Računalniški praktikum in Osnove računalništva

8. CSS

CSS

- **Cascading Style Sheet (CSS)** je bil ustvarjen leta 1996.
- CSS je prezentacijski (slogovni) jezik.
- Spoznali smo HTML, ki določe strukturo spletne strani z uporabo oznak.
- CSS pa omogoča oblikovanje sloga in videza spletnih strani.
- Omogoča nadzor nad barvami, slogi pisave, razmaki, postavitvijo, ozadji (slikami na ozadjih) in raznimi drugimi efekti.

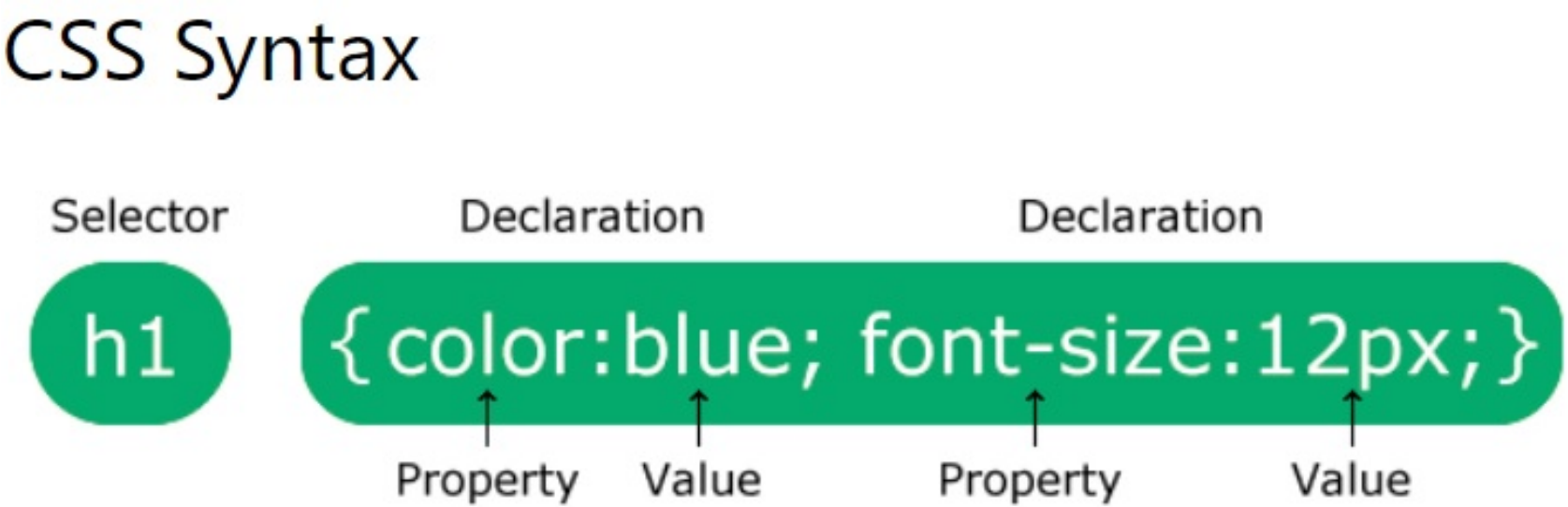
Primer: CSS-Example.html

Zakaj CSS?

- **Prihrani čas** - CSS dokument lahko napišemo enkrat in ga uporabimo na različnih spletnih straneh.
- **Hitrejši čas nalaganja strani** - ni potrebe, da za vsak HTML element določimo lastnosti, dovolj je napisati eno CSS pravilo za določen element, ki se aplicira na vseh teh elementih.
- **Enostavno vzdrževanje** - le spremenimo stil v CSS dokumentu in to se aplicira na vseh spletnih straneh, ki ta CSS uporabljajo.
- **Boljši od HTML** - omogoča veliko več oblikovanja in lastnosti kot HTML.
- **Kompatibilnost** - omogočajo optimiziran izgled za uporabo na različnih napravah.

CSS sintaksa

- Uporablja oblikovna pravila, ki jih brskalniki interpretirajo in aplicirajo na ustreznih mestih.
- Pravila so sestavljena iz treh delov: **Selector** (izbirni element/izbirnik), **Property** (lastnost) in **Value** (vrednosti)
- **Selector** - na katerih elementih bo stil uporabljen
- **Property** - npr.: barva, obroba, ozadje, idr.
- **Value** - ki jo določimo lastnosti.
- Za en Selector lahko določimo tudi več lastnosti zaporedoma, ločimo jih s podpičjem.



CSS komentarji

- Komentarje zapišemo med znake /* in */
- Enako oznako uporabljamo tako za enovrstične kot za večvrstične komentarje.
- Brskalnik pri pregledovanju stila komentarje preskoči.
- Primer: /* To je CSS komentar. */

CSS Selector (Izbirnik)

Poznamo različne vrste izbirnikov:

- Izbirnik elementa
- Univerzalni izbirnik
- Izbirnik razreda
- ID izbirnik
- Izbirnik podrejenega elementa
- Izbirnik potomca

Izbirnik elementa

Izbere elemente glede na ime HTML oznake. Na strani poišče vse HTML elemente, ki ustrezajo napisani oznaki.

V danem primeru bodo vsi elementi p pobarvani z rdečo barvo.
Primer: `p{ color: red;}`

Univerzalni izbirnik

Namesto izbire elementov določene vrste se univerzalni izbirnik ujema z imenom poljubnega elementa.

Primer: `*{ color: #000000;}`

Izbirnik razreda

Izbira elemente glede na lastnost HTML elementov **class**. Vsi elementi, ki vsebujejo izbran razred se bodo spremenjeni.

Naslednje pravilo določa barvo za vse elemente, ki vsebujejo razred **black**.
Primer: .black{ color: #000000;}

Lahko smo tudi bolj specifični in določimo pravilo le za določene elemente, ki v primeru vsebujejo razred **center**.
Primer: p.center{ text-align: center;}

Oba zgornja stila bi se aplicirala na naslednjem elementu
Ta paragraph bo s črno pisavo in na sredini.

ID izbirnik

Izbirnik ID-ja za izbiro določenega elementa uporablja atribut ID elementov v HTML. ID elementov mora biti edinstven znotraj posamezne strani. Posledično se ID izbirnik uporablja za izbiro edinstvenega elementa. Primer: #black{ color: #000000;}

Zgornji stil bi se apliciral na naslednjem elementu Ta paragraph bo s črno pisavo.

Izbirnik podrejenega elementa

Uporabimo ga, ko želimo pravilo uporabiti za določen element v kolikor leži znotraj drugega elementa.

V naslednjem primeru bo pravilo veljalo za element **em** le, če bo postavljen znotraj elementa **ul**.
Primer: `ul em{ color: #000000;}`

Izbirnik potomca

Uporabimo ga, ko želimo pravilo uporabiti za določen element v kolikor leži **neposredno** znotraj drugega elementa.

V naslednjem primeru bo pravilo veljalo za element **p** le, če bo postavljen neposredno znotraj elementa **body**. Ostali elementi **p** znotraj drugih elementov ne bodo spremenjeni.
Primer: body > p{ color: #000000;}

Gnezdenje izbirnikov (&)

Omogoča nam gnezdenje pravil znotraj drugega pravila. Pri tem se upošteva relacija starš-otrok med elementi.

V danem primeru se elementu z id-jem **sample** nastavi pisavo in velikost fonta, nato se znotraj elementa za element **a** določi barvo **crimson**, znotraj tega elementa pa se nastavi stanje **hover** in stanje **focus** na zeleno pisavo in rumeno ozadje.
Primer: CSS-NestingSelectors.html

Več hkratnih izbirnikov

En stil lahko uporabimo za več različnih izbirnikov. Le te ločimo z vejico.

Primer: h1, h2, h3 { color: #36C;}

Vključitev CSS stilov v HTML

- CSS lahko vključimo na štiri možne načine.
- **Embedded** - vključimo ga v glavo dokumenta (znotraj **head** oznake).
- **Inline** - uporabimo **style** atributa posameznega elementa.
- **External Link** - napišemo pravila v ločeno datoteko in vključimo v **link** oznaki.
- **External Import** - pravila iz ločene datoteke vključimo kot import.

Prednost CSS pravil

- Inline ima najvišjo prioriteto in povozi vsa ostala pravila.
- Embedded in external sta naslednja po stopnji prioritete, odvisno v kakšnem zaporedju so pravila podana.
- Default pravila brskalnika so na koncu.

Inline vključitev CSS pravil

- Uporabljamo lastnost **style** pri posameznem elementu za katerega določimo pravila.
- Ta pravila se uporabijo zgolj za dani element.
- Primer: <h1 style = “color: red;”>To je inline CSS</h1>

Embedded vključitev CSS pravil

- V glavo HTML dokumenta vključimo element <style>, ki ga moramo tudi zapreti.
- Znotraj elementa style napišemo CSS pravila.
- Primer: večina datotek, ki smo si jih že in si jih še bomo ogledali ima embedded CSS oblikovanje.

External Link vključitev CSS pravil

- CSS pravila zapišemo v ločeno datoteko s končnico .css.
- V datoteko napišemo le CSS pravila.
- CSS nato vključimo v HTML dokument s povezavo znotraj glave dokumenta v elementu <link>.
- Primer: <link rel="stylesheet" type = "text/css" href = "pot_do_datoteke" / >
- Obstaja tudi možnost z @import (poglejte sami).

Primer: CSS-External_Link.html in MyStyle.css

Lastnosti CSS

- barve (pisave, ozadja, obrob,...),
- ozadja (barve, slike, ponovitve,...),
- obrobe (stil, debelina, barva,...),
- prostor okoli elementov - izven okvirjev (margin),
- prostor okoli elementov - znotraj okvirjev (padding),
- dimenzije elementov,
- besedilo (barva, poravnanost, oblikovanje, transformacija,...),
- fonti pisav,
- stili seznamov
- obrobe tabel
- ...

Lastnosti CSS - podajanje barv

- Barve lahko podamo z imenom ali pa z RGB, HEX, HSL, RGBA ali HSLA.
- RBG je rdeča, modra, zelena (med 0 in 255). HEX je heksadecimalni zapis (šestnajstiški) za RGB barve (med 00 in ff). HSL (hue, saturation, lightness), kjer je hue stopinje (med 0 in 360), kjer se barva nahaja na barvnem kolesu, saturation je nasičenost barve (0% je odtenek sive, 100% je polna barva), ligthness (0% je črno, 50% ni ne svetlo ne temno, 100% je belo).
- RGBA in HSLA imata na četrtem mestu še alpha kanal, ki določa prosojnost med 0.0 (povsem prosojno) in 1.0 (povsem neprosojno).
- Primer: barva z imenom Tomato, rgb(255, 99, 71), #ff6347, hsl(9, 100%, 64%). S 50% prosojnostjo pa rgba(255, 99, 71, 0.5), hsla(9, 100%, 64%, 0.5).

Primer: CSS-ColorExample.html

Lastnosti CSS - ozadja

- background-color nastavi barvo ozadja. Elementi imajo na voljo tudi lastnost opacity za prosojnost elementa (lahko uporabimo namesto barvnih zapisov z alpha kanalom).
- background-image nastavi sliko na ozadje. Privzeto se slika ponovi, da pokrije celoten element.
- Primer: <body style="background-image: url('primer.jpg')</body>
- background-repeat nastavi ponovitve slike ozadja, če jo nastavimo pred tem. Možne vrednosti so: repeat-x, repeat-y in no-repeat.
- Za nastavitev več lastnosti ozadja hkrati lahko uporabimo kar okrajšavo background, pri tem moramo upoštevati vrstni red podanih vrednosti (nekatere lahko tudi izpustimo): color, image, repeat, attachment, position.

Primer: CSS-BackgroundExample.html, CSS-BackgroundExample2.html, CSS-BackgroundExample3.html

Lastnosti CSS - obrobe

- border-style nastavi stil obrobe. Možne vrednosti: dotted, dashed, solid, double, groove, ridge, inset, outset, none, hidden.
- Nastavimo lahko ločen stil obrobe za top, right, bottom, left, da podamo do 4 različne vrednosti ločene le s presledkom.
- border-width nastavi debelino obrobe. Vrednost podamo s številko in mersko enoto (brez presledka) v px, pt, cm, em,... ali pa z eno od treh vrednosti: thin, medium, thick.
- border-color nastavi barvo obrobe.
- Širino, stil in barvo (stil je obvezen) lahko nastavimo tudi z okrajšavo border, kjer je vrstni red: width, style, color.
- border-radius naredi robove bolj zaobljene, vrednost podamo s številko in mersko enoto px (brez presledka).

Primer: CSS-BorderExample.html, CSS-BorderExample2.html, CSS-BorderExample3.html

Lastnosti CSS - margin

- Zunanje robove nastavimo z: margin-top, margin-right, margin-bottom, margin-left.
- Uporabimo lahko okrajšavo margin, kjer podamo do štiri vrednosti (obnaša se enako kot border).
- Vrednosti lahko podamo v px, pt, cm,... procentualno glede na element (vrednosti so lahko tudi negativne).
- margin: auto; poravna element vodoravno (horizontalno) na sredino.

Primer: CSS-MarginExample.html

Lastnosti CSS - padding

- Notranje robove nastavimo z: padding-top, padding-right, padding-bottom, padding-left.
- Uporabimo lahko okrajšavo padding, kjer podamo do štiri vrednosti.
- Vrednosti lahko podamo v px, pt, cm,... procentualno glede na element (tukaj negativne vrednosti niso dovoljene).

Primer: CSS-PaddingExample.html

Lastnosti CSS - dimenzije

- Nastavimo lahko heigth in width posameznih elementov.
- Podamo jih lahko v px, pt, cm,... procentualno glede na element znotraj katerega postavljamo izbrani element.
- Nastavimo lahko tudi: min-width, min-heigth, max-width, max-heigth, ki določajo minimalne in maksimalne možne mere elementa

Primer: CSS-WidthHeightExample.html

Lastnosti CSS - besedilo

- color spremeni barvo besedila.
- text-align spremeni poravnanost. Možne vrednosti so: left, right, center, justify (poravna obojestransko).
- text-transform spremeni besedilo na enega od treh načinov: uppercase (vse v velike črke), lowercase (vse v male črke), capitalize (vse prve črke besed v velike črke).
- text-shadow nastavi senco besedilu. Nastaviti moramo horizontalen in vertikalen zamik sence, dodamo lahko tudi zamegljenost (blur), na koncu lahko dodamo tudi barvo.

Primer: CSS-TextExample.html

Lastnosti CSS - fonti

- font-family nastavi pisavo, pametno je uporabljati tudi backup pisave, če slučajno željena ni podprta. V tem primeru se dodatno pisavo (ali več) loči z vejico.
- Primer: želimo pisavo Georgia, ki spada med Serif pisave. Zato lahko navedemo font-family: Georgia, serif;, kar pomeni, da bo brskalnik poskušal prikazati Georgia pisavo, če je ni na voljo pa bo izbral sorodno pisavo iz Serif skupine.
- Več o družinah pisav si preberite sami.
- font-size nastavi velikost pisave v px ali v em (pixels/16 = em), običajno je privzeta velikost v brskalnikih 16px (1em).
- Dobra praksa je uporabljati enoto em, ki omogoča prilagajanje velikosti v vseh brskalnikih in nastaviti procentualno velikost za celotno telo HTML dokumenta (običajno na 100%), nato pa za ostale elemente ločeno.

Primer: CSS-FontExample.html

Lastnosti CSS - stili seznamov

- list-style-type nastavi način označevanja elementov seznama.
- Možnosti za neurejene sezname (ul) so: circle, disk, square, none
- Možnosti za urejene sezname (ol) so: upper-roman, lower-roman, upper-alpha, lower-alpha, decimal in še veliko drugih.
- Glej tudi prosojnice o HTML.

Primer: CSS-ListExample.html

Lastnosti CSS - obrobe tabel

- Obrobe se lahko nastavi celotni tabeli in posameznim elementom v tabeli.
- Primer CSS: table, th, td {border: 1px solid;} nastavi obrobe tabeli, header elementom tabele in data elementom tabele.
- border-collapse: collapse; poskrbi, da se prikažejo le enojne obrobe.
- border-bottom za elemente th in td nastavi vodoravne (horizontalne črte) med vrsticami tabele.

Primer: CSS-TableExample.html, CSS-TableExample2.html, CSS-TableExample3.html