi\_iznos)

Input iznos

Print (unesi mernu jedinizu - milje ili kilometre)

Input (merna\_jedinica)

Print (unesi mernu jedinicu u koju konvertujes)

Input ciljana\_merna\_jedinica

If merna\_jedinica = mile and ciljana\_merna\_jedinica = kilometer than

Result = iznos \* 1,0303

Print result

If merna\_jednica = kilometer and ciljana\_merna\_jedinica mile than

Rezultat = iznos \* 0.223

Print rezultat

End.

Convert miles to kilometers or kilometeres to miles

**Begin**

Var numbers (‘amount’, ‘press’, result’)

Print (“press 1 if you want to convert miles to kilometers or press 2 if you want to convert kilometers to miles”)

Input Press

Print (amount)

Input amount

If press = 1 than

Result = amount \* 1,041

Else if press = 2 than

Result = amount \* 0.324

Else Print (something went wrong)

End if.

Print Result

End.

**Arrays, lists - kolone, liste**

Var studenti = [ ‘Maja’ , ‘Mirko , ‘Stefan’ ]

For i=0 ; i<3 ; i=i++

Print studenti [0]

Print studenti [1]

Print studenti [2]

For i=0 ; i<3 ; i=i++

Print I (u ovom slucaju ispisuje samo brojeve(menja brojeve))

Var studenti = [ ‘Maja’ , ‘Mirko , ‘Stefan’ ]

For i=0 ; i<3; i=i++

Print studenti **[i]** (sada ce ispisati imena studenata)

Bilo bi –

Var studenti = [ ‘Maja’ , ‘Mirko , ‘Stefan’ ]

For i=0 ; i<3; i=i++

Print studenti [i]

Var studenti = [ ‘Maja’ , ‘Mirko , ‘Stefan’ , ‘Suzana’ , ‘Djordje’ ]

For i=0 ; **i<studenti.length** ; i=i++ (.length oznacava da se ispisuje do kraja, beskonacno)

Print studenti **[i]**

**Zadatak da se ispise ime jednog od svih studenata**

Begin

Var Studenti [‘Maja’ , ‘Anastazija’ , ‘Djordje’]

Var student\_pronadjen = false ;

Print ‘Unesi ime’

Input ime

For var i=0 ; i< studenti.length ; i = i++

If ime == student[i]

Print ‘ime’

? End program (ako ovo dodamo, ovde se zavrsava niz FOR I ponavljanja) ?

Student\_pronadjen == true

End for

If student\_pronadjen == false

Print ‘ime nije na spisku’

End

Begin

Var – broj1, broj2, broj3, rezultat

Print (upisi\_broj1)

Input broj1

Print (upisi\_broj2)

Input broj2

Print (upisi\_broj3)

Input broj3

Rezultat = broj1 + broj2 + broj3 / 3

Print rezultat

End.

**UVOD – HTML / CSS**

* **ARPANET** - The name of the first computer network (1969)
* **INTERNET** - collection of nodes (computers, routers, networks / networks based on the tci / ip communications protocol)
* **INTERNET** **SERVERS** (Email, WWW)
* **IP** ?
* **WWW** (father of www was Tim Berners-Lee, the universe of network-accessible information, an embodiment of human knowledge)
* **WWW** (technical definition / all the users resources on the internet using HTTP)
* **HTTP** – when clicking a link, you are transferring an URL to you browser, your browser knows which server to contact and what file to ask for.
* **URL** – fundamental network, used to specify addresses on WWW.
* **WEB SERVERS** – always connected to the internet, every site on it has unique address made up of series of four numbers between 0 and 256 separated by periods.
* **WEB SERVERS** – when you register a web address (domain (your site)), you have to specify the IP address of the web server that will host the site.
* **DOMAIN** – main part of your web site – www.mysite.com – mysite.com is domain.
* **SUBDOMAIN** – further dinision of your domain
* **DOMAIN EXTENSIONS** - .com / .net / .gov
* **WEBSITES** - a group of interlinked and well -structured web pages on the same domain
* **WEB BROWSERS** – Microsoft edge, Mozilla, chrome
* **HOSTING** - ?
* **SEO** – search engine optimization process of improving the quality and quantity of website

**HTML – hyper text markup language**

* Used to format the text, you can add instructions to tell the computer how to display the text
* <please do this> / </end please do this>
* **Tools** for HTML code – Visual Studio, Web flow
* **HTML** consists of an opening tag, content and closing tag.
* <p lang =”fr”> (lang = attribute name, fr = attribute value)
* **STRUCTURE HTML**

<html>

<head>

<title> page title here </title>

<head>

<body>

<body>

</html>

* **CTRL** a = MARKING
* **CTRL** C = COPY
* **CTRL** V = PASTE
* **CTRL** Z = UNDO
* **CTRL** S = SAVE
* **CTRL** / = Comment

**COMENTS** (<! -- comments -->) shortcut = ctrl /

* With comments you can place notifications and reminders in your HTML code:

**Headings**

**<h1>**This is heading 1</h1>

* <h1 style="color:red;">Heading 1</h1>

<h2>This is heading 2</h2>  
<h3>This is heading 3</h3>

**Paragraphs**

**<p>**This is a paragraph</p>

**<hr> (**thematic break**)**

<p>This is some text</p>  
<hr> (Ovo u tekstu ostaje prazno ili oznaceno kao horizontalna linija, nakon hr moze se uvoditi novi H ili P.)

**<br> (**a new line without starting a new paragraph**)**

<p>This is<br>a paragraph<br>with line breaks</p>

**<pre>**

<p>Ispod je pesma koja je izdeljena (pre se najvise koristi za to):</p>

<pre>

My Bonnie lies over the ocean.

My Bonnie lies over the sea.

My Bonnie lies over the ocean.

Oh, bring back my Bonnie to me.

</pre>

**Links**

<**a** href="https://www.w3schools.com">This is a link</a>

The link's destination is specified in the href attribute.

Atribut TARGET (\_blank (opens in new window) or \_self (opens in the same window))

<a href="https://www.youtube.com/" target="\_blank"> YOUTUBE </a>

**ANCHORS**

Stavlja se **id** u neki tag I na to se vraca kada se klikne na “back”

<body id="top">Ovde se pise kod </body>

<a href="#top">Back to top</a>

**Mailto**

* <a href="mailto:jon@example.org">Email Jon</a>

**Images**

<**img**src="w3schools.jpg" alt="error" width="104" height="142">

**Empty HTML Elements**

HTML elements with no content are called empty elements.

<p>This is a <br> paragraph with a line break. </p>

**Lists**

* **Ordered List**

<ol>  
  <li>Coffee</li>  
</ol>

* **Unorderd list**

<ul>  
  <li>Coffee</li>  
</ul>

* **Description list**

<dl>  
  <dt>Coffee</dt>  
   <dd>- black hot drink</dd>  
</dl>

* **Nested lists**

<ul>  
  <li>Coffee</li>  
 <li>Milk</li>  
 <li>Water

<ul>  
 <li> Plain </li>  
 <li> Mineral </li>

</ul>

</li>  
</ul>

* ***liste-style-type : …***

**Text éléments**

**<title>** (kad se predje klikom preko reci, pisace dodatni tekst)

**<b>**Bold </b> **BOLD**

**<i>**Italic</i> *ITALIC*

**<u>**Underline</u> UNDERLINE

**Strong** <p><strong>This text is important! </strong></p>

**EM (emphasis)** <p><em>This text is emphasized. </em></p> (This text is emphasized.)

**Small** <p><small>This is some smaller text. </small></p>

**Mark** <p>Do not forget to buy <mark>milk</mark> today. </p> (Do not forget to buy milk today.)

**Del** <p>My favorite color is <del>blue</del> red.</p> (My favorite color is  red.)

**Ins** <p>My favorite color is <del>blue</del> <ins>red</ins>. </p> (My favorite color is  red.)

**SUB** <p>This is <sub>subscripted</sub> text. </p> (This is subscripted text.)

**SUP** <p>This is <sup>superscripted</sup> text. </p> (This is superscripted text.)

**Blockquote**  <blockquote cite="http://www.worldwildlife.org/who/index.html">

For 60 years, WWF has worked to help people and nature thrive.

</blockquote>

<blockquote onclick="alert('Hello!')"> <p>Hello!</p> </blockquote>

**<q>** <p>WWF's goal is to: <q>Build a future where people live in harmony with nature. </q></p>

WWF's goal is to: “Build a future where people live in harmony with nature. “

**<abbr>** <p>The <abbr title="World Health Organization">WHO</abbr> was founded in 1948.</p>

The WHO was founded in 1948.

**<address>**

<address>Written by John Doe.<br>

Visit us at:<br>

Example.com<br>

Box 564, Disneyland<br>

USA

</address>

*Written by John Doe.  
Visit us at:  
Example.com  
Box 564, Disneyland  
USA*

***<cite*>** <p><cite>The Scream</cite> by Edvard Munch. Painted in 1893. </p>

**<bdo>** <bdo dir="rtl">This text will be written from right to left</bdo>

**TABELE**

 <**table>** - Specifies a table

 <**caption>** - Specifies a table caption

 <**thead>** - Specifies table header row

 <**tbody>** - Specifies table body rows

 <**tfoot>** - Specifies table footer rows

 <**tr>** - Specifies a table row

 <**th>** - Specifies a table header cell

 <**td>** - Specifies table data cell

<table>

<thead>

<tr>

<th colspan="2">The table header</th>

</tr>

</thead>

<tbody>

<tr>

<td>The table body</td>

<td>with two columns</td>

</tr>

</tbody>

<tfoot> </tfoot>

</table>

 <**table border="n">** - ordedjuje debljinu ivice

 <**table cellpadding="n">** - velicinu celije

 <**table cellspacing="n">** - razmak izmedju celija

**FORMS**

**Dodavanje teksta**

* Text input
* Password input
* Text area (multi-line)

**Davanje izbora**

* Radio button
* Checkboxes
* Drop-down boxes

**Upload**

* File upload

**Submitting forms**

* Submit
* Reset

**Posebni elementi**

* Date
* Email
* url
* search

HTML attributes

* Methods - <https://www.w3schools.com/tags/ref_httpmethods.asp>
* **Get** – (limited, can be bookmarked, shown in URL)
* **POST** – (for sensitive personal info, has no limits, can’t be bookmarked)
* **Target** -
* Action- The action attribute defines the action to be performed when the form is submitted. (URL na koji ces biti poslat kada kliknes submit na kraju ispunjavanja obrasca). <form action="/action\_page.php">

**Text**

* <form action="some site" method="POST">
* <label for="username">Username:</label>
* <input type="text" name="username" id="username" />
* </form>

(u input se moze dodati ***value*** ako zelimo da u polju nesto vec stoji kao primer )

**Password**

* <form action="" method="POST">
* <label for="username">Username:</label>
* <input type="text" name="username" id="username" />
* <label for="pwd">Password:</label>
* <input type="password" name="pwd" id="pwd" />
* </form>

**Text area**

* <form action="" method="POST">
* <label for="comments">What did you think of this gig? </label>
* <textarea id="comments" name="comments" cols="20" rows="4">
* Enter your comments...
* </textarea>
* </form>

(name atribut je neophodan da bi se taj tekst uneo nakon sto se potvrdi forma)

**Checkbox**

* <form action="" method="POST">

<p>Cekiraj sta zelis</p>

* <input type="checkbox" name="service" value="itunes" ***checked***="checked" id="itunes" />
* <label for="itunes">iTunes</label>
* <input type="checkbox" name="service" value="lastfm" id="lastfm" />
* <label for="lastfm">Last.fm</label>
* </form>

(***checked*** atribut sluzi da prikaze kao primer da je nsto vec cekirano)

**Drop down / select list box**

* <form action="" method="POST">
* <label for="devices">What device do you listen to music on?</label>
* <select name="devices" id="devices">
* <option value="ipod" ***selected***>iPod</option>
* <option value="radio">Radio</option>
* <option value="computer">Computer</option>
* </select>
* </form>

(***selected*** atribut sluzi da prikaze kao primer da je nsto vec cekirano)

**Multiple select box**

* <form action="" method="POST">
* <label for="instruments">Do you play any of the following instruments?</label>
* <select name="instruments" id="instruments" size="3" ***multiple***="multiple">
* <option value="guitar" selected>Guitar</option>
* <option value="drums">Drums</option>
* <option value="keyboard" selected="***selected***">Keyboard</option>
* <option value="bass">Bass</option>
* </select>
* </form>

**File input box**

* <form action="" method="POST">
* <label for="user-song">Upload your song in MP3 format:</label>
* <input type="file" name="user-song" id="user-song" />
* </form>

**Submit and reset buttons**

* <form action="" method="POST">
* <input type="submit" value="Subscribe" />
* <input type="reset" value="Reset Form" />
* </form>

**Radio button**

* <form action="" method="POST">
* <p>Please select your favorite genre:</p>
* <input type="radio" name="genre" value="rock" id="rock" />
* <label for="rock">Rock</label>
* <input type="radio" name="genre" value="pop" id="pop" />
* <label for="pop">Pop</label>
* <input type="radio" name="genre" value="jazz" id="jazz" />
* <label for="jazz">Jazz</label>
* </form>

**Button / Hidden controls**

* <form action="button.php">
* <button>
* <img src="images/icon-plus.svg" alt="plus-sign"
* width="10" /> Add
* </button>
* <input type="hidden" name="product-code"
* value="P1234" />
* </form>

**<button id="button" type="button"**

**onclick="alert('Button was clicked!');">My Button</button>**

* (pojavljuje se dugme “my button”)
* <p>An invisible button:</p>
* <button type="button" hidden>My Button</button>
* An invisible button:

**Grouping form elements**

* <fieldset>
* <legend>Contact details</legend>
* <label for="email">Email:</label>
* <input type="text" name="email" id="email" />
* <label for="mobile">Mobile:</label>
* <input type="text" name="mobile" id="mobile" />
* <label for="telephone">Telephone:</label>
* <input type="text" name="telephone" id="telephone" />
* </fieldset>

**Validation**

* **(specific elements – email, date, url, search)**

<input type="email" name="email" ***required placeholder***="Enter a valid email address">

* Atribut ***required*** se stavlja !
* ***Placeholder*** atribut (ono sto pise dok se ne klikne na to polje)

input type="url" name="website" ***required pattern***="https? ://.+">

VALIDATION - https://www.w3schools.com/js/js\_validation.asp

**Number and range**

Age: <input type="number" size="6" name="age" min="18" max="99" value="21"><br>

Satisfaction: <input type="range" size="2" name="satisfaction" min="1" max="5" value="3">

![A screenshot of a video

Description automatically generated]()

![A computer screen with several squares

Description automatically generated with medium confidence]()

* **Header** - <header>……</header> = company information (Moze biti unutar tagova article ili section, u tom slucaju je poseban deo tog dela)
* **Footer** - <footer>……</footer> = contact information, copyright notices.
* **Navigation** - <nav>……</nav> = (site navigation)
* **Div** - <div>……</div> = (different parts of the page)
* **Aside** - <aside>……</aside> = (dopunjuje main deo i sa tim je indirektno povezan, najcesce opisuje deo u kom je nested).
* **Section** - <section>……</section> = (maps or headlines, summaries, it’s considered best practice to begin each section with a heading.)
* **Article** - <article>……</article> = (forum post, blog post, news story).
* **Main** - <main>……</main> = (treba da postoji samo jedan main deo na stranici I ne treba da se ugnezdava u ostale tagove).
* **Span** - <span>……</span> = (inline document – u liniji, istoj recenici).

***AUDIO, VIDEO, IFRAMES, META TAGS***

* **Video** <video> src=”.” type= (tip videa) poster= (odredjuje sliku koja se pojavljuje dok se video skida ili dok se ne pusti) width height controls (pretrazivac treba da obezbedi svoje kontrole za pustanje videa) autoplay loop.</video>

**Preload** atribut

* NONE (nece se ucitati video dok korisnik ne klikne paly)
* AUTO (pretrazivac ce skinuti video kada se ucita stranica)
* METADATA (pretrazivac ce skupiti info, first frame, size,track list)
* **AUDIO** – the same without poster.
* **IFRAME** <iframe src=… height width> da ubaci drugu stranicu kao posebnu na aktuelnu stranicu, obicno su to google maps. </iframe>

**META** – selfclosing tag, inside <head>. Vrednosti su

* description (opis stranice),
* keywords (odvojene reci koje se najcisce koriste za tu vrstu stranice),
* robots ( vrednost noindex (pretrazivac treba da dodaje ovu straanicu svojim rezultatima pretrage), nofollow (ne treba)).
* Author (autor stranice)
* Pragma (Ово спречава претраживач да кешира страницу. (Односно, чувате га локално да бисте уштедели време приликом преузимања приликом наредних посета.)
* Expires (pretrazivac cesto kesira stranicu, ovo se koristi da oznaci kada bi stranica trebalo da istekne I da ne bude vise kesirana).

<head>

<title>Information About Your Pages</title>

<meta name="description"

content="An Essay on Installation Art" />

<meta name="keywords"

content="installation, art, opinion" />

<meta name="robots"

content="nofollow" />

<meta http-equiv="author"

content="Jon Duckett" />

<meta http-equiv="pragma"

content="no-cache" />

<meta http-equiv="expires"

content="Fri, 04 Apr 2014 23:59:59 GMT" />

</head>

**List**

* <form action="/action\_page.php">
* <input list="browsers" name="browser">
* <datalist id="browsers">
* <option value="Edge">
* <option value="Firefox">
* <option value="Chrome">
* <option value="Opera">
* <option value="Safari">
* </datalist>
* <input type="submit" value="Submit">
* </form>

**CSS - Cascading Style Sheets**

Sectors:

* **id** <p id="nesto">Hello World! </p> - a u HEAD se pise:
* #nesto {text-align: center; color: red;}
* **Class** <div class="nesto">This heading will not be affected</div>  
  - div.nesto { text-align: center; color: red;}

(Klasi dodajemo vrednosti, tu vrednost ubacujemo u HEAD I njoj pisemo CSS elemente. Svakom tagu mozemo dati drugaciju class). Mozemo imati iste vrednosti klase a da bi se razlikovao CSS stavljamo ispred center ime taga.

Moze se dodati IMPORTANT nakon svakog property i to oznacava da je bitnije od ostalih pravila.

* **\* {…} - groupise sve elemente na toj stranici**
* **Goupping h1, p, h3, div {…}**
* **More classes.**

<html>

<head>

<style>

p.center {

text-align: center;

color: red;

}

p.large {

font-size: 300%.

}

</style>

</head>

<body>

<h1 class="center">This heading will not be affected</h1>

<p class="center">This paragraph will be red.</p>

<p class="center large">This paragraph will be center-aligned.</p>

</body>

</html>

**External CSS**

<head>  
<link rel="stylesheet" href="mystyle.css">  
</head>

- If the internal style is defined before the link to the external style sheet, it will has priority.

 Some properties can be **inherited**

 Specify property on the parent element, they will apply to most child elements.

 Some properties are not **inherited**

 E.g. background-color or border properties

 Explicitly saying that some element will *inherit* values from their parent elements by using *inherit* for the value of the properties.

**Colors**:

* **rgb values** (rgb (red, green,blue) (100,100,90))
* **hex codes** (These are six-digit codes that represent the amount of red, green and blue in a color, preceded by a pound or hash # sign. For example: #ee3e80).
* **color names** (There are 147 predefined color names that are recognized by browsers. For example: DarkCyan).
* **Opacity** (can be atribut, used in **RGBA**, always between 0.0 and 1.0)
* **HSL** - Hue, Saturation, Lightness (background-color: **hsl** (0,0%,78%);)

**FONT**

* Otvara se sa font-family
* @font-face {

font-family: "Open Sans";

src: url("fonts/OpenSans.woff2") format("woff2");

}

**Type** : serif (linije na na kraju slova), sans-serif (bez linija), monospace (iste velicine karakteri).

**Weight** – light, medium, bold, black (velicina slova)

**Style** - normal, italic, oblique

**Stretch**- condensed, regular, extended (sirina fonta, izmedju slova)

**Font size**

* Absolute (xx-small, small, medium, large, xxx-large, x-large)
* Relative (larger, smaller od parent taga)
* Length ( distance, relative em, ex, ic, cap, absolute px, cm, mm, in)
* Percentage (%)
* **Weight** (zadebljanje fonta, normal, bold, lighter, bolder, ili u brojevima 100;)

**Text-transform**

* None, capitalize, uppercase, lowercase
* Text-decoration-style: solid, double, dotted, dashed, wavy
* Text-decoration-line: none, underline, overline, line-through, blink, underline overline, overline underline line-through
* Text-decoration-color: red, rbga, rgb, hsla.
* line-height CSS property sets the height of a line box.
* It's commonly used to set the distance between lines of text.
* **Letter-spacing** (razmak izmedju slova)
* **Word-spacing** (razmak izmedju reci)

**Text-align**

* Left, right, center,
* Justify, justify-all
* Start, end.
* Match-parent.

**Vertical-align**

* Baseline, sub, super, text-top, middle, top, bottom.

**Text-shadow**

* Offsets (X or Y) (indicates the distance to the top or bottom that the shadow should fall.)
* Blur radius (specifies the amount of blur that should be applied to the drop shadow.)
* Color

/\* offset-x | offset-y | blur-radius | color \*/

text-shadow: 1px 1px 2px black;

/\* color | offset-x | offset-y | blur-radius \*/

text-shadow: #fc0 1px 0 10px;

/\* offset-x | offset-y | color \*/

text-shadow: 5px 5px #558abb;

/\* color | offset-x | offset-y \*/

text-shadow: white 2px 5px;

/\* offset-x | offset-y \*/

text-shadow: 5px 10px;

**Styling links**

* : link (stil linka koji jos nije otvoren)

a: link {color: pink;}

* : visited (stil linka koji je otvoren)

a: visited {color: black;}

* : hover (kada se stavi mis preko)
* : active (kada se klikne na link)
* : focus (Focus occurs when a browser discovers that you are ready to interact with an element on the page.)

**BOX**

* Sastavljen je od:

1. Magin
2. Border
3. Padding
4. Content

**Display**

* block (div, h1, p, form, header, footer, section)
* inline (ne moze mu se dodati visina I sirina) (span, a, img)
* inline - block (formatiran je kao inline ali mu se moze dodati visina i sirina).
* none (sakriva taj element, ponasao se kao da ga ima ali se na stranici ne vidi). Isto se moze postici sa Visibility : hidden.
* list-item (ponasaju se elementi kao liste, li).

Moze se to kontrolisati sa FLEX vrednoscu

**Box-sizing**

* border-box (velicina celog box-a je 200 sa pedingom i borderom)
* content-box (velicina je 200 samo content-a plus peding i border)

**Margin**

* top, right, bottom, left. (velicina margin : 1px 2 px 3px 4px) ili svaki posebno.

**Padding**

* top, right, bottom, left. (velicina margin : 1px 2 px 3px 4px) ili svaki posebno.

**Border**

* top, right, bottom, left. (velicina margin : 1px 2 px 3px 4px) ili svaki posebno.
* Style (none, hidden, dotted,dashed, solid, double, grouve, ridge, inset, outset)
* Color (red, yellow,green, transparent) ili za svaku stranu posebno.

**Overflow**

* Hidden (sakriva ono sto ne staje u box)
* Scroll (dodaje scrollbar)
* Auto (dodaje scrollbar gde god treba, horizontalno I vertikalno)

**Position**

* Absolute (ne utice na polozaj ostalih elemenata, odnosi se na element u kom se nalazi) (<https://www.w3schools.com/html/tryit.asp?filename=tryhtml_css_borders>)
* Relative (ne utice na polozaj ostalih elemenata, odnosi se na tok dokumenta) (<https://www.w3schools.com/html/tryit.asp?filename=tryhtml_css_borders>)
* Fixed (sadrzalac fixed pozicije je pocetni element)

(<https://www.w3schools.com/html/tryit.asp?filename=tryhtml_css_borders>)

* Sticky (ostaje vidljiv i ako se skroluje) - https://www.w3schools.com/css/tryit.asp?filename=trycss\_position\_sticky

**Z-index** (ako je negativno, znaci da se pojavljuje iza drugog elementa, ako je pozitivno, onda je ispred drugog elementa. Ako ima vise elemenata preko kojih treba da se pojavljuje, stavlja se odredjeni broj (broj elemenata koje treba prekriti))

* (<https://www.w3schools.com/cssref/tryit.php?filename=trycss_zindex>)

**Float** – none, left, right, initial, inherit.

**Pseudo-elements:**

* Firste-line (da se doda poseban style na prvu liniju teksta) - p::first-line {}
* Firste-letter (za style prvog slova) ::firste-letter
* Before (sluzi da se ubaci neki content pre kontenta nekog elementa) ::before
* After (da se ubaci content nakog nekog kontenta) ::after

**p::after {**

**content: " - Remember this";**

**}**

* Selection (markira delove elementa koje je selektovao korisnik (color, background, cursor, outline) ) ::selection

**FLEX –** jednodimenzionalni prikaz

* **1. flex-direction** : (row, row-reverse : column, column-reverse) )
* flex-wrap : wrap, nowrap. Ako su zbir velicina item-a veca od glavnog diva, a stavimo wrap, onda ce se njihova velicina ispostovati a ostatak elemenata ce se prikazati u drugom recu inline. Ako stavimo nowrap, on ice se sabiti u jedan red da ispune velicinu glavnog diva. U kodu se moze pisati **flex-flow** (row wrap) (direction and wrap).
* Vrednost se moze ispisati skracenicom **flex-flow**
* Flex-grow (Koliko pozitivnog slobodnog prostora zauzima taj item. Kada mali divovi nemaju velicinu a glavni ima, onda se uz flex-grow ispunjava ceo glavni div. To se oznacava brojevima, 1,1,1,4,3. Broj oznacava koliko prostora ce zauzeti svaki od malih divova). Ako neki mali div ima velicinu, onda on se postavlja u toj velicini a ostali divovi se dele prema zadatoj proporciji.
* Flex-shrink (Koliko negativnog slobodnog prostora zauzima taj item) ?
* Flex-basis (Odredjuje velicinu item-a. moze biti auto, ili broj px). Tako se moze izdeliti glavni div na odredjene manje divove zadate velicine.
* Vrednost se ispisuje skracenicom u redosledu – grow, shrink, basis (flex: 2 2 10%;)
* Positive free space (kada je glavni div veci od divova u njemu, ostaje prazan prostor neispunjen)
* Negative free space (kada je glavni div manji od divova u njemu, mali divovi prelaze velicinu glavnog diva.)
* **2. Justify-content**: center, flex-start (default), flex-end, space-between. (horizontalno poravnjanje)
* **3. Align-items**: center, flex-start, flex-end, stretch (default), baseline (u sredini kao da je linija / <https://www.w3schools.com/css/tryit.asp?filename=trycss3_flexbox_align-items_baseline>) (vertikalno poravnjanje)
* **4. Align-content**: space-between, space-around, stretch (default), center, flex-start, flex-end.

**Flex items property**

* Order (<div class="flex-container">  
    <div style="order: 3">1</div>) </div>) (odredjuje u kom redosledu ce se pojaviti koji div).
* Flex-grow
* Flex-shrink
* Flex-basis
* Justify-self
* **Align-self** (odredjuje poravnjanje child elementa I preovladava u odnosu na align-item u glavnom elementu) center, flex-start, flex-end, stretch (default), baseline, auto - <https://www.w3schools.com/css/tryit.asp?filename=trycss3_flexbox_align-self_flex>

**Flex links :**

* https://stackoverflow.com/questions/32551291/in-css-flexbox-why-are-there-no-justify-items-and-justify-self-properties/33856609#33856609

**Flex Responsive**

* Koristi se da se odredi kako ce se sadrzaj pojavljivati na manjim ekranima, telefonima, tabletima…

@media (max-width: 800px) {

.flex-container {

flex-direction: column; }

Drugi nacin je da se u glavni element doda flex-wrap : wrap ; i da se u @media dodaju klase manjih divova kako bi se znalo da se na njih odnosi.

* <https://www.w3schools.com/css/css3_flexbox_responsive.asp>
* <https://www.w3schools.com/css/tryit.asp?filename=trycss3_flexbox_website2> / ceo sajt

**Transition (prvo je duration) -** <https://www.w3schools.com/css/css3_transitions.asp>

1. **transition-property**

background-color, background-position

border-color, border-width, border-spacing

left, rigth, top, bottom

color, opacity

font-size, font-weight, letter-spacing, word-spacing, line-height, text-indent

width, height, max-width, max-heaight, min-width, min-height

padding, margin

text-shadow

visibility

z-index

vertical-align

outline-color, outline-offset, outline-width

1. **transition-duration (npr 2s)**

* Koristi se da se izmeni tranzicija osnovnog stila. Koristi se :hover, sto oznacava da se tranzicija aplikuje kada se stavi mis na odredjeni element.
* <https://www.w3schools.com/css/css3_transitions.asp>

1. **div {transition-timing-fonction}** - Odredjuje brzinu efekta koji mogu biti

* ease - specifies a transition effect with a slow start, then fast, then end slowly (this is default)
* linear - specifies a transition effect with the same speed from start to end
* ease-in - specifies a transition effect with a slow start
* ease-out - specifies a transition effect with a slow end
* ease-in-out - specifies a transition effect with a slow start and end
* cubic-bezier(n,n,n,n) - lets you define your own values in a cubic-bezier function

1. **div {transition-delay}  -** odredjuje vreme nakon koga ce se aplikovati tranzicija. Odredjuje se u sekundama 1s.
2. **div {transform}** odredjuje kako ce se element transformisati (rotiranje npr). Potrebno je staviti u :hover koja transformacija ce biti uradjena a u transition se pise kojom brzinom.

* -webkit-transform: scale(1.5, 1.5);

**Transition vendor prefixes**

* -webkit- (chrome, safari, newer versions of opera)
* -moz- (firefox)
* -o- (old version of opera)
* -ms- (internet explorer, edge)
* -webkit-transition-duration: 1s; -

**Animation**

* 1. **animation-name**
* Prvo se odredjuje ime u **@keyframes** (@keyframes IME), a onda se u glavnim karakteristikama dodaju promene u snimaciji.

div

{

height: 100px;

background-color: red;

animation-name: example;

animation-duration: 4s;

}

@keyframes example {

from {background-color: red;}

to {background-color: yellow;}

}

* U sledecem primeru se vidi da se moze menjati vise boja :
  1. **animation-duration 4s ()**

div {

height: 100px;

background-color: red;

animation-name: example;

animation-duration: 4s;

}

@keyframes example {

0% {background-color: red;}

25% {background-color: yellow;}

50% {background-color: blue;}

100% {background-color: green;}

}

* U ovom primeru se vidi da se moze menjati i pozicija elementa

div {

height: 100px;

background-color: red;

position: relative;

animation-name: example;

animation-duration: 4s;

}

@keyframes example {

0% {background-color:red; left:0px; top:0px;}

25% {background-color:yellow; left:200px; top:0px;}

50% {background-color:blue; left:200px; top:200px;}

75% {background-color:green; left:0px; top:200px;}

100% {background-color:red; left:0px; top:0px;}

}

**3. animation-delay** (vreme za koje ce poceti izvrsenje animacije, mogu se pisati obicni brojevi ili negativni brojevi sto znaci da ce izgledati kao da se vec odvija animacija.)

**4. animation-iteration-count** (podesava koliko puta ce se izvrsiti animacija. U brojevima il infinite)

**5. animation-direction** (kako se pojavljuje. Moze biti : normal, reverse, alternate (brze), alternate-reverse).

**6. animation-timing-function** (isto kao kod tranzicije)

**7. animation-fill-mode** (prikacuje kako izgleda element kada animacija nije igrana)

***Zapazanja***

1. ***Odraditi : prezentacija 20 I cas 11. RESPONSIVE (prezentacija 16 da se negde nauci)***
2. **Line-height** – odvaja jedan red teksta od drugog. Moze se koristiti za pozicioniranje teksta, vertikalno. (div height 200px ; p line-height200px = tekst na sredini).
3. **Clear** – odredjuje sta ce se desiti sa elementom pored plutajuceg elementa. Sto znaci da jedan mora biti FLOAT a drugom dodajemo CLEAR.