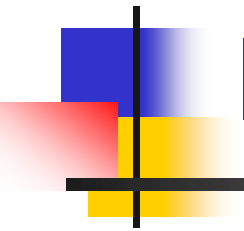
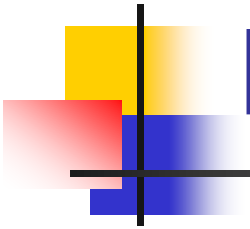


Interneto technologijos

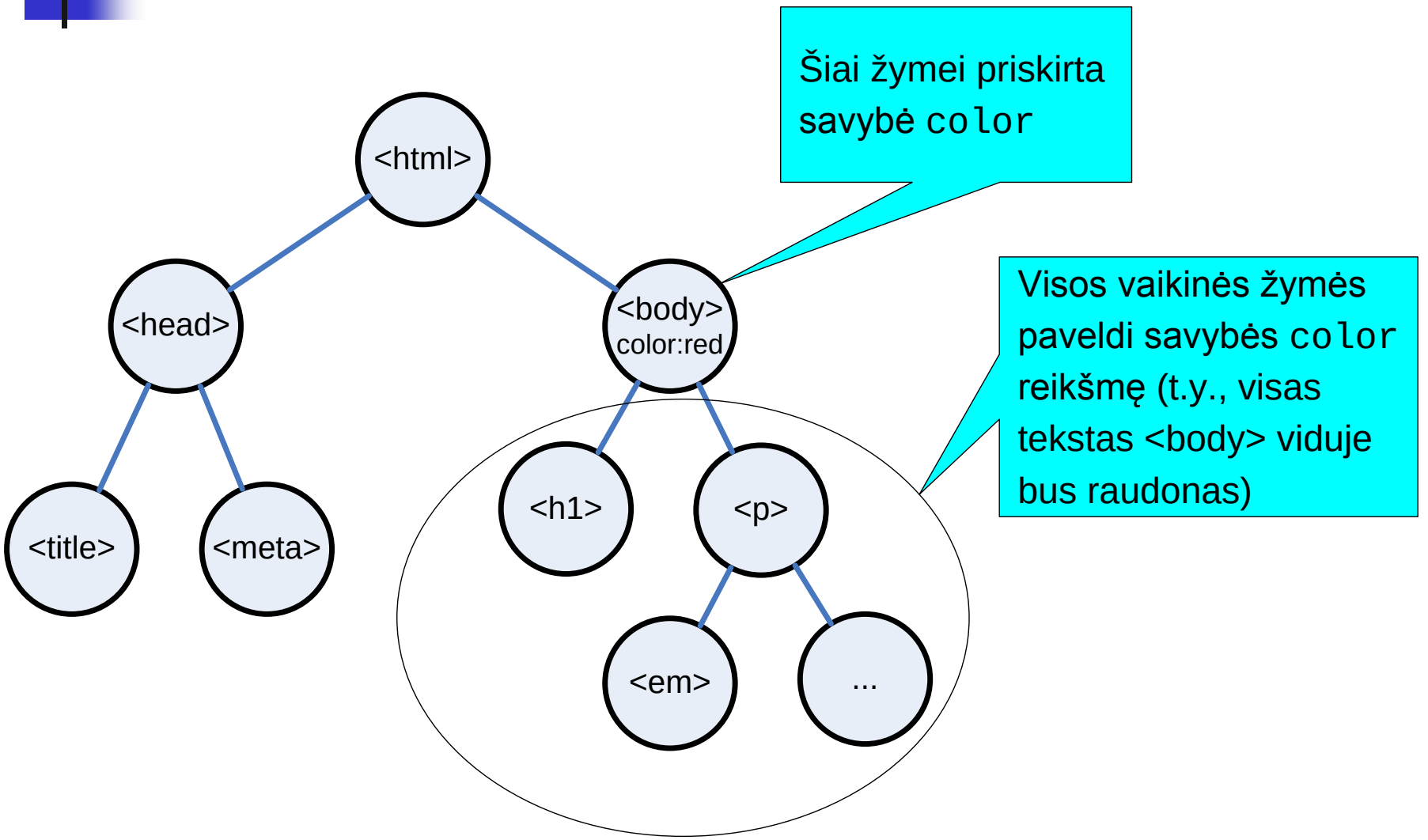
CSS
Paveldėjimas,
kaskadų mechanizmas,
konfliktų sprendimas



Paveldėjimas

- Kai kurios CSS savybės yra paveldimos
  - Pvz.: `color`, `font-family`, `font-size`, `text-align`, `visibility`
- Kitos – nėra paveldimos
  - Pvz.: `background-color`, `background-image`, `margin`, `padding`
- Jei savybė paveldima, tai visi žymės, turinčios savybę, įpėdiniai pagal nutylėjimą taip pat įgauną tą savybės reikšmę
  - Jei ta pati savybė yra nurodyta ir įpėdiniui, jis įgyja nurodytą (o ne paveldėtą) tos savybės reikšmę
    - Panašiai kaip OO kalbose, paveldėtos savybės reikšmę galima *perrašyti*

Paveldėjimo pavyzdys



Paveldėjimo pavyzdys

- Perrašant paveldimą savybę, galima ne tik nurodyti visiškai naują reikšmę, bet ir pakeisti (nurodant procentus) paveldėtą

```
body { font-size: 10pt }
h1 { font-size: 130% }
```

h1 teksto šrifto dydis bus 130% nuo paveldėtos reikšmės

- Procentai ne visada reiškia, kad bus imama paveldėta reikšmė – reikia žiūrėti savybės dokumentaciją

Kokios savybės paveldimos

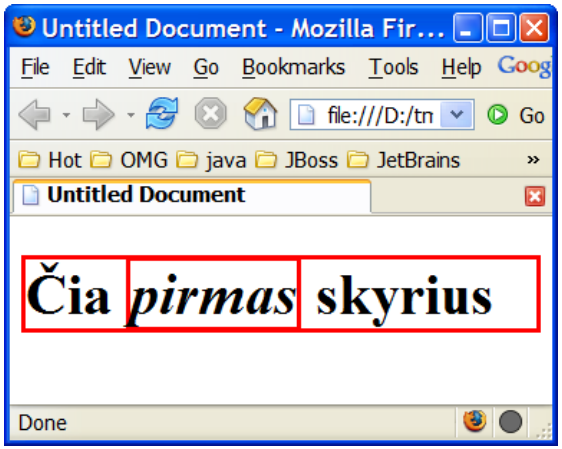
- CSS 2.1 specifikacijos priede F kiekvienai savybei yra pasakyta, ar ji paveldima, ar ne
  - http://www.w3.org/TR/CSS21/propidx.html

Name	Values	Initial value	Applies to (Default: all)	Inherited?	Percentages (Default: N/A)	Media groups
'azimuth'	<angle> | [[left-side | far-left | left | center-left | center | center-right | right | far-right | right-side] || behind] | leftwards | rightwards | inherit	center		yes		aural
'background-attachment'	scroll | fixed | inherit	scroll		no		visual
'background-color'	<color> | transparent | inherit	transparent		no		visual

Raktinis žodis `inherit`

- Jei savybė pagal dokumentaciją nėra paveldima, vaikas (bet ne įpėdinis!) vis tiek gali paveldėti jos reikšmę, jei panaudos raktinį žodį `inherit`:

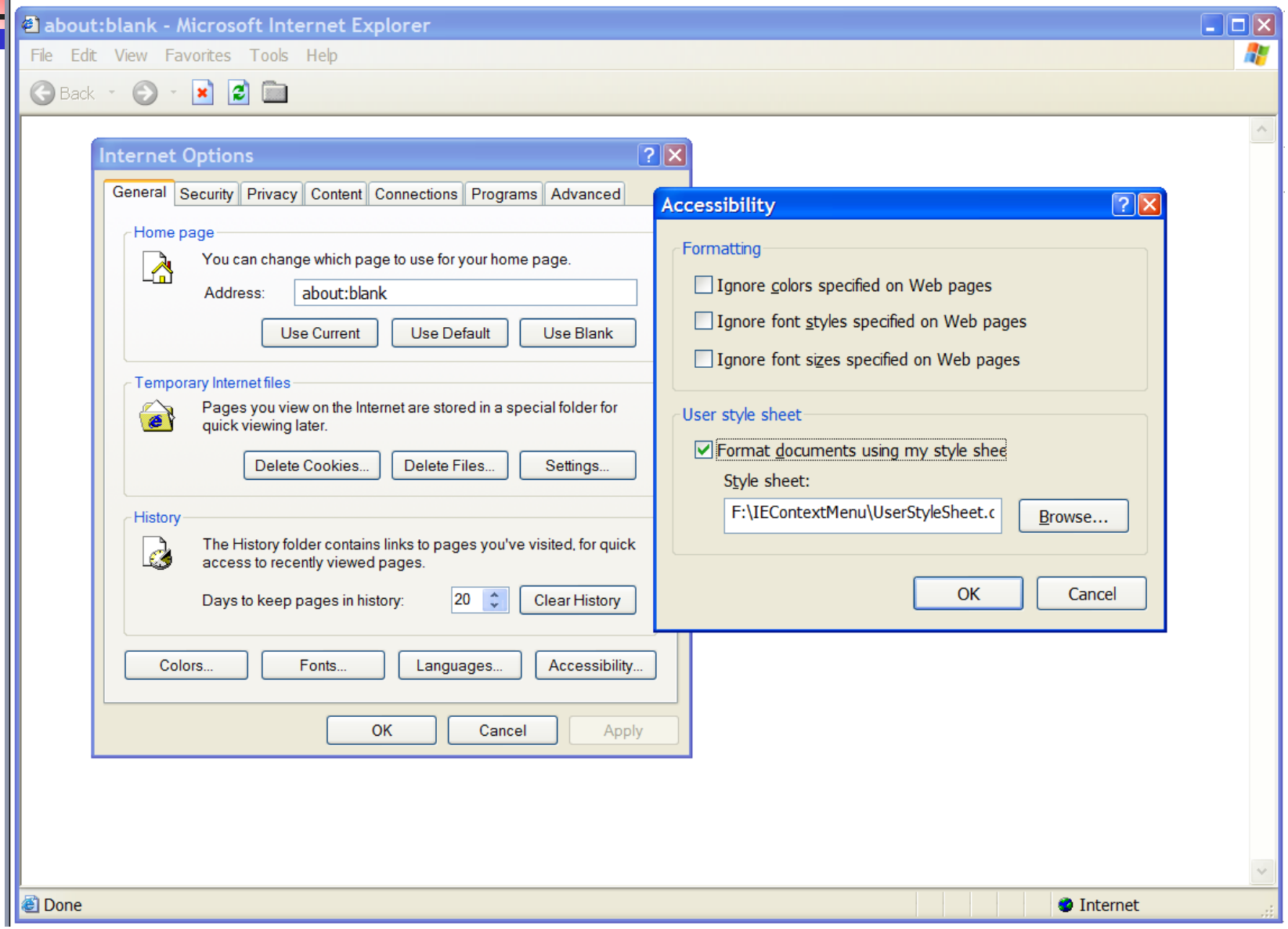
```
h1 {border: solid red}
em {border: inherit}
```



Autoriaus, vartotojo ir naršyklės CSS dokumentai

- CSS dokumentai į naršyklę paprastai pakliūna iš WWW serverio (t.y., juos sukuria svetainės autorius)
  - Tokie CSS dokumentai vadinami *autoriaus* dokumentais
- Vartotojas irgi turi teisę susikurti savo CSS dokumentus ir (jei naršyklė turi tokias galimybes) liepti naršyklei visus puslapius vizualizuoti su jo stiliais
  - Tokie CSS dokumentai vadinami *vartotojo* CSS dokumentais
- Naršyklė taip pat turi savo stilių lentelę, su kuria ji suteikia stilių tiems (X)HTML dokumentams, kuriems nėra asocijuotų nei autoriaus, nei vartotojo CSS dokumentų
  - Tai taip vadinamas *naršyklės* (angl. user-agent) CSS dokumentas
- Atskiru atveju (X)HTML dokumentui gali būti nurodytas tiek autoriaus, tiek vartotojo CSS dokumentas (naršyklės CSS dokumentas yra visada neišreikštinai nurodytas)
  - Galime gauti *stilių konfliktą*

Vartotojo CSS dokumentas – Internet Explorer



Stilių konfliktas

- Paprasčiausias stilių konflikto pavyzdys:

```
p {color: blue}
body p {color: red}
```

- Abi taisyklės nustato žymės <p> spalvą. Klausimas, kokia gi ta spalva bus?
  - Turime stilių konfliktą
- Konfliktas gali kilti ir dėl to, kad yra keli autoriaus stilių dokumentai (konfliktuojantys tarpusavyje), ir dėl to, kad vartotojas papildomai nusirodė savo stiliaus dokumentą
  - Jei yra bent vienas stiliaus dokumentas (autoriaus arba vartotojo), tai jis greičiausiai konfliktuos su naršyklės stiliumi, kuris apibrėžia išvaizdą pagal nutylėjimą

Konfliktų sprendimas – kaskadų mechanizmas

- Jei vienai žymei kelios CSS taisyklės nurodo prieštaraujančias reikšmes, turime stilių konfliktą. Jis sprendžiamas, taisyklėms priskiriant prioritetus:

1. Pirmiausia vertinamas stilių šaltinis ir taisyklių svarba:
   1. naršyklės stiliai
   2. vartotojo stiliai
   3. autoriaus stiliai
   4. autoriaus *svarbūs* stiliai
   5. vartotojo *svarbūs* stiliai

Mažiausias prioritetas

Didžiausias prioritetas

2. Jei konflikto išspręsti nepavyko, tai:
   1. Nugali ta CSS taisyklė, kurios selektoriaus *specifiškumas* yra didesnis
   2. Jei specifiškumas vienodas, nugali vėliau paskelbta taisyklė

Svarbūs stiliai

- Savybė gali būti paskelbta svarbia, naudojant raktinį žodį `!important`

```
p {font-style: italic !important}
```

- Šį raktinį žodį savo stilių dokumentuose naudoti gali tiek autorius, tiek vartotojas

Taisyklių selektorių specifiškumo skaičiavimas

- Taisyklės specifiškumui skaičiuoti naudojamas vektorius (a, b, c, d). Reikšmės įvertinamos taip:
  - Jei CSS taisyklė paskelbta HTML žymės atribute `style` (o ne stilių lentelėje), tai a=1, priešingu atveju a=0
  - b = {ID selektorių skaičius}
  - c = {atributų selektorių ir pseudo-klasių skaičius}
  - d = {žymių selektorių ir pseudo-elementų skaičius}
- Taisyklės specifiškumas gaunamas atlikus a, b, c, d reikšmių konkatenaciją: abcd. Pvz.: (0, 1, 4, 1) pavirsta į skaičių 0141.

Specifiškumo skaičiavimo pavyzdžiai

```
*                       a=0 b=0 c=0 d=0
li                      a=0 b=0 c=0 d=1
li:first-line           a=0 b=0 c=0 d=2
ul li                   a=0 b=0 c=0 d=2
body ol > li            a=0 b=0 c=0 d=3
ol > [att=up]           a=0 b=0 c=1 d=1
div ol li.red           a=0 b=0 c=1 d=3
li#x34                  a=0 b=1 c=0 d=1
style="color:red"       a=1 b=0 c=0 d=0
```

Savybės `display` ir `visibility`

- `display: inline | block | none | ...`
- `visibility: visible | hidden | ...`
- Abi šios savybės kontroliuoja, ar žymės turinys bus matomas naršyklės lange, ar nebus matomas
- Skirtumai:
  - jei stilių lentelėje žymei nurodysite savybę `visibility:hidden`, žymės turinys nebus rodomas, bet jam bus palikta (rezervuota) tuščia vieta
  - `display:none` atveju žymės turinys nerodomas, ir tuščia vieta nepaliekama (rezultatas toks, tarsi tos žymės HTML dokumente iš viso nėra)

`inline` ir `block` skirtumai

- `block` reiškia, kad žymės turinys bus vaizduojamas naršyklės lange atskiru bloku (stačiakampio formos sritimi)
  - Taip elgiasi žymės `h1-h6`, `p`, `div` ir daugelis kitų
- `inline` reiškia, kad žymės turinys naujo bloko nekurs, o bus atvaizduotas einamojo bloko eilutėje
  - Žymės `i`, `b`, `em`, `strong`, `span`, ir t.t.

inline pavyzdys (neprasmingas)

```
p {display: inline}
```

- HTML dokumentas:

```
<p>Pirmas paragrafas.</p>
<p>Antras paragrafas.</p>
```

- bus pavaizduotas taip: