CSS & Eclipse 4 Application Platform
Brian de Alwis
Manumitting Technologies
bsd@mt.ca
<table>
<thead>
<tr>
<th>Task</th>
<th>Notes</th>
<th>Journal</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Task:</strong></td>
<td>Import my tasks from a Toodledo account</td>
<td></td>
</tr>
<tr>
<td><strong>Starred:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Project:</strong></td>
<td>Kizby Desktop: Welcome</td>
<td></td>
</tr>
<tr>
<td><strong>Status:</strong></td>
<td>Tentative</td>
<td></td>
</tr>
<tr>
<td><strong>Location:</strong></td>
<td>Computer</td>
<td></td>
</tr>
<tr>
<td><strong>Details:</strong></td>
<td>Start populating your Kizby Desktop repository by importing tasks from a Toodledo account. Select File -&gt; Import... and then follow the prompts.</td>
<td></td>
</tr>
<tr>
<td><strong>Start:</strong></td>
<td>Wed Jan 12, 2011 08:42 PM (tomorrow)</td>
<td></td>
</tr>
</tbody>
</table>
Eclipse 4 goal: make writing plugins easy
Eclipse 4 goal:
make writing plugins easy

- Allow better control over the look of Eclipse-based products
Eclipse 4 goal: make writing plugins easy

- Allow better control over the look of Eclipse-based products
- Let those who can, do
Meet the Pseudo Class Selectors

CSS-Tricks

A Web Design Community
Making
Websites
Better
Since 2007

- Articles The blog
- Forums Get & give help
- Snippets Clip and save
- Videos Watch and learn
- Almanac Reference guide
- Gallery Ooooh Aaah
- Downloads Just a seed
- Deals Save bucks

What do you want to learn today?
Treehouse is the best place to learn web design, web development, and iOS.

Meet the Pseudo Class Selectors
by: Chris Coyier
Mar 17 2010
66

Pseudo class selectors CSS selectors with a colon preceding them. You are probably very familiar with a few of them. Like hover:

a:hover { /* some css */

CSS-Tricks

A Web Design Community
Making Websites Better

Since 2007

- Articles The blog
- Forums Get & give help
- Snippets Clip and save
- Videos Watch and learn
- Almanac Reference guide
- Gallery Ooohh Aaahh
- Downloads Just a seed
- Deals Save bucks

What do you want to learn today?
Treehouse is the best place to learn web design, web development, and iOS.

Meet the Pseudo Class Selectors

by: Chris Coyier
Mar 17 2010

Pseudo class selectors CSS selectors with a colon preceding them. You are probably very familiar with a few of them. Like hover:

```
:hover { /* hover */
}
```

They are immensely useful in a variety of situations. Some of them are :hover, some CSS2... it depends on each particular one. Outside of IE, they have great browser support. In IE land, even IE8, support is pretty baren. However, the IE9 preview has full support of them. The link-related ones work but not much else. Let's take a brief look at each one of them. But they are there just to make...
HTML
Jon Dearden’s winning entry for the EclipseCon e4 Rover Competition (2010)
CSS Rules

h1 { color: red; }

selector | property | value
CSS Rules  Selectors
CSS Rules  Selectors

h1
h1.class
CSS Rules

h1

h1.class

h1#id
CSS Rules

Selectors

**h1**

**h1.class**

**h1#id**

```javascript
getClass() == h1

instanceof

==
```
CSS Selectors for SWT

Shell.MTrimmedWindow
  |-- Composite.MArea
     |-- CTabFolder.EditorStack
        |-- CTabItem.Editor
              |-- CTabItem.Editor
                          |-- Composite#f1
                                      |-- Group
                                      |-- Text
                                          |-- Table#serverList
                                          |-- Text
                                          |-- Table
                                          |-- Combo
CSS Selectors for SWT

Composite

Composite.MArea

Shell.MTrimmedWindow

Composite.MArea

CTabFolder.EditorStack

CTabItem.Editor

CTabItem.Editor

Composite#f1

Composite#f2

Group

Text

Text

Table

Table#serverList

Table Combo
Compound Selectors

Shell.MTrimmedWindow
  Text
  CTabFolder.EditorStack
    CTabItem
      Group
      Text
    CTabItem
      Label
      Tree
      Combo
      Text
Child Selectors

.MTrimmedWindow > Text

Shell.MTrimmedWindow

Text

CTabFolder.EditorStack

CTabItem

Group Text

Text

CTabItem

Label Tree Combo
Descendant Selectors

```
Shell.MTrimmedWindow
  Text
  CTabFolder.EditorStack
    CTabItem
      Group
        Text
          Text
    CTabItem
      Label
      Tree
      Combo
```
Descendant Selectors

```
Shell
| Text
```

```
Shell.MTrimmedWindow

CTabFolder.EditorStack

CTabItem

Group

Text

Text

CTabItem

Label

Tree

Combo
```
Attribute Selectors

Button[style~='SWT.CHECK']

Widget.class ~ Widget[class~='class']

= Exact value

~ Exact value
Pseudo-elements and classes

Text:focus { background: parchment; }

Shell:parented { background: red; }

What are these properties?
Demo: CSS Spy & CSS Scratchpad
Exposing Information

Leveraging the CSS for your own widgets
Extend in 3 easy steps

- Identify your widgets
- Define matching DOM elements where necessary
- Add custom properties
Identifying Widgets

- E4 exposes SWT widgets too: `type#id.class`
  - `type` corresponds to the widget class
  - `#id` matches an object's elementId `id` (with `.' translated to `-'.`)
  - `.class` matches against the modelled element type, any tags on the modelled element, style bits, and widget data key-value items
Aside: Assigning Class & Id

```java
import org.eclipse.e4.ui.css.swt.dom.WidgetElement;

WidgetElement.setCSSClass(widget, "class string");

WidgetElement.setID(widget, "id");

WidgetElement.getEngine(widget).applyStyles(widget, true);
```

```css
.error { border: 1px red; }

#cssSource { font: Courier; }
```
Widgets ⇔ DOM Elements

- Defines:
  - Local + package name

- Provides:
  - Attributes
  - Pseudo elements
<extension point="org.eclipse.e4.ui.css.core.elementProvider">
  <provider>
    class="org.eclipse.e4.ui.css.swt.dom.SWTElementProvider">
  <widget class="org.eclipse.swt.widgets.Control" />
  <widget class="org.eclipse.swt.widgets.Composite" />
  <widget class="org.eclipse.swt.widgets.Button" />
  </provider>
</extension>
Add custom properties handlers

```xml
<extension
    point="org.eclipse.e4.ui.css.property.handler">
  <handler
    adapter="org.eclipse.e4.ui.css.swt.dom.CTabFolderElement"
    composite="false"
    handler="o.e.e4.ui.css.swt.properties.custom.CSSPropertyCornerRadiusSWTHandler">
    <property-name name="swt-corner-radius"/>
    <property-name name="corner-radius"
      deprecated="renamed as swt-corner-radius"/>  
  </handler>
</extension>
```
public class XXXPropertyHandler implements ICSSPropertyHandler {
    public boolean applyCSSProperty(Object element, String property, CSSValue value, String pseudo, CSSEngine engine) throws Exception {
        ...
    }

    public String retrieveCSSProperty(Object widget, String pseudo, CSSEngine engine) throws Exception {
        return ...
    }
}
Pitfalls
Composite.getBackgroundMode()
Composite.getBackgroundMode()
Using parent’s background
Which rule wins?
Which rule wins?

Specificity   Cascades
CSS Differences: HTML vs SWT

- SWT.BORDER vs “border” property
  - “border” is emulated
- Gradients
- Background mode vs gradients & images
CSS & JFace

- How to use CSS for setting fonts and colours for label providers?
Building Your Own Engine
Best Practices
Best practices

- Identify rules for labeling/identifying structure
- Scoping
- SWT Quirks
  - background mode
  - background image & background colour
- (Lack of) support for non-px dimensions
Scoping

Text {
  color: 'white';
  background-color: 'black';
}

Tuesday, 27 March, 12
Scoping

Text {
  color: ‘white’;
  background-color: ‘black’;
}

Affects Text instances everywhere. Including startup dialogs.
Scoping

.MTrimmedWindow Text,
.MWindow Text {
  color: 'white';
  background-color: 'black';
}

Tuesday, 27 March, 12
Use classes, not types
Use CSS Tools

- SCSS/SASS (sass-lang.net)
  - CSS precompiler
- CSS Lint (csslint.net)
- Web Designers (e.g., OOCSS)
  - Learn from lessons of what to do and not to do
Be careful out there, kids

- Many articles generated by non-developers promulgating unsound techniques (e.g., “classitis”, avoiding semantically-meaningful names!!!)
- But these are the people who will be doing the styling
Future work (M7 and beyond)

- CSS Box model + floats — stop embedding layouts?
- Text transforms
- Inheritance on type field?
Resources

- e4 forum and mailing list
Thank you!

- Angelo Zerr — original implementation
- Bogdan Georghe
Pre-emptive Questions
How does this work with the SWT Browser widget?