

## 1 CSS – Cascading Style Sheets (kratko podsećanje)

- stilovi definišu način prikaza HTML elemenata
- uvedeni su sa verzijom HTML 4.0

Upotrebom stilova, razdvaja se kod koji opisuje sadržaj HTML dokumenta od koda koji opisuje formatiranje elemenata. Kod za formatiranje nalazi se, najčešće, u posebnoj **.css** datoteci, čime je omogućeno lakše menjanje formata prikaza.

### CSS pravila

Sastoje se iz dva glavna dela:

- selector (najčešće HTML element koji se stilizuje)
- jedna ili više deklaracija (deklaracija se sastoji od svojstva i vrednosti razdvojenih dvotačkom; svaka deklaracija završava se ';' i sve su ograđene vitičastim zagradama)

### Primeri

```
p {color:red;text-align:center;}
```

ili svaka deklaracija može biti zadata u posebnom redu, radi preglednosti:

```
p
{
color:red;
text-align:center;
}
```

### CSS komentari

Format je: `/* ... */`

Komentari se ignorišu od strane prikazivača.

```
/* Komentar */
p
{
text-align:center;
/* jos jedan komentar */
color:black;
font-family:arial;
}
```

### Id i class selektori

Pored toga što se može postaviti stil za HTML element, koji je u tom slučaju selektor odgovarajućeg pravila, CSS daje mogućnost korisniku da definiše nove selektore koji se označavaju kao "id" i "class" selektori.

### Id selektor

Koristi se da postavi stil za jedan, jedinstveni element. Id selektor koristi atribut *id* HTML elementa i zadaje se sa prefiksom „#“.

Naredno pravilo primeniće se nad HTML elementom sa id atributom `paral`:

```
#paral
{
text-align:center;
color:red;
}
```

### ***Class selektor***

Koristi se za definisanje stila za više elemenata. Svi elementi koji pripadaju istoj klasi, imaju isti stil. Class selektor koristi atribut *class* HTML elementa i zadaje se sa prefiksom „.“.

U narednom primeru, svi HTML elementi sa atributom *class*="center" biće centralno poravnati:

```
.center {text-align:center;}
```

Može se zadati i da samo neki HTML elementi date klase imaju odgovarajući stil:

```
p.center {text-align:center;}
```

### ***Načini zadavanja CSS stilova***

- eksterni style sheet
- interni style sheet
- inline style

### ***Eksterni style sheet***

Idealan kada se stil primenjuje na veći broj strana, na primer na sve strane koje čine jedan veb sajt. Promena izgleda kompletnog veb sajta vrši se samo menjanjem .css datoteke koja sadrži eksterni style sheet.

Svaka strana u tom slučaju mora da sadrži vezu ka .css datoteci, što se zadaje tagom `<link>` koji se nalazi u sekciji head HTML strane:

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

Datoteka sa stilovima ne sme da sadrži html tagove. Primer:

```
hr {color:sienna;}
p {margin-left:20px;} /* izmedju vrednosti i jedinice ne sme da bude razmak */
body {background-image:url("images/back40.gif");}
```

### ***Interni style sheet***

Koristi se kada jedan html dokument ima jedinstveni stil. Zadaje se tagom <style> u sekciji head html strane:

```
<head>
<style>
hr {color:sienna;}
p {margin-left:20px;}
body {background-image:url("images/back40.gif");}
</style>
</head>
```

### Inline style

Ne preporučuje se česta upotreba, jer se na ovaj način kod za opis sadržaja html strane meša sa kodom za formatiranje.

Zadaje se upotrebom style atributa za odgovarajući html tag:

```
<p style="color:sienna;margin-left:20px">This is a paragraph.</p>
```

### Višestruki stilovi se slivaju u jedan (cascading style sheets)

Primenjuje se kada za isti html element postoji više zadatih stilova. U tom slučaju svi stilovi se slivaju (cascade) u jedan (virtualni) stil na osnovu sledećih pravila (navedena su opadajuće po prioritetu):

- podrazumevana podešavanja prikazivača
- eksterni style sheet
- interni style sheet
- inline style

Ako je veza ka .css datoteci zadata nakon internog stila u okviru sekcije head, eksterni stil će preklopiti interni.

Detaljniji pregled:

<http://www.w3schools.com/css/>

## 2 JavaFX CSS - Uvod

JavaFX CSS (Cascading Style Sheets) bazira se na verziji 2.1 W3C CSS-a, sa nekim dodacima koji trenutno postoje u verziji 3 i proširenjima specifičnim za JavaFX.

JavaFX CSS je dizajniran tako da je omogućeno parsiranje stilova od strane bilo kog CSS parsera, čak i onih koji ne podržavaju JavaFX ekstenzije. Stoga je moguće u istoj .css datoteci definisati, na primer, JavaFX stilove i HTML stilove.

Svojstva koja se opisuju JavaFX stilovima u nazivu imaju prefiks „-fx-“. To važi i za ona svojstva koja su kompatibilna sa standardnim HTML CSS svojstvima, jer JavaFX koristi nešto drugačiju semantiku za njihove vrednosti.

JavaFX ne podržava CSS layout svojstva kao što su: *float*, *position*, *overflow* i *width*. CSS svojstva: *padding* i *margins* podržana su za neke JavaFX objekte.

JavaFX ne podržava zadavanje liste familije fontova razdvojenih zapetom u okviru svojstva `-fx-font-family`.

Koristi HSB model boja umesto HSL modela.

### 3 CSS stilovi i graf FX scene

CSS stilovi se primenjuju na čvorove grafa scene na sličan način kao i CSS stilovi na HTML elemene. Najpre se stil primenjuje na roditelja, a potom i na decu.

CSS stilovi se primenjuju na čvor asinhrono, tj. nakon što je čvor kreiran i dodat u graf scene, ali pre prikazivanja grafa scene.

#### Promenljiva `styleClass`

- svaki čvor grafa scene ima promenljivu `styleClass`, tipa `List<String>`
- analogno atributu `class="..."` HTML elementa

#### Promenljiva `id`

- svaki čvor grafa scene ima promenljivu `id`, tipa `String`
- analogno atributu `id="..."` HTML elementa

CSS stilovi mogu da se definišu:

- u posebnoj datoteci (*style sheet*)
- direktno u kodu (*inline style*)

URL za *style sheet* datoteku zadaje se kao vrednost **stylesheet** promenljive za scenu. URL može biti zadat kao apsolutna ili relativna putanja.

Inline stilovi zadaju se direktno u kodu za odgovarajući čvor metodom `setStyle()`.

Analogni su atributu `style="..."` HTML elementa.

Postavljeni Inline stilovi imaju prednost nad stilovima definisanim u *style sheet* datoteci.

Od JavaFX 2.1, klasa `Parent` ima svojstvo **stylesheets**, čime je omogućena primena stilova nad kontejnerom. Dakle, jedan ogranak grafa scene može imati drugačije stilove od ostatka grafa. Bilo koja instanca ove klase može imati postavljene stilove. Stilovi za dete čvor se postavljaju redom, na osnovu inline stilova (najviši prioritet), stilova svih njegovih predaka i stilova same scene.

#### Nasleđivanje svojstava

Da bi svojstvo moglo da se nasledi, treba mu zadati vrednost **"inherit"**.

Svojstva navedena u tabeli se podrazumevano nasleđuju.

Class	Property	CSS Property	Initial Value
<code>javafx.scene.Node</code>	<code>cursor</code>	<code>-fx-cursor</code>	<code>javafx.scene.Cursor.DEFAULT</code>
<code>javafx.scene.text.Text</code>	<code>textAlignment</code>	<code>-fx-text-alignment</code>	<code>javafx.scene.text.TextAlignment.LEFT</code>
<code>javafx.scene.text.Font</code>	<code>font</code>	<code>-fx-font</code>	<code>Font.DEFAULT (12px system)</code>

## Primer

```
Scene scene = new Scene(new Group());
scene.getStylesheets().add("test.css");
Rectangle rect = new Rectangle(100,100);
rect.setLayoutX(50);
rect.setLayoutY(50);
rect.getStyleClass().add("my-rect");
((Group)scene.getRoot()).getChildren().add(rect);
```

Bez stilova, pravougaonik će biti prikazan uobičajeno, crnom bojom.

Ako style sheet datoteka test.css sadrži:

```
.my-rect { -fx-fill: red};
```

pravougaonik će biti crven umesto crn.

## Upozorenja JavaFX CSS parsera

JavaFX CSS parser prijavljuje upozorenje u slučaju sintaksne greške.

### Primer:

```
WARNING: com.sun.javafx.css.parser.CSSParser declaration Expected '<percent>'
while parsing '-fx-background-color' at ?[1,49]
```

Konstrukcija '?[1,49]' odnosi se na lokaciju greške. Format lokacije greške je:

```
<url>[line, position]
```

gde <url> može biti:

- URL datoteke, u slučaju da se greška javila prilikom parsiranja datoteke
- ?, ako je u pitanju inline stil

## ***Detaljni pregled za JavaFX CSS:***

<http://docs.oracle.com/javafx/2/api/javafx/scene/doc-files/cssref.html>