

Programmazione WEB Avanzata per la Fisica Sperimentale

WEB



Strumenti per il trattamento
dei dati client-side al tempo di HTML5

Overview

SVG is a platform for two-dimensional graphics. It has two parts: an XML-based file format and a programming API for graphical applications. Key features include shapes, text and embedded raster graphics, with many different painting styles. It supports scripting through languages such as ECMAScript and has comprehensive support for animation.

SVG is used in many business areas including Web graphics, animation, user interfaces, graphics interchange, print and hardcopy output, mobile applications and high-quality design.

SVG is a royalty-free vendor-neutral open standard developed under the W3C Process. It has strong industry support; Authors of the SVG specification include Adobe, Agfa, Apple, Canon, Corel, Ericsson, HP, IBM, Kodak, Macromedia, Microsoft, Nokia, Sharp and Sun Microsystems. SVG viewers are deployed to over 100 million desktops, and there is a broad range of support in many authoring tools.

SVG builds upon many other successful standards such as XML (SVG graphics are text-based and thus easy to create), JPEG and PNG for image formats, DOM for scripting and interactivity, SMIL for animation and CSS for styling.

Full Pixel Tracker Page - Mozilla Firefox

http://pcmagni.mib.infn.it:1956/urn:daq-application:id=51/XGI_RequestTrackerComponent?trackerComponent=BPix_Bml_LYR3#

Mozilla Firefox

http://pcmagni.mib.infn.it:1956/urn:daq-application:id=51/XGI_RequestNewFTopoMap#

Barrel Tracker State Forward Tracker State

BPix_Bml - Layer3

Looking towards LHC Center +Z axis

ROC: Value:

Configuration Integrity Checks for 45483|AddressLevelCalibrationWithPixels_P5_Full

Missing Components	Comment
BPix_Bml_SEC2_LYR3_LDR4F_MOD3_ROC0	Module Completely Absent
BPix_Bml_SEC2_LYR3_LDR4F_MOD3_ROC1	Module Completely Absent
BPix_Bml_SEC2_LYR3_LDR4F_MOD3_ROC10	Module Completely Absent
BPix_Bml_SEC2_LYR3_LDR4F_MOD3_ROC11	Module Completely Absent
BPix_Bml_SEC2_LYR3_LDR4F_MOD3_ROC12	Module Completely Absent
BPix_Bml_SEC2_LYR3_LDR4F_MOD3_ROC13	Module Completely Absent
BPix_Bml_SEC2_LYR3_LDR4F_MOD3_ROC14	Module Completely Absent
BPix_Bml_SEC2_LYR3_LDR4F_MOD3_ROC15	Module Completely Absent
BPix_Bml_SEC2_LYR3_LDR4F_MOD3_ROC2	Module Completely Absent
BPix_Bml_SEC2_LYR3_LDR4F_MOD3_ROC3	Module Completely Absent

Page 1 of 4 Items 1 - 10 of 40 warning messages

Check all KOCs Optional: ? Complete: ? Mandatory: 1 Unchecked: 16

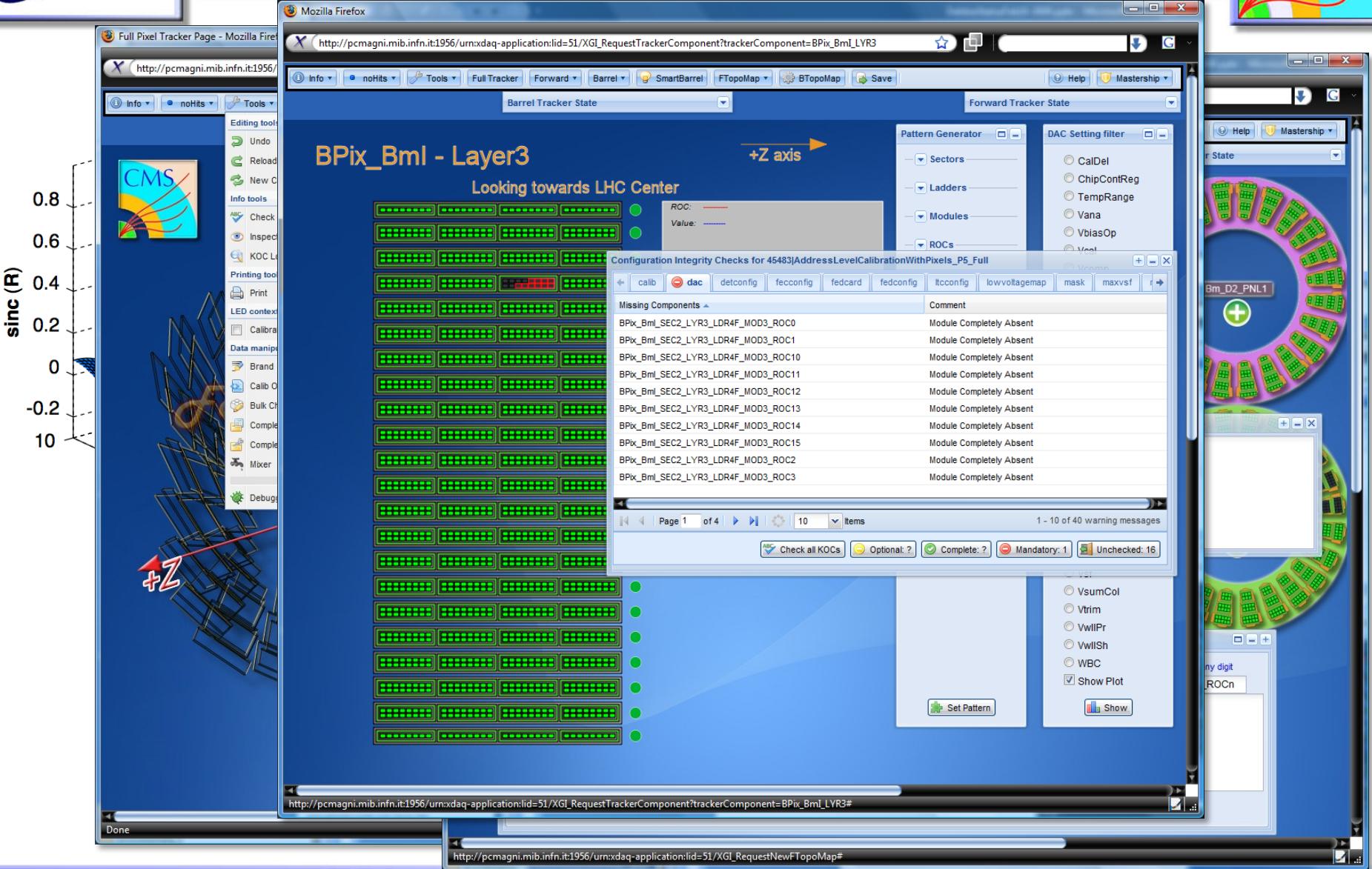
VsumCol Vtrim VwillPr VwillSh WBC Show Plot

Set Pattern Show

Barrel State Forward State

Bm_D2_PNL1

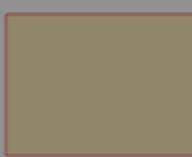
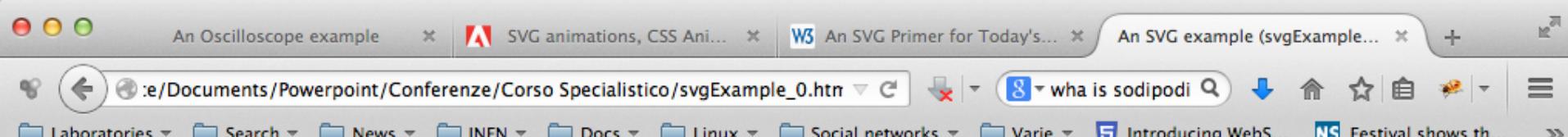
Any digit ROCn



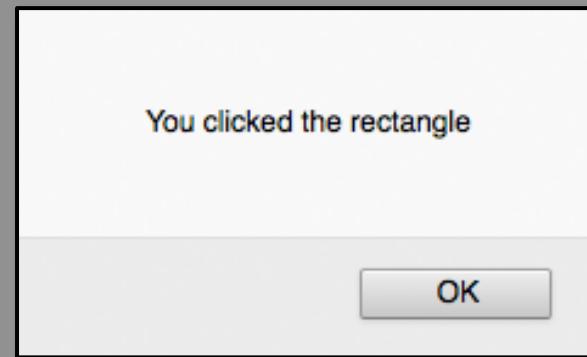


```
1 <html>
2   <head>
3     <meta charset="UTF-8">
4     <title>An SVG example (svgExample_0.html)</title>
5   </head>
6
7   <body>
8
9     <svg width="300px"
10       height="300px"
11       xmlns="http://www.w3.org/2000/svg">
12       <text x="10"
13             y="120"
14             font-size="20">This is svgExample_0.html</text>
15       <rect x="10"
16             y="10"
17             width="100"
18             height="75"
19             style="stroke: #000000; fill: #eeeeee;"
20             onmouseover="this.style.stroke = '#ff0000'; this.style['fill'] = '#fce37d';"
21             onmouseout = "this.style.stroke = '#000000'; this.style['fill'] = '#eeeeee';"
22             onclick="alert('You clicked the rectangle'));">
23     </rect>
24   </svg>
25
26 </body>
27
28 </html>
29
```





This is `svgExample_0.html`



File Edit Search Preferences Shell Macro Windows Help

svgExample_0.html svgExample_1.html svgExample_2.html

/Users/menasce/Documents/Powerpoint/Conferenze/CORSO Specialistico/svgExample_1.html byte 712 of 1319 L: 25 C: 41

```
19     object.width.baseVal.value = '100' ;
20     object.x.baseVal.value    = '10' ;
21 }
22 else if( direction == 'on' )
23 {
24     object.style.stroke        = '#000000' ;
25     object.style.fill         = '#654321' ;
26     object.width.baseVal.value = '400' ;
27     object.x.baseVal.value   = '100' ;
28 }
29 }
30 </script>
31 </head>
32
33 <body>
34
35 <svg width="300px"
36     height="300px"
37     xmlns="http://www.w3.org/2000/svg">
38     <text x="10"
39         y="120"
40         font-size="20">This is svgExample_1.html</text>
41     <rect x="10"
42         y="10"
43         width="100"
44         height="75"
45         style="stroke: #000000; fill: #eeeeee;"
46         onmouseover="change(this, 'in' );"
47         onmouseout  ="change(this, 'out' );"
48         onclick      ="change(this, 'on' )"/>
49 </svg>
50
51 </body>
52
53 </html>
54
```

File Edit Search Preferences Shell Macro Windows Help

svgExample_0.html svgExample_1.html svgExample_2.html

/Users/menasce/Documents/Powerpoint/Conferenze/CORSO SPECIALISTICO/svgExample_1.html byte 712 of 1319 L: 25 C: 41

```
1 <html>
2   <head>
3     <meta charset="UTF-8">
4     <title>An SVG example (svgExample_1.html)</title>
5   <script>
6     function change(object, direction)
7     {
8       if(      direction == 'in'  )
9       {
10         object.style.stroke      = '#ff0000' ;
11         object.style.fill        = '#fce37d' ;
12         object.width.baseVal.value = '150' ;
13         object.x.baseVal.value   = '50'   ;
14       }
15     else if( direction == 'out' )
16     {
17       object.style.stroke      = '#000000' ;
18       object.style.fill        = '#eeeeee' ;
19       object.width.baseVal.value = '100' ;
20       object.x.baseVal.value   = '10'   ;
21     }
22     else if( direction == 'on' )
23     {
24       object.style.stroke      = '#000000' ;
25       object.style.fill        = '#654321' ;
26       object.width.baseVal.value = '400' ;
27       object.x.baseVal.value   = '100' ;
28     }
29   }
30 </script>
31 </head>
32 <body>
33   <body>
34     <svg width="300px"
35       height="300px"
36       onmouseover="change(this, 'in' );"
37       onmouseout  ="change(this, 'out' );"
38       onclick     ="change(this, 'on' )"/>
```



This is svgExample_1.html

File Edit Search Preferences Shell Macro Windows Help

svgExample_0.html svgExample_1.html **svgExample_2.html**

/Users/menasce/Documents/Powerpoint/Conferenze/CORSO_SPECIALISTICO/svgExample_2.html 1436 bytes

L: --- C: ---

```
20     object.x.baseVal.value      = '10'      ;
21   }
22 }
23 </script>
24 </head>
25
26 <body>
27
28 <svg width="300px"
29   height="300px"
30   xmlns="http://www.w3.org/2000/svg">
31
32 <defs>
33   <filter id="shadow" x="0" y="0" width="200%" height="200%">
34     <feOffset result="offOut" in="SourceGraphic" dx="5" dy="5" />
35     <feGaussianBlur result="blurOut" in="offOut" stdDeviation="3" />
36     <feBlend in2="blurOut" in="SourceGraphic" mode="normal" />
37   </filter>
38
39 </defs>  <text x="10"
40   y="120"
41   font-size="20">This is svgExample_2.html</text>
42 <rect filter="url(#shadow)"
43   x="10"
44   y="10"
45   width="100"
46   height="75"
47   style="stroke: #000000; fill: #eeeeee;"
48   onmouseover="change(this, 'in');"
49   onmouseout = "change(this, 'out');" />
50
51 </svg>
52
53 </body>
54
55 </html>
```

Il protocollo SVG accetta una ricchissima serie di attributi grafici: in questo esempio vediamo come fornire un'ombra ad un oggetto:





This is `svgExample_2.html`

È ovvio come sia possibile e relativamente facile descrivere figure semplici mediante opportuni tag SVG. Ma è altrettanto ovvio come questo approccio sia praticamente inutilizzabile per figure più complesse o estremamente complesse.

Ecco allora venirci in aiuto uno strumento indispensabile per chi abbia necessità di rappresentare oggetti bidimensionali (ma non solo) mediante grafica vettoriale su WEB:

Inkscape



INKSCAPE

Draw Freely.

Follow us on:

Search



ABOUT

DOWNLOAD

NEWS

COMMUNITY

LEARN

CONTRIBUTE

DEVELOP

DONATE



Inkscape is a professional vector graphics editor for Windows, Mac OS X and Linux. It's free and open source.

Download

Current stable version: 0.48.5

One Day in Farm by RebirthArt

Overview



What is Inkscape and how can you get it?

Features



Find out what Inkscape is capable of.

Gallery



Showcase of creations from the Inkscape community.

Learning Resources

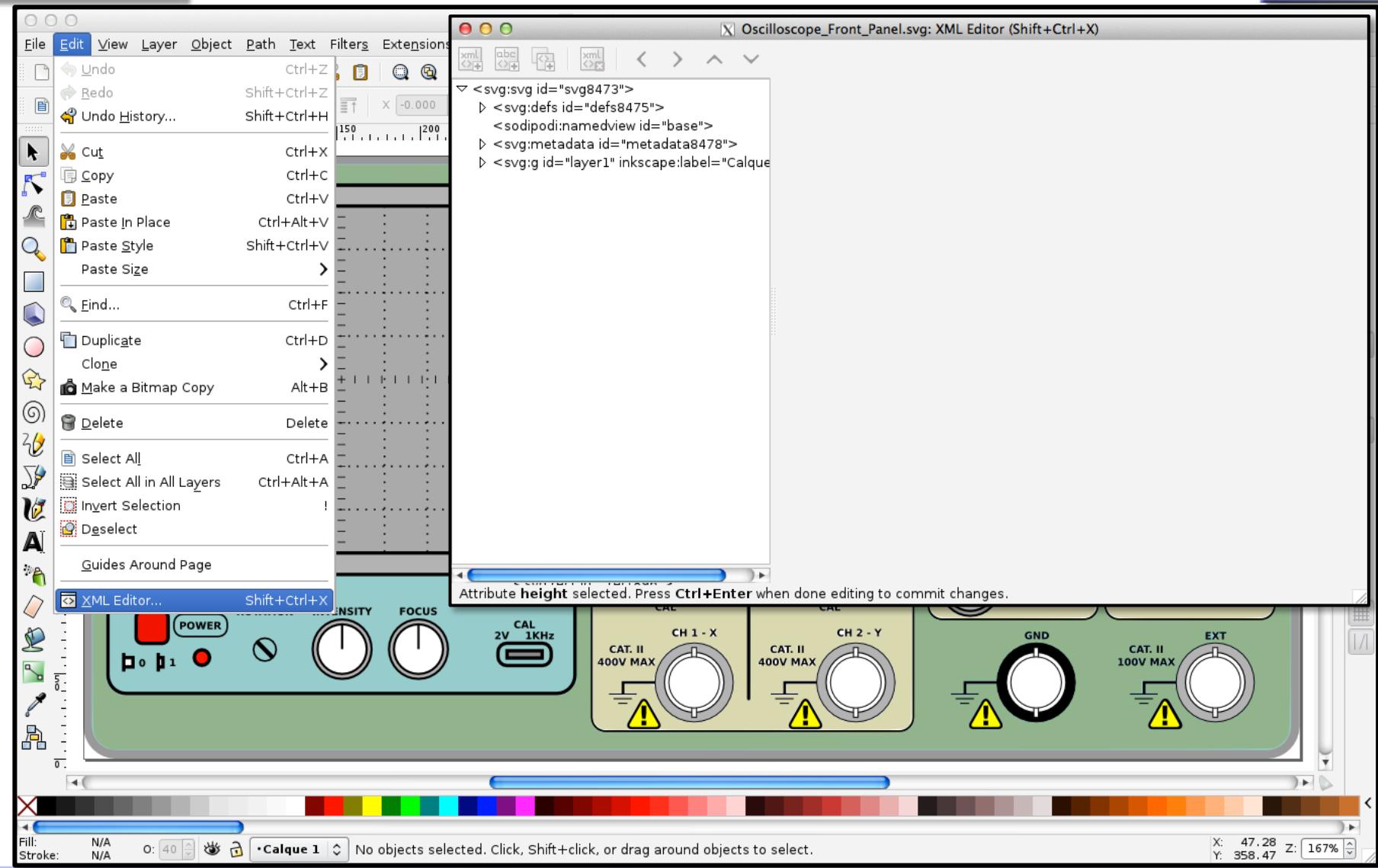


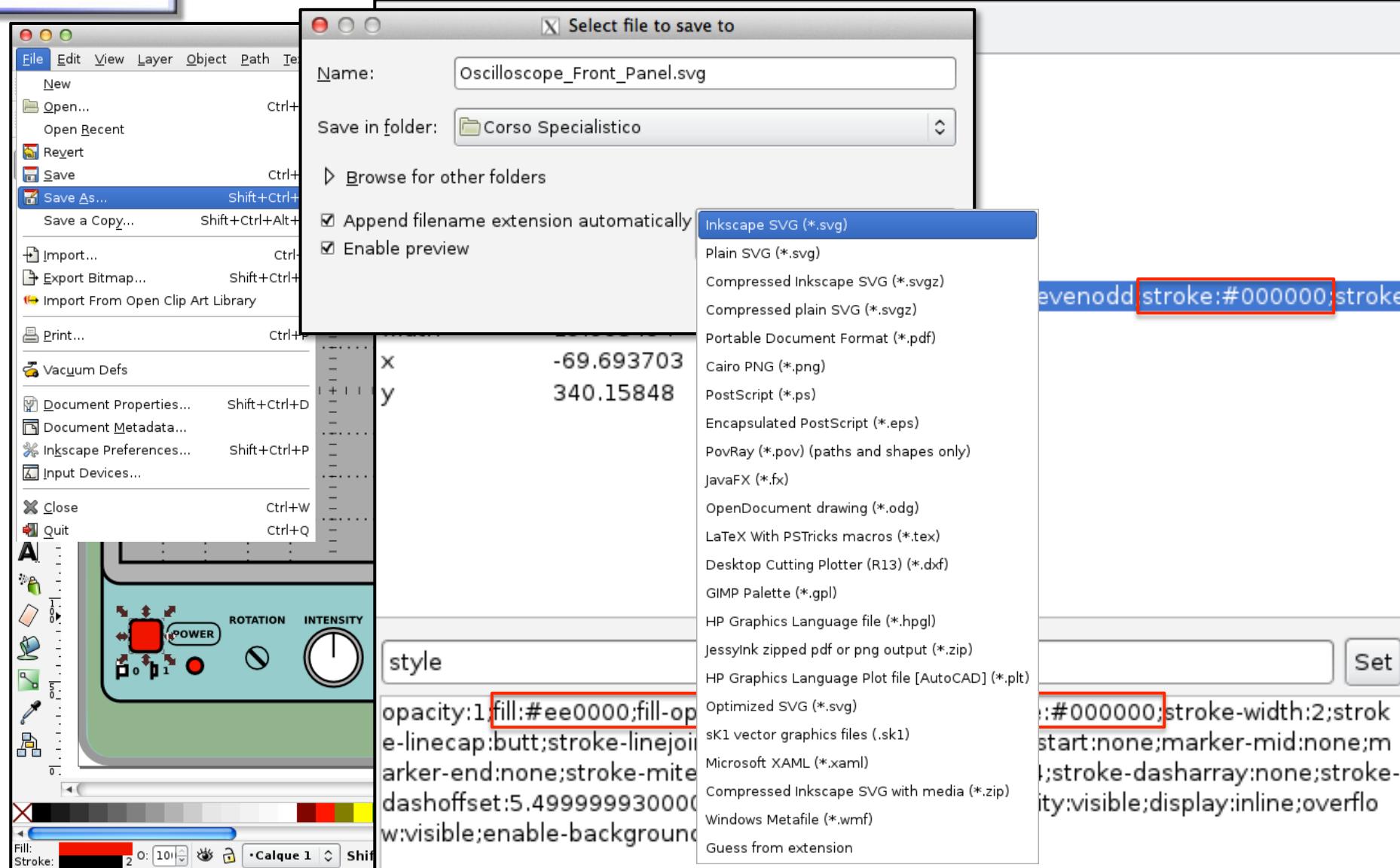
Resources to help you get the most out of Inkscape.

Users

Developers

Recent News





```
109      sodipodi:cx="100.23888"
110      id="path12758"
111      style="opacity:1;fill:none;fill-opacity:1;fill-rule:evenodd;stroke:#0000
112      sodipodi:type="arc" />
113    <path
114      id="path12760"
115      d="M 2.2208818,358.25408 L 11.232728,367.26595"
116      style="opacity:1;fill:none;fill-opacity:1;fill-rule:evenodd;stroke:#0000
117    </g>
118    <!-- The power switch button -->
119    <rect
120      rx="3.4723992"
121      ry="3.4723992"
122      y="340.15848"
123      x="-69.693703"
124      id="powerButton"
125      height="18.940359"
126      width="19.003494"
127      style="opacity:1;fill:#ee0000;fill-opacity:1;fill-rule:evenodd;stroke:#000
128      onmouseover="this.style.stroke='#ffaa00';"
129      onmouseout="this.style.stroke='#000000';"
130      onclick="power(evt);"/>
131    <!-- The power switch button -->
132    <use
133      height="354.87555"
134      width="722.13477"
135      transform="translate(-145.06288,227.91567)"
136      id="use12696"
137      xlink:href="#Small_Knob_2_"
138      y="0"
139      x="0" />
140    <rect
141      style="opacity:1;fill:none;fill-opacity:1;fill-rule:evenodd;stroke:#000000
142      id="rect12666"
143      width="277.2869"
144      height="299.00516"
145      x="14.647212"
```

Più di 2000 linee...

Ma questo è un file in puro formato **svg** (altrimenti **Inkscape** non sarebbe in grado di gestirlo), per cui dove sta il codice JavaScript corrispondente alla funzione **power(evt)**?

```
1 <!-- Author D. Menasce
2   Example of a rather complex, interactive, svg drawing -->
```

```
3
4 <html>      Ovviamente in un file HTML!
```

```
5   <head> ... file che contiene, nello head, gli
6     <meta charset="UTF-8"> script di controllo, e, nel body, il
7     <title>An Oscilloscope example</title> codice svg prodotto con Inkscape
8     <script>
```

```
9       function power(evt)
10      {
11        var osat, pbat, ldat ;
12        var os = document.getElementById('oscilloscopeScreen') ;
13        var pb = document.getElementById('powerButton') ;
14        var ld = document.getElementById('LED') ;
15        osat = os.getAttribute("style") ;
16        pbat = pb.getAttribute("style") ;
17        ldat = ld.getAttribute("style") ;
18        if( pbат.match(/fill: rgb\((238, 0, 0\))/) )
19        {
20          osat = osat.replace(/#aaaaaa/, '#68cd3b') ;
21          pbат = pbат.replace(/fill: rgb\((238, 0, 0\)\)/, 'fill: rgb(0, 238, 0)' ) ;
22          ldat = ldat.replace(/fill: rgb\((238, 0, 0\)\)/, 'fill: rgb(0, 238, 0)' ) ;
23        }
24        else if(pbat.match(/fill: rgb\((0, 238, 0\)\)/) )
25        {
26          osat = osat.replace('#68cd3b', '#aaaaaa') ;
27          pbат = pbат.replace(/fill: rgb\((0, 238, 0\)\)/, 'fill: rgb(238, 0, 0)' ) ;
28          ldat = ldat.replace(/fill: rgb\((0, 238, 0\)\)/, 'fill: rgb(238, 0, 0)' ) ;
29        }
30        pb.setAttribute('style',pbат) ;
31        ld.setAttribute('style',ldat) ;
32        os.setAttribute('style',osat) ;
33      }
34    </script>
35  </head>
```

```
36
37 <body>
```

File Edit Search Preferences Shell Macro Windows Help

Oscilloscope_Front_Panel.html - /Users/menasce/Documents/Powerpoint/Conferenze/CORSO SPECIALISTICO/Oscilloscope_Front_Panel.html 200729 bytes

L: --- C: ---

```
37 <body>
38 <?xml version="1.0" encoding="UTF-8" standalone="no"?>
39 <!-- Created with Inkscape by D. Menasce -->
40
41 <svg
42   xmlns:dc="http://purl.org/dc/elements/1.1/"
43   xmlns:cc="http://creativecommons.org/ns#"
44   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
45   xmlns:svg="http://www.w3.org/2000/svg"
46   xmlns="http://www.w3.org/2000/svg"
47   xmlns:xlink="http://www.w3.org/1999/xlink"
48   xmlns:sodipodi="http://sodipodi.sourceforge.net/DTD/sodipodi-0.dtd"
49   xmlns:inkscape="http://www.inkscape.org/namespaces/inkscape"
50   width="722.13477"
51   height="354.87555"
52   id="svg8473"
53   sodipodi:version="0.32"
54   inkscape:version="0.48.2 r9819"
55   version="1.0"
56   sodipodi:docname="Oscilloscope_Front_Panel.svg"
57   inkscape:output_extension="org.inkscape.output.svg.inkscape">
58 <defs
59   id="defs8475">
60   <marker
61     inkscape:stockid="StopL"
62     orient="auto"
63     refY="0"
64     refX="0"
65     id="StopL"
66     style="overflow:visible">
67   <path
68     id="path5559"
69     d="M 0,5.65 L 0,-5.65"
70     style="fill:none;fill-opacity:0.75;stroke:#000000;stroke-width:1pt"
71     transform="scale(0.8,0.8)" />
72 </marker>
```

```

1 <!-- Author D. Menasce
2     Example of a rather complex, interactive, svg drawing -->
3
4 <html>
5   <head>
6     <meta charset="UTF-8">
7     <title>An Oscilloscope example</title>
8     <script>
9       function power(evt)
10      {
11        var osat, pbat, ldat ;
12        var os = document.getElementById('oscilloscope');
13        var pb = document.getElementById('powerButton');
14        var ld = document.getElementById('LED') ;
15        osat = os.getAttribute("style") ;
16        pbat = pb.getAttribute("style") ;
17        ldat = ld.getAttribute("style") ;
18        if( pbат.match(/fill: rgb\((238, 0, 0\)/) )
19        {
20          osat = osat.replace(/#aaaaaa/, '#68cd3b');
21          pbат = pbат.replace(/fill: rgb\((238, 0, 0\)/);
22          ldat = ldat.replace(/fill: rgb\((238, 0, 0\)/);
23        }
24        else if(pbat.match(/fill: rgb\((0, 238, 0\)/)
25        {
26          osat = osat.replace(/#aaaaaa/, '#68cd3b');
27          pbат = pbат.replace(/fill: rgb\((0, 238, 0\)/);
28          ldat = ldat.replace(/fill: rgb\((0, 238, 0\)/);
29        }
30        pb.setAttribute('style', ldat);
31        ld.setAttribute('style', osat);
32        os.setAttribute('style', pbат);
33      }
34   </script>
35 </head>
36 <body>
```

Torniamo ora allo **head...**

Quirk: nel testo (SVG in HTML) compare:

fill:#ee0000

ma in memoria è rappresentato come:

rgb(238, 0, 0)

(scoperto tramite Firebug, curioso...)

```

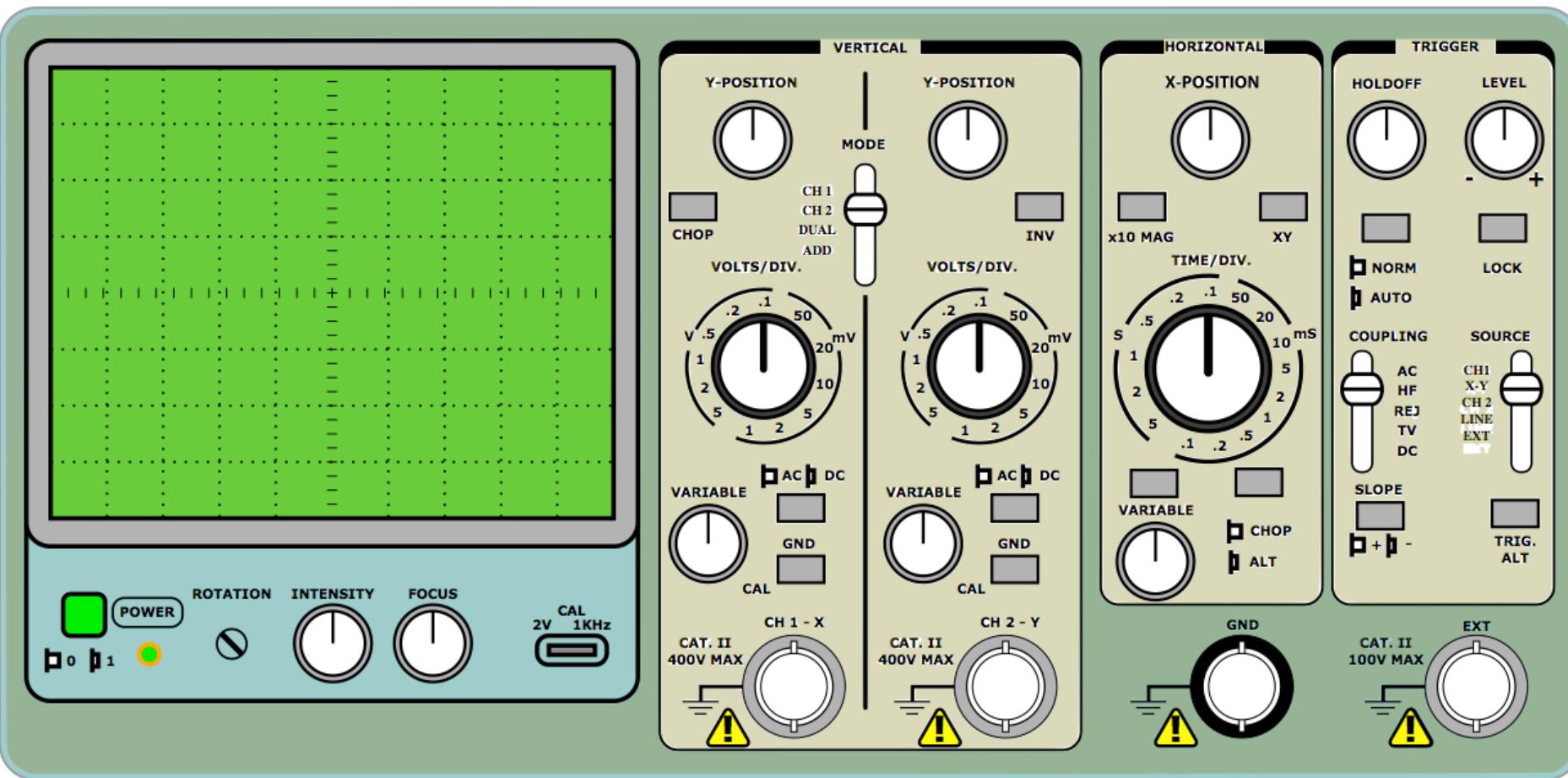
<rect
  rx="3.4723992"
  ry="3.4723992"
  y="340.15848"
  x="-69.693703"
  id="powerButton"
  height="18.940359"
  width="19.003494"
  style="opacity:1;fill:#ee0000;fill-opacity:1;
  onmouseover="this.style.stroke='#ffaa00';
  onmouseout="this.style.stroke='#000000';
  onclick='power(evt);' />
```

/Users/menasse/Documents/Powerpoint/Conference/Corse_Specialistiche/Oscilloscope_Front_Panel.html [svgExample_1.htm] [svgExample_2.htm]

Lipid Cycles

```
143 </a>
144 <!-- The power switch button -->
145 <rect rx="3.4723992"
146     ry="3.4723992"
147     y="340.15848"
148     x="-69.693703"
149     id="powerButton"
150     height="18.940359"
151     width="19.003494"
152     style="fill:#ee0000;stroke:#000000;stroke-width:2;visibility:visible;"
153     onmouseover="this.style.stroke='#ffaa00';"
154     onmouseout="this.style.stroke='#000000';"
155     onclick="power(evt);"
156     />
157 <!-- The power switch button -->
158 <use
159     height="354.87555"
160     width="722.13477"
161     transform="translate(-145.06288,227.91567)"
```

```
250 style="opacity:1;fill:#b3b3b3;stroke:#000000;stroke-width:2;stroke-linejoi  
251 <g  
252   id="g7712"  
253   transform="matrix(1.0128461,0,0,1.0128461,-99.673259,50.553784)">  
254     <!-- The oscilloscope screen -->  
255     <rect style="fill:#aaaaaa;stroke:#000000;stroke-width:2"  
256       id="oscilloscopeScreen"  
257       height="201.55099"  
258       width="252"  
259       y="50.548"  
260       x="23.969999"/>  
261     <!-- The oscilloscope screen -->  
262     <path  
263       sodipodi:nodetypes="cccccccccccccccccccccccccccccccccccc"
```



Vediamo ora come immergere un file **SVG** all'interno di un **panel** di **Extjs**, in modo di dotarlo di un più ampio spettro di possibilità di interazione con l'utente.

[INFN@MiB](#)[Notizie](#)[Eventi](#)[Linee di Ricerca](#)[Servizi](#)[Opportunita'](#)[Strumenti](#)[Collegamenti](#)

Istituto Nazionale di Fisica Nucleare - Sezione di Milano Bicocca

INFN@MiB

INFN - Sezione di Milano Bicocca



L'Istituto Nazionale di Fisica Nucleare (INFN) è l'ente dedicato allo studio dei costituenti fondamentali della materia e svolge attività di ricerca, teorica e sperimentale, nei campi della fisica subnucleare, nucleare e astroparticellare.

La ricerca fondamentale in questi settori richiede l'uso di tecnologie e strumenti di ricerca d'avanguardia che l'INFN sviluppa nei propri laboratori ed in collaborazione con il mondo dell'industria.

L'Istituto promuove inoltre il trasferimento delle competenze, delle metodologie e delle tecniche strumentali sviluppate nell'ambito della propria attività verso campi di ricerca diversi quali la medicina, i beni culturali e l'ambiente. Tutte queste attività si svolgono in stretta collaborazione con il mondo universitario.

L'INFN venne istituito l'8 agosto 1951 da gruppi delle Università di Roma, Padova, Torino e le Sezioni e i Laboratori Nazionali. Le Sezioni hanno sede in dipartimenti universitari e realizzano il collegamento diretto tra l'Istituto e le Università.

[INFN@MiB](#)[La Sezione](#)[Il Direttore](#)[Come contattarci](#)[Come raggiungerci](#)

Site news

- CMS closes major chapter of Higgs measurements
- Premio Nobel a Higgs e Englert

File Edit Search Preferences Shell Macro Windows Help

OscilloscopeWithControls.html - /user/gr1/e831/menasce/public_html/esercizi/wireless/htmlFiles/
OscilloscopeWithControls.html panExample.html svgExample_0.htm svgExample_1.htm svgExample_2.htm temp.html wireless_0.html wireless_1.html wireless_2.html wireless_3.html wireless_4.html wireless_4_withF wireless_5.html
/user/gr1/e831/menasce/public_html/esercizi/wireless/htmlFiles/OscilloscopeWithControls.html byte 365 of 828 L: 11 C: 69

```
1 <!-- Author D. Menasce
2     Example of a rather complex, interactive, svg drawing -->
3
4 <html>
5
6   <head>
7     <meta charset="UTF-8">
8     <title>An Oscilloscope example</title>
9     <link rel="stylesheet"
10        type="text/css"
11        href="../../extjsLib/examples/shared/example.css"      />
12     <link rel="stylesheet"
13        type="text/css"
14        href="../cssFiles/wireless.css"                      />
15     <script type="text/javascript"
16        src="../../extjsLib/examples/shared/include-ext.js"  ></script>
17     <script type="text/javascript"
18        src="../../extjsLib/examples/shared/options-toolbar.js"></script>
19     <script type="text/javascript"
20        src="../jsFiles/Oscilloscope.js"                    ></script>
21   </head>
22
23   <body>
24   </body>
25
26 </html>
27
```

File Edit Search Preferences Shell Macro Windows Help

OscilloscopeWithControls Oscilloscope.js

/raid2/users/menasce/public_html/dottorato/esercizi/wireless/jsFiles/Oscilloscope.js 2501 bytes L: --- C: ---

```
1 /*=====
2  Author: D. Menasce
3
4  Technique to embed a scriptable svg element
5  into an extjs panel, plus histograms and video
6 =====*/
7
8 Ext.onReady
9 (
10 function()
11 {
12     var showPlot = function ()
13     {
14         var htm = '<embed src="../svgFiles/ratioPlot.svg" ' +
15             'type="image/svg+xml" />' ;
16         var win = new Ext.Window
17             (
18                 {
19                     title      : 'Data plot',
20                     width     : 540
21                     height    : 560
22                     layout    : 'fit'
23                     collapsible: true
24                     closable   : true
25                     html       : htm
26                 }
27             );
28         win.show();
29         win.setPosition(
30             310,100,
31             {
32                 duration: 6000,
33                 easing  : 'elasticIn'
34             }
35         );
36     }
}
```

File Edit Search Preferences Shell Macro Windows Help

OscilloscopeWithControls Oscilloscope.js

/raid2/users/menasce/public_html/dottorato/esercizi/wireless/jsFiles/Oscilloscope.js byte 2493 of 2508 L: 101 C: 0

```

66 var htm = '<embed src="../svgFiles/Oscilloscope.svg" ' +
67   'type="image/svg" />' ;
68 var win = new Ext.Window(
69   (
70     {
71       title      : 'Oscilloscope control panel',
72       width     : 732
73       height    : 436
74       layout    : 'fit'
75       collapsible: true
76       closable   : false
77       tbar      : [
78         (
79           {
80             text  : 'Show data plot',
81             handler: showPlot
82           },
83           {
84             text  : 'Show movie',
85             handler: showMovie
86           },
87         ],
88       html      : htm
89     }
90   );
91 win.show();
92 win.setPosition(
93   10,10,
94   {
95     duration: 6000,
96     easing  : 'elasticIn'
97   }
98 );
99 )
100 
```

20/06/17

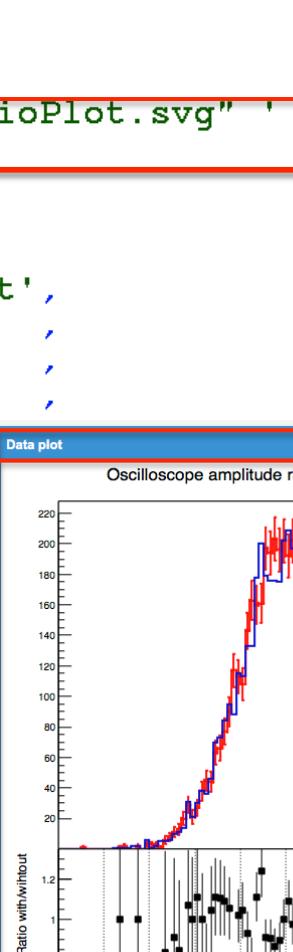
Dario Menasce - Introduzione al CGI Web Programming

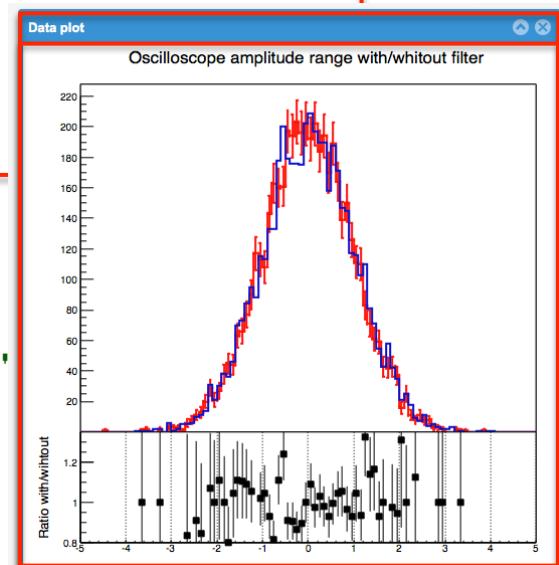
24

/raid2/users/menasse/public_html/dottorato/eserciziAvireless/icFile/Oscilloscope.js 2501 byte

L---C---

```
8 Ext.onReady
9 (
10 function()
11 {
12     var showPlot = function ()
13     {
14         var htm = '<embed src="../../svgFiles/ratioPlot.svg" ' +
15             'type="image/svg+xml" />' ;
16         var win = new Ext.Window
17             (
18                 {
19                     title      : 'Data plot',
20                     width     : 540
21                     ,
22                     height    : 560
23                     ,
24                     layout    : 'fit'
25                     ,
26                     collapsible: true
27                     ,
28                     closable   : true
29                     ,
30                     html       : htm
31                 }
32             );
33         win.show();
34         win.setPosition(
35             310,100,
36             {
37                 duration: 6000,
38                 easing  : 'elasticIn'
39             }
40         );
41         var showMovie = function ()
42         {
43             var htm = ' <video controls>
44             ' <source src="../../images/Osc1.ogv" ' +
45                 ' type="video/ogg">
46             '</video>
```





File Edit Search Preferences Shell Macro Windows Help

OscilloscopeWithControls Oscilloscope.js

/raid2/users/menasce/public_html/dottorato/esercizi/wireless/jsFiles/Oscilloscope.js byte 535 of 2501 L: 20 C: 68

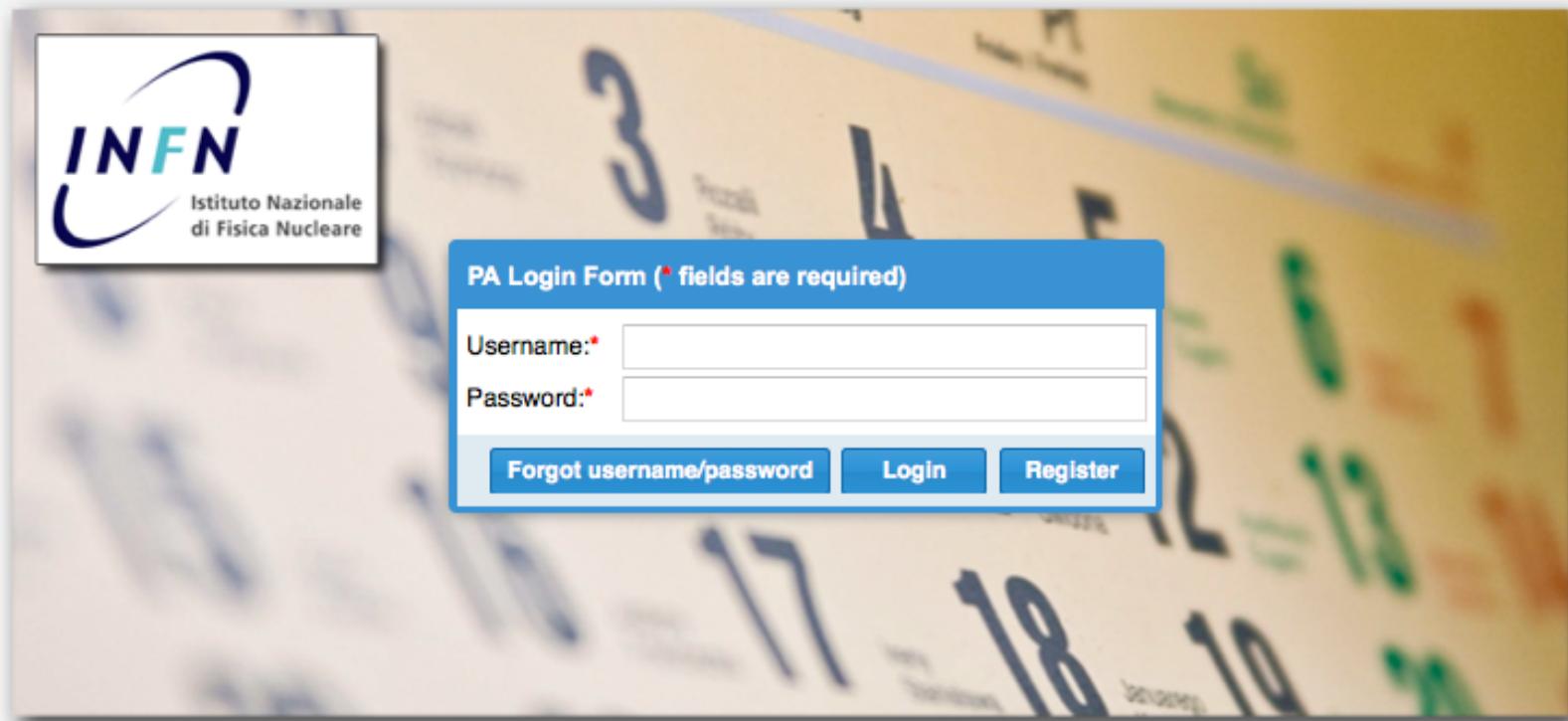
```
38 var showMovie = function ()
39 {
40     var htm = '<video controls>
41             <source src="../images/Osc1.ogv">
42             <type="video/ogg">
43             </video>
44     var win = new Ext.Window
45         (
46             {
47                 title      : 'Demo Movie',
48                 width     : 858,
49                 height    : 520,
50                 layout    : 'fit',
51                 collapsible: true,
52                 closable   : true,
53                 html       : htm
54             }
55         );
56     win.show();
57     win.setPosition(
58         11
59     {
60         d
61         e
62     }
63     );
64 }

66 var htm = '<embed src="../svgFiles/
67             &type="image/svg" />';
68 var win = new Ext.Window
69     (
70         {
71             title      : 'Oscil
72             width     : 732,
73             height    : 436
```

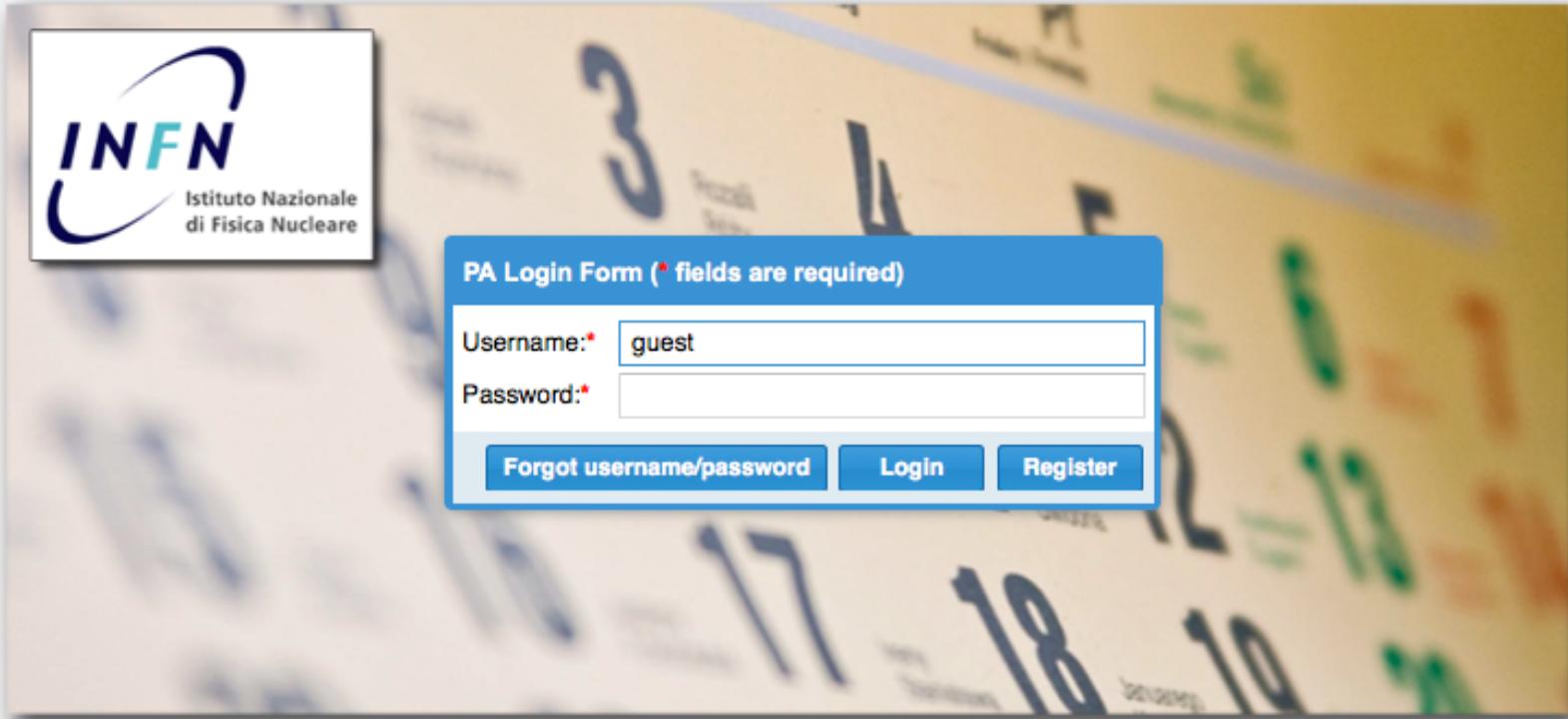
I cookies



I **cookies** sono uno strumento inventato per sopperire alla caratteristica **stateless** di una connessione **http**. Permettono di rendere persistente un certo numero di informazioni tra transazioni diverse (strumento utile in limitati contesti).



Vediamo un possibile caso d'uso: si tratta di un'applicazione totalmente priva di caratteristiche di sicurezza, la presentiamo unicamente come supporto pratico ad una introduzione al tema dei **cookies**.

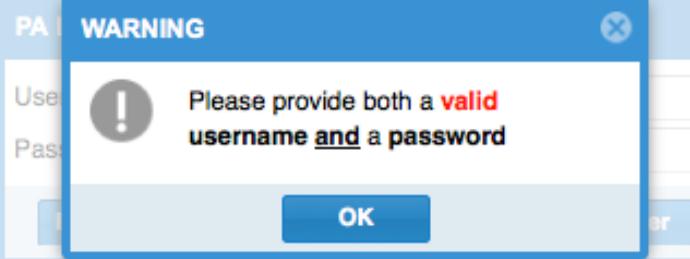


PA Login Form (* fields are required)

Username: *

Password:

[Forgot username/password](#) [Login](#) [Register](#)



The image shows a web browser window with a login form. The background of the page is a photograph of a particle detector's inner structure, featuring various pipes, wires, and a large number '3'.

PA Login Form (* fields are required)

Username: *

Password: *

[Forgot username/password](#) [Login](#) [Register](#)

The login form has a blue header and a white body. The 'Username' field contains the value 'guest'. The 'Password' field is empty and has a red border, indicating it is a required field. Below the form are three buttons: 'Forgot username/password', 'Login', and 'Register'.



PA Login Form (* fields are required)

Username:^{*} Password:^{*} [Forgot username/password](#)[Login](#)[Register](#)



Form di Prenotazione Auto INFN per utenti

Sezione/laboratorio INFN: Milano Celoria

Submit

Reset

Logout

Modulo di richiesta

Nome:

Cognome:

Auto:

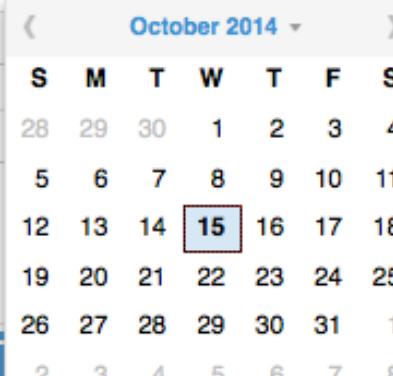
Selezione un modello

Email:

Motivazione:

Helvetica

B I U T^{*} T^{**} T^{***}



Today

10/15/2014

Ricerche

Nome:

Cognome:

Disponibilità (per mese)

Sommario (data)

Sommario (per auto)

Cerca utente

Sommario (completo)

Rimuovi utente

Rimuovi (per data)

Rimuovi (per auto)

```
[menasce@hal9000 PA]$ ll
total 8
drwxr-xr-x 7 menasce e831      80 Aug  8 09:04 .
drwxr-xr-x 6 menasce e831    4096 Aug  8 08:47 ..
drwxr-xr-x 2 menasce e831      24 Sep  3 14:32 cssFiles
-rw-r--r-- 1 menasce e831     319 Aug  8 12:08           → toolbar.css
drwxr-xr-x 2 menasce e831     101 Aug  8 12:11 htmlFiles
-rw-r--r-- 1 menasce e831    1049 Aug  8 11:59           → controlPanelSuper.html
-rw-r--r-- 1 menasce e831    1135 Aug  8 12:00           → controlPanelUser.html
-rw-r--r-- 1 menasce e831    1243 Aug  8 12:11           → PA.html
drwxr-xr-x 2 menasce e831      48 Aug  8 08:49 images
-rw-r--r-- 1 menasce e831 348336 Jul 25 15:41           → calendario.png
drwxr-xr-x 2 menasce e831    4096 Aug  8 12:57 jsFiles
-rw-r--r-- 1 menasce e831    2267 Aug  8 12:52           → balloons.js
-rw-r--r-- 1 menasce e831    1295 Sep  3 14:19           → commonStuff.js
-rw-r--r-- 1 menasce e831   16040 Aug  8 10:01           → controlPanelSuper.js
-rw-r--r-- 1 menasce e831   16218 Aug  8 12:57           → controlPanelUser.js
-rw-r--r-- 1 menasce e831    2588 Aug  8 09:51           → cookies.js
-rw-r--r-- 1 menasce e831    8084 Aug  8 12:44           → loginManager.js
drwxr-xr-x 2 menasce e831      25 Aug  8 10:30 perlFiles
-rwxr-xr-x 1 menasce e831    2267 Aug  8 10:42           → authorize.pl
```

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl commonStuff.js controlPanelUser.js toolbar.css balloons.js controlPanelSuper.js cookies.js loginManager.js

/user/gr1/e831/menasce/public_html/esercizi/PA/htmlFiles/PA.html byte 821 of 1191 L: 13 C: 65

```
1 <html>
2
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
5   <title>Form di login</title>
6   <link href="../../extjsLib/examples/shared/example.css" rel="stylesheet"
7   <link href="../cssFiles/toolbar.css" rel="stylesheet"
8   <script src="../../extjsLib/examples/shared/include-ext.js" type="text/javascript"
9   <script src="../../extjsLib/examples/shared/examples.js" type="text/javascript"
10  <script src="../jsFiles/commonStuff.js" type="text/javascript"
11  <script src="../jsFiles/cookies.js" type="text/javascript"
12  <script src="../jsFiles/balloons.js" type="text/javascript"
13  <script src="../jsFiles/loginManager.js" type="text/javascript"
14 <style>
15   #docbody
16   {
17     background-image : url('../images/calendario.png');
18     background-repeat : no-repeat;
19     background-attachment: fixed ;
20     background-position : center ;
21   }
22 </style>
23 </head>
24
25 <body id="docbody">
26   <div id="content" class="x-hidden">
27   </div>
28 </body>
29
30 </html>
31
```

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl commonStuff.js controlPanelUser.js toolbar.css balloons.js controlPanelSuper.js cookies.js loginManager.js /user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/loginManager.js byte 0 of 8084 L: 1 C: 0

```
1 ****
2 File : loginManager.js
3 Autors : D. Menasce
4 Date : Jul 2014
5
6 Purpose: example of cookies usage for authorization
7 ****
8
9 Ext.require([
10     'Ext.form.*',
11     'Ext.layout.container.Column',
12     'Ext.tab.Panel',
13     'PA.tools.balloons.msg'
14 ]
15 );
16
17 Ext.ns('PA.loginManager') ;
18
19 PA.controlPanel = function(userType)
20 {
21     var hrefLoc = PA.get.htmlName() ;
22     hrefLoc += '/controlPanel' + userType + '.html' ;
23     window.location = hrefLoc ;
24 }
25
26 Ext.onReady ←
27 (
28     function()
29     {
30         var mX = 340 ;
31         var mY = 300 ;
32
33         Ext.QuickTips.init();
34         if( PA.Cookies['PAUserName'] &&
35             PA.Cookies['PAUserName'].match(/\w+/))
```

22/06/17{ Dario Menasce - Introduzione al CGI Web Programming 37

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl commonStuff.js controlPanelUser.js toolbar.css balloons.js controlPanelSuper.js cookies.js loginManager.js L: --- C: ---

/user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/loginManager.js 8084 bytes

```
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283 PA.loginManager.window = Ext.create(
284
285
286
287
288
289
290
291
292
293
294
295
296 PA.loginManager.window.show() ;
297
298 )
299
300
```

The screenshot shows a code editor with a red box highlighting the window configuration in line 283. Inside this box, another red box highlights the title bar of the login form window. The window itself is displayed above the code editor, showing a blue header with the title "PA Login Form (* fields are required)". It contains two text input fields for "Username" and "Password", and three buttons: "Forgot username/password", "Login", and "Register".

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl commonStuff.js controlPanelUser.js toolbar.css balloons.js controlPanelSuper.js cookies.js loginManager.js L: --- C: ---

/user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/loginManager.js 8084 bytes

```
246     allowBlank : false,
247     listeners :
248     specialkey: function(field, e)
249     {
250         if (e.getKey() == e.ENTER)
251         {
252             pwdField.focus();
253         }
254     }
255 );
256
257
258
259 var simpleForm = Ext.widget({
260     bodyPadding : '5 5 5 5',
261     buttons : [
262         forgotBtn,
263         loginButton,
264         regBtn
265     ],
266     xtype : 'form',
267     layout : 'form',
268     id : 'simpleForm',
269     frame : false,
270     width : 360,
271     fieldDefaults: {
272         msgTarget : 'side',
273         labelWidth: 75
274     },
275     defaultType : 'textfield',
276     items : [
277         usrField,
278         pwdField
279     ],
280
281});
```

The screenshot shows a Java-based IDE interface. At the top, there's a menu bar with File, Edit, Search, Preferences, Shell, Macro, Windows, Help, and tabs for PA.html, controlPanelUser.html, authorize.pl, commonStuff.js, controlPanelUser.js, toolbar.css, balloons.js, controlPanelSuper.js, cookies.js, and loginManager.js. Below the menu, a status bar displays the path /user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/loginManager.js and the byte count 8084 bytes. The main area contains two parts: a visual representation of a login form and its corresponding JavaScript configuration code.

The visual representation of the login form is a blue-bordered window titled "PA Login Form (* fields are required)". It contains two input fields labeled "Username:" and "Password:", each with a red asterisk indicating it's required. Below these fields are three buttons: "Forgot username/password" (disabled), "Login" (disabled), and "Register".

The configuration code below the form is written in JavaScript using the ExtJS framework. It defines a variable "simpleForm" which is an instance of the Ext.widget function. The configuration object has several properties: "bodyPadding" set to '5 5 5 5', "buttons" as an array containing three buttons (forgotBtn, loginButton, regBtn), "xtype" set to 'form', "layout" set to 'form', "id" set to 'simpleForm', "frame" set to false, "width" set to 360, and "fieldDefaults" as an object with "msgTarget" set to 'side' and "labelWidth" set to 75. The "defaultType" is set to 'textfield'. The "items" property is an array containing two fields: "usrField" and "pwdField". The entire configuration is enclosed in a pair of parentheses at the end of the line.

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl commonStuff.js controlPanelUser.js toolbar.css balloons.js controlPanelSuper.js cookies.js loginManager.js L: --- C: ---

/user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/loginManager.js 8084 bytes

```
153 var loginButton = Ext.create('Ext.button.Button', {
154     text: 'Login',
155     renderTo: '#loginForm',
156     width: 100,
157     height: 30,
158     style: 'margin-top: 10px; margin-left: 10px;',
159     listeners: {
160         click: function() {
161             Ext.getCmp('username').setFocus();
162         }
163     }
164 });
165
166 var regBtn = {
167     text: 'Register',
168     tooltip: 'Use this button to register yourself as a PA service user',
169     handler: function() {
170         var text = '';
171         var subject = 'Requesting%20authorization';
172         var body = 'I would kindly ask for an PA account for the following username';
173         body = body.replace(/\s+/g, '%20');
174         text = 'Please send an email to <a href="mailto:dario.menasce@mib.infn.it" ?subject=';
175         subject = '&body=';
176         body = '">'>dario.menasce@mib.infn.it' '</a><br>';
177         text += 'You will be asked some questions....';
178         var a = Ext.Msg.alert('Suggestion', text);
179     }
180 };
181
182 Ext.onReady(function() {
183     Ext.create('Ext.form.Panel', {
184         title: 'PA Login Form (* fields are required)',
185         width: 300,
186         height: 200,
187         renderTo: '#loginForm',
188         items: [
189             {
190                 type: 'text',
191                 name: 'username',
192                 fieldLabel: 'Username:*',
193                 width: 200,
194                 height: 30
195             },
196             {
197                 type: 'password',
198                 name: 'password',
199                 fieldLabel: 'Password:*',
200                 width: 200,
201                 height: 30
202             }
203         ],
204         buttons: [
205             {
206                 text: 'Forgot username/password',
207                 width: 150
208             },
209             {
210                 text: 'Login',
211                 width: 100
212             },
213             {
214                 text: 'Register',
215                 width: 100
216             }
217         ]
218     });
219 });
220
221 Ext.onReady(function() {
222     Ext.create('Ext.form.Panel', {
223         title: 'PA Login Form (* fields are required)',
224         width: 300,
225         height: 200,
226         renderTo: '#loginForm',
227         items: [
228             {
229                 type: 'text',
230                 name: 'username',
231                 fieldLabel: 'Username:*',
232                 width: 200,
233                 height: 30
234             },
235             {
236                 type: 'password',
237                 name: 'password',
238                 fieldLabel: 'Password:*',
239                 width: 200,
240                 height: 30
241             }
242         ],
243         buttons: [
244             {
245                 text: 'Forgot username/password',
246                 width: 150
247             },
248             {
249                 text: 'Login',
250                 width: 100
251             },
252             {
253                 text: 'Register',
254                 width: 100
255             }
256         ]
257     });
258 });
259
260 Ext.onReady(function() {
261     Ext.create('Ext.form.Panel', {
262         title: 'PA Login Form (* fields are required)',
263         width: 300,
264         height: 200,
265         renderTo: '#loginForm',
266         items: [
267             {
268                 type: 'text',
269                 name: 'username',
270                 fieldLabel: 'Username:*',
271                 width: 200,
272                 height: 30
273             },
274             {
275                 type: 'password',
276                 name: 'password',
277                 fieldLabel: 'Password:*',
278                 width: 200,
279                 height: 30
280             }
281         ],
282         buttons: [
283             {
284                 text: 'Forgot username/password',
285                 width: 150
286             },
287             {
288                 text: 'Login',
289                 width: 100
290             },
291             {
292                 text: 'Register',
293                 width: 100
294             }
295         ]
296     });
297 });
298
299 Ext.onReady(function() {
300     Ext.create('Ext.form.Panel', {
301         title: 'PA Login Form (* fields are required)',
302         width: 300,
303         height: 200,
304         renderTo: '#loginForm',
305         items: [
306             {
307                 type: 'text',
308                 name: 'username',
309                 fieldLabel: 'Username:*',
310                 width: 200,
311                 height: 30
312             },
313             {
314                 type: 'password',
315                 name: 'password',
316                 fieldLabel: 'Password:*',
317                 width: 200,
318                 height: 30
319             }
320         ],
321         buttons: [
322             {
323                 text: 'Forgot username/password',
324                 width: 150
325             },
326             {
327                 text: 'Login',
328                 width: 100
329             },
330             {
331                 text: 'Register',
332                 width: 100
333             }
334         ]
335     });
336 });
337
338 Ext.onReady(function() {
339     Ext.create('Ext.form.Panel', {
340         title: 'PA Login Form (* fields are required)',
341         width: 300,
342         height: 200,
343         renderTo: '#loginForm',
344         items: [
345             {
346                 type: 'text',
347                 name: 'username',
348                 fieldLabel: 'Username:*',
349                 width: 200,
350                 height: 30
351             },
352             {
353                 type: 'password',
354                 name: 'password',
355                 fieldLabel: 'Password:*',
356                 width: 200,
357                 height: 30
358             }
359         ],
360         buttons: [
361             {
362                 text: 'Forgot username/password',
363                 width: 150
364             },
365             {
366                 text: 'Login',
367                 width: 100
368             },
369             {
370                 text: 'Register',
371                 width: 100
372             }
373         ]
374     });
375 });
376
377 Ext.onReady(function() {
378     Ext.create('Ext.form.Panel', {
379         title: 'PA Login Form (* fields are required)',
380         width: 300,
381         height: 200,
382         renderTo: '#loginForm',
383         items: [
384             {
385                 type: 'text',
386                 name: 'username',
387                 fieldLabel: 'Username:*',
388                 width: 200,
389                 height: 30
390             },
391             {
392                 type: 'password',
393                 name: 'password',
394                 fieldLabel: 'Password:*',
395                 width: 200,
396                 height: 30
397             }
398         ],
399         buttons: [
400             {
401                 text: 'Forgot username/password',
402                 width: 150
403             },
404             {
405                 text: 'Login',
406                 width: 100
407             },
408             {
409                 text: 'Register',
410                 width: 100
411             }
412         ]
413     });
414 });
415
416 Ext.onReady(function() {
417     Ext.create('Ext.form.Panel', {
418         title: 'PA Login Form (* fields are required)',
419         width: 300,
420         height: 200,
421         renderTo: '#loginForm',
422         items: [
423             {
424                 type: 'text',
425                 name: 'username',
426                 fieldLabel: 'Username:*',
427                 width: 200,
428                 height: 30
429             },
430             {
431                 type: 'password',
432                 name: 'password',
433                 fieldLabel: 'Password:*',
434                 width: 200,
435                 height: 30
436             }
437         ],
438         buttons: [
439             {
440                 text: 'Forgot username/password',
441                 width: 150
442             },
443             {
444                 text: 'Login',
445                 width: 100
446             },
447             {
448                 text: 'Register',
449                 width: 100
450             }
451         ]
452     });
453 });
454
455 Ext.onReady(function() {
456     Ext.create('Ext.form.Panel', {
457         title: 'PA Login Form (* fields are required)',
458         width: 300,
459         height: 200,
460         renderTo: '#loginForm',
461         items: [
462             {
463                 type: 'text',
464                 name: 'username',
465                 fieldLabel: 'Username:*',
466                 width: 200,
467                 height: 30
468             },
469             {
470                 type: 'password',
471                 name: 'password',
472                 fieldLabel: 'Password:*',
473                 width: 200,
474                 height: 30
475             }
476         ],
477         buttons: [
478             {
479                 text: 'Forgot username/password',
480                 width: 150
481             },
482             {
483                 text: 'Login',
484                 width: 100
485             },
486             {
487                 text: 'Register',
488                 width: 100
489             }
490         ]
491     });
492 });
493
494 Ext.onReady(function() {
495     Ext.create('Ext.form.Panel', {
496         title: 'PA Login Form (* fields are required)',
497         width: 300,
498         height: 200,
499         renderTo: '#loginForm',
500         items: [
501             {
502                 type: 'text',
503                 name: 'username',
504                 fieldLabel: 'Username:*',
505                 width: 200,
506                 height: 30
507             },
508             {
509                 type: 'password',
510                 name: 'password',
511                 fieldLabel: 'Password:*',
512                 width: 200,
513                 height: 30
514             }
515         ],
516         buttons: [
517             {
518                 text: 'Forgot username/password',
519                 width: 150
520             },
521             {
522                 text: 'Login',
523                 width: 100
524             },
525             {
526                 text: 'Register',
527                 width: 100
528             }
529         ]
530     });
531 });
532
533 Ext.onReady(function() {
534     Ext.create('Ext.form.Panel', {
535         title: 'PA Login Form (* fields are required)',
536         width: 300,
537         height: 200,
538         renderTo: '#loginForm',
539         items: [
540             {
541                 type: 'text',
542                 name: 'username',
543                 fieldLabel: 'Username:*',
544                 width: 200,
545                 height: 30
546             },
547             {
548                 type: 'password',
549                 name: 'password',
550                 fieldLabel: 'Password:*',
551                 width: 200,
552                 height: 30
553             }
554         ],
555         buttons: [
556             {
557                 text: 'Forgot username/password',
558                 width: 150
559             },
560             {
561                 text: 'Login',
562                 width: 100
563             },
564             {
565                 text: 'Register',
566                 width: 100
567             }
568         ]
569     });
570 });
571
572 Ext.onReady(function() {
573     Ext.create('Ext.form.Panel', {
574         title: 'PA Login Form (* fields are required)',
575         width: 300,
576         height: 200,
577         renderTo: '#loginForm',
578         items: [
579             {
580                 type: 'text',
581                 name: 'username',
582                 fieldLabel: 'Username:*',
583                 width: 200,
584                 height: 30
585             },
586             {
587                 type: 'password',
588                 name: 'password',
589                 fieldLabel: 'Password:*',
590                 width: 200,
591                 height: 30
592             }
593         ],
594         buttons: [
595             {
596                 text: 'Forgot username/password',
597                 width: 150
598             },
599             {
600                 text: 'Login',
601                 width: 100
602             },
603             {
604                 text: 'Register',
605                 width: 100
606             }
607         ]
608     });
609 });
610
611 Ext.onReady(function() {
612     Ext.create('Ext.form.Panel', {
613         title: 'PA Login Form (* fields are required)',
614         width: 300,
615         height: 200,
616         renderTo: '#loginForm',
617         items: [
618             {
619                 type: 'text',
620                 name: 'username',
621                 fieldLabel: 'Username:*',
622                 width: 200,
623                 height: 30
624             },
625             {
626                 type: 'password',
627                 name: 'password',
628                 fieldLabel: 'Password:*',
629                 width: 200,
630                 height: 30
631             }
632         ],
633         buttons: [
634             {
635                 text: 'Forgot username/password',
636                 width: 150
637             },
638             {
639                 text: 'Login',
640                 width: 100
641             },
642             {
643                 text: 'Register',
644                 width: 100
645             }
646         ]
647     });
648 });
649
650 Ext.onReady(function() {
651     Ext.create('Ext.form.Panel', {
652         title: 'PA Login Form (* fields are required)',
653         width: 300,
654         height: 200,
655         renderTo: '#loginForm',
656         items: [
657             {
658                 type: 'text',
659                 name: 'username',
660                 fieldLabel: 'Username:*',
661                 width: 200,
662                 height: 30
663             },
664             {
665                 type: 'password',
666                 name: 'password',
667                 fieldLabel: 'Password:*',
668                 width: 200,
669                 height: 30
670             }
671         ],
672         buttons: [
673             {
674                 text: 'Forgot username/password',
675                 width: 150
676             },
677             {
678                 text: 'Login',
679                 width: 100
680             },
681             {
682                 text: 'Register',
683                 width: 100
684             }
685         ]
686     });
687 });
688
689 Ext.onReady(function() {
690     Ext.create('Ext.form.Panel', {
691         title: 'PA Login Form (* fields are required)',
692         width: 300,
693         height: 200,
694         renderTo: '#loginForm',
695         items: [
696             {
697                 type: 'text',
698                 name: 'username',
699                 fieldLabel: 'Username:*',
700                 width: 200,
701                 height: 30
702             },
703             {
704                 type: 'password',
705                 name: 'password',
706                 fieldLabel: 'Password:*',
707                 width: 200,
708                 height: 30
709             }
710         ],
711         buttons: [
712             {
713                 text: 'Forgot username/password',
714                 width: 150
715             },
716             {
717                 text: 'Login',
718                 width: 100
719             },
720             {
721                 text: 'Register',
722                 width: 100
723             }
724         ]
725     });
726 });
727
728 Ext.onReady(function() {
729     Ext.create('Ext.form.Panel', {
730         title: 'PA Login Form (* fields are required)',
731         width: 300,
732         height: 200,
733         renderTo: '#loginForm',
734         items: [
735             {
736                 type: 'text',
737                 name: 'username',
738                 fieldLabel: 'Username:*',
739                 width: 200,
740                 height: 30
741             },
742             {
743                 type: 'password',
744                 name: 'password',
745                 fieldLabel: 'Password:*',
746                 width: 200,
747                 height: 30
748             }
749         ],
750         buttons: [
751             {
752                 text: 'Forgot username/password',
753                 width: 150
754             },
755             {
756                 text: 'Login',
757                 width: 100
758             },
759             {
760                 text: 'Register',
761                 width: 100
762             }
763         ]
764     });
765 });
766
767 Ext.onReady(function() {
768     Ext.create('Ext.form.Panel', {
769         title: 'PA Login Form (* fields are required)',
770         width: 300,
771         height: 200,
772         renderTo: '#loginForm',
773         items: [
774             {
775                 type: 'text',
776                 name: 'username',
777                 fieldLabel: 'Username:*',
778                 width: 200,
779                 height: 30
780             },
781             {
782                 type: 'password',
783                 name: 'password',
784                 fieldLabel: 'Password:*',
785                 width: 200,
786                 height: 30
787             }
788         ],
789         buttons: [
790             {
791                 text: 'Forgot username/password',
792                 width: 150
793             },
794             {
795                 text: 'Login',
796                 width: 100
797             },
798             {
799                 text: 'Register',
800                 width: 100
801             }
802         ]
803     });
804 });
805
806 Ext.onReady(function() {
807     Ext.create('Ext.form.Panel', {
808         title: 'PA Login Form (* fields are required)',
809         width: 300,
810         height: 200,
811         renderTo: '#loginForm',
812         items: [
813             {
814                 type: 'text',
815                 name: 'username',
816                 fieldLabel: 'Username:*',
817                 width: 200,
818                 height: 30
819             },
820             {
821                 type: 'password',
822                 name: 'password',
823                 fieldLabel: 'Password:*',
824                 width: 200,
825                 height: 30
826             }
827         ],
828         buttons: [
829             {
830                 text: 'Forgot username/password',
831                 width: 150
832             },
833             {
834                 text: 'Login',
835                 width: 100
836             },
837             {
838                 text: 'Register',
839                 width: 100
840             }
841         ]
842     });
843 });
844
845 Ext.onReady(function() {
846     Ext.create('Ext.form.Panel', {
847         title: 'PA Login Form (* fields are required)',
848         width: 300,
849         height: 200,
850         renderTo: '#loginForm',
851         items: [
852             {
853                 type: 'text',
854                 name: 'username',
855                 fieldLabel: 'Username:*',
856                 width: 200,
857                 height: 30
858             },
859             {
860                 type: 'password',
861                 name: 'password',
862                 fieldLabel: 'Password:*',
863                 width: 200,
864                 height: 30
865             }
866         ],
867         buttons: [
868             {
869                 text: 'Forgot username/password',
870                 width: 150
871             },
872             {
873                 text: 'Login',
874                 width: 100
875             },
876             {
877                 text: 'Register',
878                 width: 100
879             }
880         ]
881     });
882 });
883
884 Ext.onReady(function() {
885     Ext.create('Ext.form.Panel', {
886         title: 'PA Login Form (* fields are required)',
887         width: 300,
888         height: 200,
889         renderTo: '#loginForm',
890         items: [
891             {
892                 type: 'text',
893                 name: 'username',
894                 fieldLabel: 'Username:*',
895                 width: 200,
896                 height: 30
897             },
898             {
899                 type: 'password',
900                 name: 'password',
901                 fieldLabel: 'Password:*',
902                 width: 200,
903                 height: 30
904             }
905         ],
906         buttons: [
907             {
908                 text: 'Forgot username/password',
909                 width: 150
910             },
911             {
912                 text: 'Login',
913                 width: 100
914             },
915             {
916                 text: 'Register',
917                 width: 100
918             }
919         ]
920     });
921 });
922
923 Ext.onReady(function() {
924     Ext.create('Ext.form.Panel', {
925         title: 'PA Login Form (* fields are required)',
926         width: 300,
927         height: 200,
928         renderTo: '#loginForm',
929         items: [
930             {
931                 type: 'text',
932                 name: 'username',
933                 fieldLabel: 'Username:*',
934                 width: 200,
935                 height: 30
936             },
937             {
938                 type: 'password',
939                 name: 'password',
940                 fieldLabel: 'Password:*',
941                 width: 200,
942                 height: 30
943             }
944         ],
945         buttons: [
946             {
947                 text: 'Forgot username/password',
948                 width: 150
949             },
950             {
951                 text: 'Login',
952                 width: 100
953             },
954             {
955                 text: 'Register',
956                 width: 100
957             }
958         ]
959     });
960 });
961
962 Ext.onReady(function() {
963     Ext.create('Ext.form.Panel', {
964         title: 'PA Login Form (* fields are required)',
965         width: 300,
966         height: 200,
967         renderTo: '#loginForm',
968         items: [
969             {
970                 type: 'text',
971                 name: 'username',
972                 fieldLabel: 'Username:*',
973                 width: 200,
974                 height: 30
975             },
976             {
977                 type: 'password',
978                 name: 'password',
979                 fieldLabel: 'Password:*',
980                 width: 200,
981                 height: 30
982             }
983         ],
984         buttons: [
985             {
986                 text: 'Forgot username/password',
987                 width: 150
988             },
989             {
990                 text: 'Login',
991                 width: 100
992             },
993             {
994                 text: 'Register',
995                 width: 100
996             }
997         ]
998     });
999 });
999
```

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl commonStuff.js controlPanelUser.js toolbar.css balloons.js controlPanelSuper.js cookies.js loginManager.js /user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/loginManager.js 8084 bytes L: --- C: ---

```
218 var pwdField = Ext.create
219 (
220     'Ext.form.field.Text',
221 {
222     fieldLabel      : 'Password',
223     afterLabelTextTpl: required,
224     name           : 'password',
225     allowBlank     : false,
226     inputType      : 'password',
227     listeners       :
228         {
229             specialkey: function(field, e)
230                 {
231                     if (e.getKey() == e.ENTER)
232                     {
233                         loginFunction();
234                     }
235                 }
236         }
237     );
238
239 var usrField = Ext.create
240 (
241     'Ext.form.field.Text',
242     {
243         fieldLabel      : 'Username',
244         afterLabelTextTpl: required,
245         name           : 'userName',
246         allowBlank     : false,
247         listeners       :
248             {
249                 specialkey: function(field, e)
250                     {
251                         if (e.getKey() == e.ENTER)
252                         {
253                             pwdField.focus();
254                         }
255                     }
256             }
257     );
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
```

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl commonStuff.js controlPanelUser.js toolbar.css balloons.js controlPanelSuper.js cookies.js loginManager.js /user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/loginManager.js 8084 bytes L: --- C: ---

```
128     theURL += PA.get.cgiBName() + '/authorize.pl';
129     theURL += '?userName=' + userName;
130     theURL += '&password=' + password;
131     theURL += '&action=' + 'login' ;
132     Ext.Ajax.request
133     (
134         {
135             url      : theURL ,
136             success: function(response)
137             {
138                 if(response.userName)
139             },
140             failure: function(response)
141             {
142                 Ext.Msg.alert
143                 (
144                     'Failure',
145                     'Sorry, could not submit ' +
146                     'login request '
147                 );
148             }
149         }
150     );
151 };
152
153 var loginButton = Ext.create(
154     'Ext.button.Button',
155     {
156         text    : 'Login',
157         tooltip: 'Use this button to login as ' +
158                         'an already registered PA ' +
159                         'service user',
160         handler: loginFunction
161     }
162 );
163 
```



```
109 var loginFunction = function()
110 {
111     var theFields = simpleForm.getForm().getFields();
112     var theURL = '';
113     var userName = theFields.items[0].value;
114     var password = theFields.items[1].value;
115     if( !userName || !password)
116     {
117         PA.tools.Msg('WARNING',
118                     'Please provide both a '
119                     '<b><font color="red">valid</font></b> ' +
120                     '<b>username</b> <u><b>and</b></u> a ' +
121                     '<b>password</b>';
122                     'Thank you'
123                     'Please retry...'
124                     function(){}
125                     );
126     }
127     return ;
128 }
129 theURL += PA.get.cgiName() + '/authorize.pl';
130 theURL += '?userName=' + userName;
131 theURL += '&password=' + password;
132 theURL += '&action=' + 'login';
133 Ext.Ajax.request
134 (
135     {
136         url : theURL,
137         success: function(response)
138         {
139             if(response,userName)
140             },
141             failure: function(response)
142             {
143                 Ext.Msg.alert
144                 (
145                     'Failure'
146                 );
147             }
148         }
149     );
150 }
```

Supponiamo che lo script **authorize.pl**, in caso di **username** e **password** corretti, risponda con una stringa contenente **SUCCESS...**

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl commonStuff.js controlPanelUser.js toolbar.css balloons.js controlPanelSuper.js cookies.js loginManager.js /user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/loginManager.js 8084 bytes L: --- C: ---

```
75 var lf = function(response, userName)
76 {
77     var responseText = response.responseText ;
78     if( responseText.match(/^SUCCESS/))
79     {
80         PA.tools.Msg("Success",
81                     "Thank you, you have " +
82                     PA.get.expirationTime() +
83                     " minutes left before you'll have to login again",
84                     function(){}
85                 );
86         PA.Cookies.create('PAUserName',
87                         userName,
88                         PA.get.expirationTime()
89                     );
90         if( responseText.match(/^SUCCESS normal user/))
91         {
92             PA.controlPanel('User') ;
93         }
94         else if( responseText.match(/^SUCCESS super user/))
95         {
96             PA.controlPanel('Super') ;
97         }
98     }
99     else
100    {
101        PA.tools.Msg('Failure'
102                    'Reason: ' +
103                    responseText,
104                    function(){}
105                );
106    }
107}
108
109 var loginFunction = function()
110 {
```

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl balloons.js commonStuff.js controlPanelSuper.js controlPanelUser.js cookies.js loginManager.js

/user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/cookies.js 2588 bytes L: --- C: ---

```
1 /*****  
2 File : cookies.js  
3 Autors : D. Menasce  
4 Date : Jul 2014  
5  
6 Purpose: class to manage cookies client side  
7 *****/  
8 Ext.ns('PA.Cookies') ;  
9  
10 PA.Cookies =  
11 {  
12     init : function ()  
13     {  
14         var allCookies = document.cookie.split('; ');  
15         for (var i=0; i<allCookies.length; i++) // >  
16         {  
17             var cookiePair = allCookies[i].split('=');  
18             this[cookiePair[0]] = cookiePair[1];  
19         }  
20     },  
21     create: function (name,value,minutes)  
22     {  
23         var expires ;  
24         if (minutes)  
25         {  
26             var date = new Date();  
27             var now = date.toGMTString() ;  
28             date.setTime(date.getTime()+(minutes*60*1000));  
29             expires = "; expires=" + date.toGMTString();  
30         }  
31         else  
32         {  
33             var expires = "";  
34         }  
35         document.cookie = name + "=" + value + expires + "; path=/";  
36         this[name] = value;
```

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl balloons.js commonStuff.js controlPanelSuper.js controlPanelUser.js cookies.js loginManager.js

/user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/cookies.js byte 2587 of 2588 L: 88 C: 1

```
12  init  : function ()
13  {
14      var allCookies = document.cookie.split('; ');
15      for (var i=0; i<allCookies.length; i++) // >
16      {
17          var cookiePair = allCookies[i].split('=');
18          this[cookiePair[0]] = cookiePair[1];
19      }
20  },
```

Firebug - Form di login

Con... HTML CSS Script DOM Net Cookies Illu... GFX Dojo Cha... ▶ 🔍 Se: ⌂ ⌄ ⌄ ⌄

Cookies ▾ Filter ▾ Default (Accept cookies) ▾

Name	Value	Domain	Raw Size	Expires
▶ 67414c66cd1fcdeccf79a2b067254918	it-IT	www.infn.it	37 B	25/01/2015 18:42:27
▶ 6c435bf558168bc57b23959d35fc4063	9ad1favib1jll2gcppmeb1oir0	www.infn.it	58 B	Session
▶ 82d0b5be23ddff5ae40d81eb5c2099a3	it-IT	www.infn.it	37 B	16/10/2015 11:38:36
▶ PAUserName	guest	mobile.mib.infn.it	15 B	15/10/2014 14:11:49
▶ _utma	201308338.470728258.1379...13463031.1413465626.157	.mib.infn.it	62 B	15/10/2016 15:20:26
▶ _utma	211720397.872847529.1384...1384881328.1384881328.1	.infn.it	60 B	19/11/2015 18:16:11
▶ _utmb	201308338.1.10.1413465626	.mib.infn.it	31 B	16/10/2014 15:50:26
▶ _utmc	201308338	.mib.infn.it	15 B	Session
▶ _utmt	1	.mib.infn.it	7 B	16/10/2014 15:30:26
▶ _utmz	201308338.1410218454.146...user/guest_redirect.jsp	.mib.infn.it	118 B	17/04/2015 03:20:26
▶ wb_session_id	j9no6l6qlr2q2f92rch3g8jnu2	www.mib.infn.it	39 B	Session

File Edit Search Preferences Shell Macro Windows Help

controlPanelUser.html authorize.pl balloons.js commonStuff.js controlPanelSuper.js controlPanelUser.js cookies.js loginManager.js

/user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/cookies.js 2588 bytes L: --- C: ---

Firebug - Form di login

Con... HTML CSS Script DOM Net Coo... Illu... GFX Dojo Cha... Watch Stack Breakpoints

all cookies.js {} <.Cookies.create > If > loginFunc....s > Watch Stack Breakpoints

24 if (minutes)
25 {
26 var date = new Date();
27 var now = date.toGMTString();
28 date.setTime(date.getTime()+(minutes*60*1000));
29 expires = "; expires=" + date.toGMTString();
30 }
31 else
32 {
33 var expires = "";
34 }
35 document.cookie = name + "=" + value + expires + "; path=/";
36 this[name] = value;
37 },
38 erase : function (name)
39 {

function (name, value, minutes)
name PAUserName
value "guest"
minutes "1"
arguments ["PAUserName", "guest", "1"]
expires "; expires=Thu, 16 Oct 2014 16:03:34 GMT"
date Date { Thu Oct 16 2014 18:03:34 GMT+0200 (C)
now Thu, 16 Oct 2014 16:02:34 GMT
Global Scope [Window] Window PA.html

GET http://hal9000.mib.infn.it/cgi-bin/dottorato/eser...st&password=dario&action=login&_dc=1413475354649 200 OK 25ms ext-all-debug.js (line 32386)
! Relative positioning of table rows and row groups is now supported. This site may need to be updated because it may depend on this feature having no effect ext-all-debug.js (line 13460)

23
24 var expires :
25 if (minutes)
26 {
27 var date = new Date();
28 var now = date.toGMTString();
29 date.setTime(date.getTime()+(minutes*60*1000));
30 expires = "; expires=" + date.toGMTString();
31 }
32 else
33 {
34 var expires = "";
35 }
36 document.cookie = name + "=" + value + expires + "; path=/";
37 this[name] = value;

Ma quando e dove si verifica se il cookie è ancora valido o è spirato?

controlPanelUser.js - /user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl balloons.js commonStuff.js controlPanelSuper.js controlPanelUser.js cookies.js loginManager.js

/user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/controlPanelUser.js byte 1161 of 16218 L: 55 C: 33

```
1 Ext.require
2 (
3 [
4 'Ext.window.Window',
5 'Ext.toolbar.Toolbar',
6 'Ext.form.field.Date'
7 ]
8 );
9
10 Ext.onReady
11 (
12 function()
13 {
14 if( !(PA.Cookies['PAUserName']) )
15 {
16 window.location = PA.get.home() ;
17 }
18
19 PA.sessionTimer = setTimeout(PA.checkSession,PA.checkPeriod) ;
20
21 var toolBar= Ext.create('Ext.toolbar.Toolbar');
22 toolBar.suspendLayouts();
23
24 Ext.define(
25     'modelSezioni',
26     {
27         extend: 'Ext.data.Model',
28         fields: [
29             {type: 'string', name: 'abbr' },
30             {type: 'string', name: 'name' },
31             {type: 'string', name: 'where'}
32         ]
33     });
34
35 Ext.define(
36     'modelAuto',
```

Appena caricato il documento nel DOM in memoria, viene eseguita la funzione **checkSession**

L: 55 C: 33

File Edit Search Preferences Shell Macro Windows Help

controlPanelUser.html authorize.pl balloons.js commonStuff.js controlPanelSuper.js controlPanelUser.js cookies.js loginManager.js

L: --- C: ---

```
PA.html /user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/cookies.js 2584 bytes
```

```
43     check : function(name)
44     {
45         var theSessionCookie = [];
46         var allCookies = document.cookie.split('; ');
47         for (var i=0; i<allCookies.length; i++) // >
48         {
49             var cookiePair = allCookies[i].split('=');
50             theSessionCookie[cookiePair[0]] = cookiePair[1];
51         }
52         return theSessionCookie[name] ;
53     }
54 };
55
56 PA.Cookies.init();
57
58 PA.checkPeriod = 5000, // Check every 5 seconds (msecs)
59 PA.checkSession = function(newPage)
60 {
61     console.log("Checking for cookie expiration ");
62     if( PA.Cookies.check('PAUserName') )
63     {
64         var date = new Date();
65         now = date.toGMTString();
66         PA.sessionTimer = setTimeout(PA.checkSession,PA.checkPeriod)
67     } else {
68         console.log("Timeout expired");
69         Ext.Msg.show(
70             {
71                 title : 'Session expired',
72                 msg : 'Your current session is expired, '
73                             ' you need to login again to continue'
74                 buttons: Ext.MessageBox.OK,
75                 icon : Ext.MessageBox.INFO,
76                 fn : function(btn)
77                 {
78                     PA.Cookies.erase('PAUserName');
79                 }
80             }
81         );
82     }
83 }
84
85 Ext.onReady(function()
86 {
87     PA.checkSession();
88 })
```

File Edit Search Preferences Shell Macro Windows Help

controlPanelUser.html authorize.pl balloons.js commonStuff.js controlPanelSuper.js controlPanelUser.js cookies.js loginManager.js

/user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/cookies.js byte 1791 of 2584 L: 68 C: 47

```
54 };
55
56 PA.Cookies.init();
57
58 PA.checkPeriod = 5000 ; // Check every 5 seconds (msecs)
59 PA.checkSession = function(newPage)
60 {
61     console.log("Checking for cookie expiration ");
62     if( PA.Cookies.check('PAUserName') )
63     {
64         var date = new Date();
65         now = date.toGMTString();
66         PA.sessionTimer = setTimeout(PA.checkSession,PA.checkPeriod)
67     } else {
68         console.log("Timeout expired");
69         Ext.Msg.show(
70             {
71                 title : 'Session expired',
72                 msg : 'Your current session is expired, '
73                             ' you need to login again to continue'
74                 buttons: Ext.MessageBox.OK,
75                 icon : Ext.MessageBox.INFO,
76                 fn : function(btn)
77                 {
78                     PA.Cookies.erase('PAUserName');
79                     clearTimeout(PA.sessionTimer);
80                     var hrefLoc = PA.get.htmlName() +
81                             '/PA.html';
82                     window.location = hrefLoc;
83                 }
84             }
85         );
86     }
87 }
88
89 }
```

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl commonStuff.js controlPanelUser.js toolbar.css balloons.js controlPanelSuper.js cookies.js loginManager.js

/user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/balloons.js byte 1499 of 2267 L: 60 C: 34

```
1 /*****  
2 File : loginManager.js  
3 Autors : D. Menasce  
4 Date : Jul 2014  
5 *****/  
6  
7 Ext.ns('PA.tools') ;  
8  
9 PA.tools.balloons =  
10 function()  
11 {  
12     var msgCt;  
13  
14     function createBox(t, s)  
15     {  
16         return '<div class="msg"><h3><font color="#1c61ff">' +  
17             t +  
18             '</font></h3><hr size=2/><p>' +  
19             s +  
20             '</p></div>';  
21     }  
22     return {  
23         msg : function(title, format)  
24         {  
25             var timeOut = PA.tools.timeOut ;  
26             if( timeOut == undefined || timeOut == '') {timeOut = 6000;}  
27             if(!msgCt)  
28             {  
29                 msgCt = Ext.DomHelper.insertFirst(  
30                     'balloonPlaceholder',  
31                     {id:'msg-div'},  
32                     true  
33                     );  
34             }  
35             var s = Ext.String.format.apply(  
36                                         String,
```

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl commonStuff.js controlPanelUser.js toolbar.css balloons.js controlPanelSuper.js cookies.js loginManager.js

/user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/balloons.js 2267 bytes L: --- C: ---

```
22 return {
23     msg : function(title, format)
24     {
25         var timeOut = PA.tools.timeOut ;
26         if( timeOut == undefined || timeOut == '' ) {timeOut = 6000;}
27         if(!msgCt)
28         {
29             msgCt = Ext.DomHelper.insertFirst(
30                             'balloonPlaceholder',
31                             {id:'msg-div'},
32                             true
33                         );
34         }
35         var s = Ext.String.format.apply(
36             String,
37             Array.prototype.slice.call
38             (
39                 arguments,
40                 1
41             )
42         );
43         var m = Ext.DomHelper.append(
44             msgCt
45             createBox(title, s),
46             true
47         );
48         m.hide();
49         m.slideIn('t').ghost("t", { delay: timeOut, remove: true});
50     },
51
52     init: function()
53     {
54     }
55 };
56 }();
```

File Edit Search Preferences Shell Macro Windows Help

PA.html controlPanelUser.html authorize.pl commonStuff.js controlPanelUser.js toolbar.css balloons.js controlPanelSuper.js cookies.js loginManager.js

/user/gr1/e831/menasce/public_html/esercizi/PA/jsFiles/commonStuff.js byte 1310 of 1310 L: 36 C: 0

```
1 /*****  
2 File : commonStuff.js  
3 Autors : D. Menasce  
4 Date : Jul 2014  
5  
6 *****/  
7  
8 Ext.ns('PA.get') ;  
9  
10 PA.get = (  
11     function()  
12     {  
13         var nodeName = 'http://mobile.mib.infn.it' ;  
14         var userName = 'menasce' ;  
15         var subDir = '/esercizi/PA' ;  
16         var PADir = '/user/gr1/e831/menasce/public_html' +  
17             subDir ;  
18         var CGIDir = '/cgi-bin/dottorato/esercizi/PA' +  
19             '/perlFiles' ;  
20         var expirationTime = '5' ; // Minutes  
21         return {  
22             home : function() { return nodeName ; },  
23             paDir : function() { return PADir ; },  
24             cgibName : function() { return nodeName + CGIDir ; },  
25             expirationTime: function() { return expirationTime ; },  
26             htmlName : function() {  
27                 return nodeName +  
28                     '~/' +  
29                     userName +  
30                     subDir +  
31                     '/htmlFiles' ; },  
32         } ;  
33     }  
34 )();
```



An example of an HTML5 canvas and related capabilities

An example originally created by David Geary. This code has been modified by D. Menasce from the book Core HTML5 Canvas, published by Prentice-Hall in 2012.

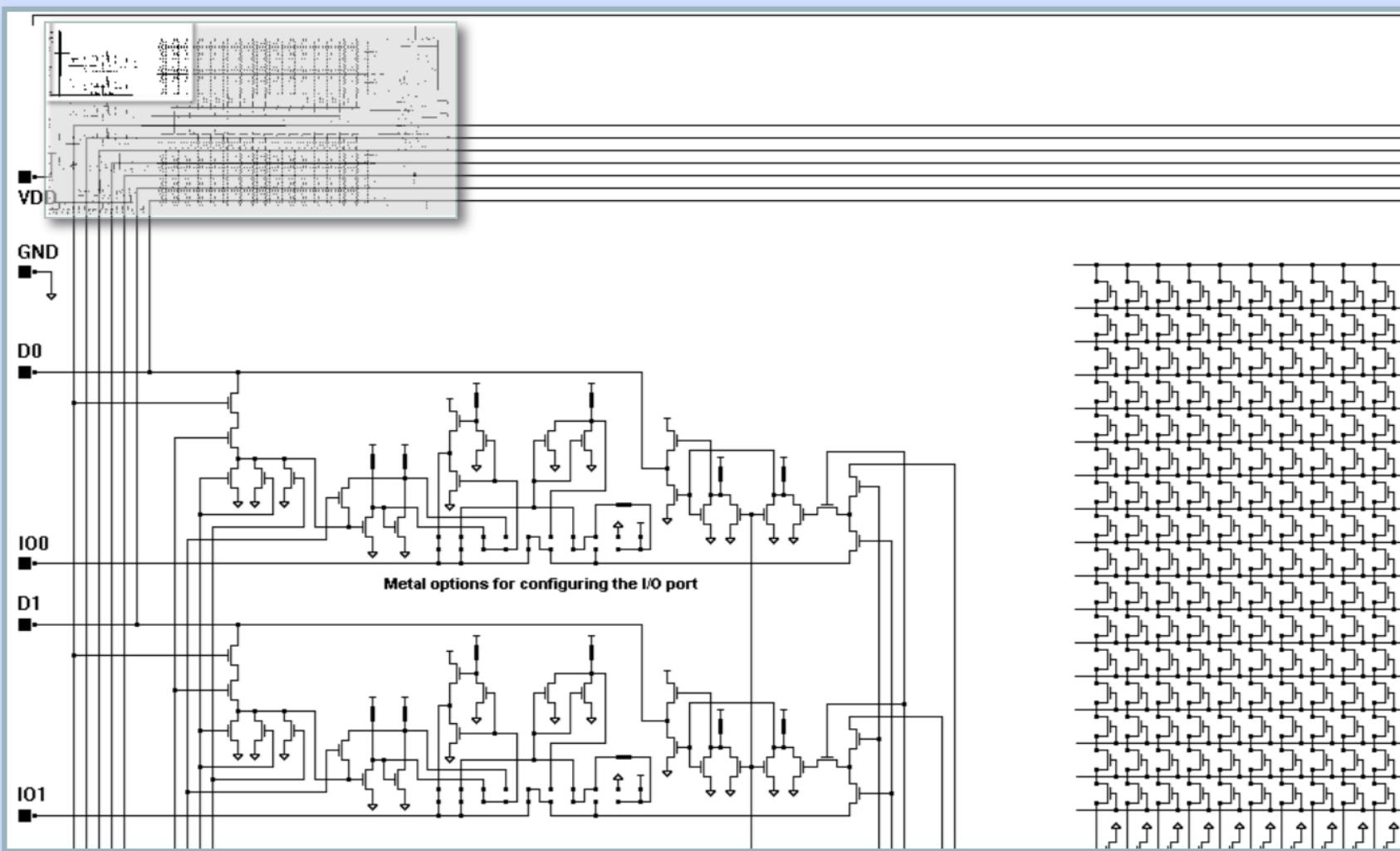
Lens
size

10%



Lens
transparency

50%



Change Pic

An example of an HTML5 canvas and related capabilities

An example originally created by David Geary. This code has been modified by D. Menasce from the book Core HTML5 Canvas, published by Prentice-Hall in 2012.

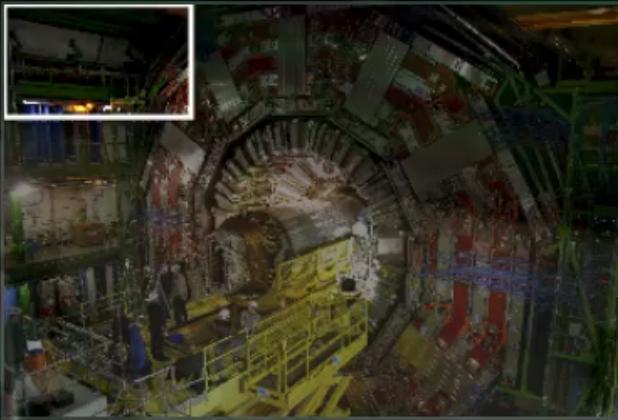
Lens
size

10%

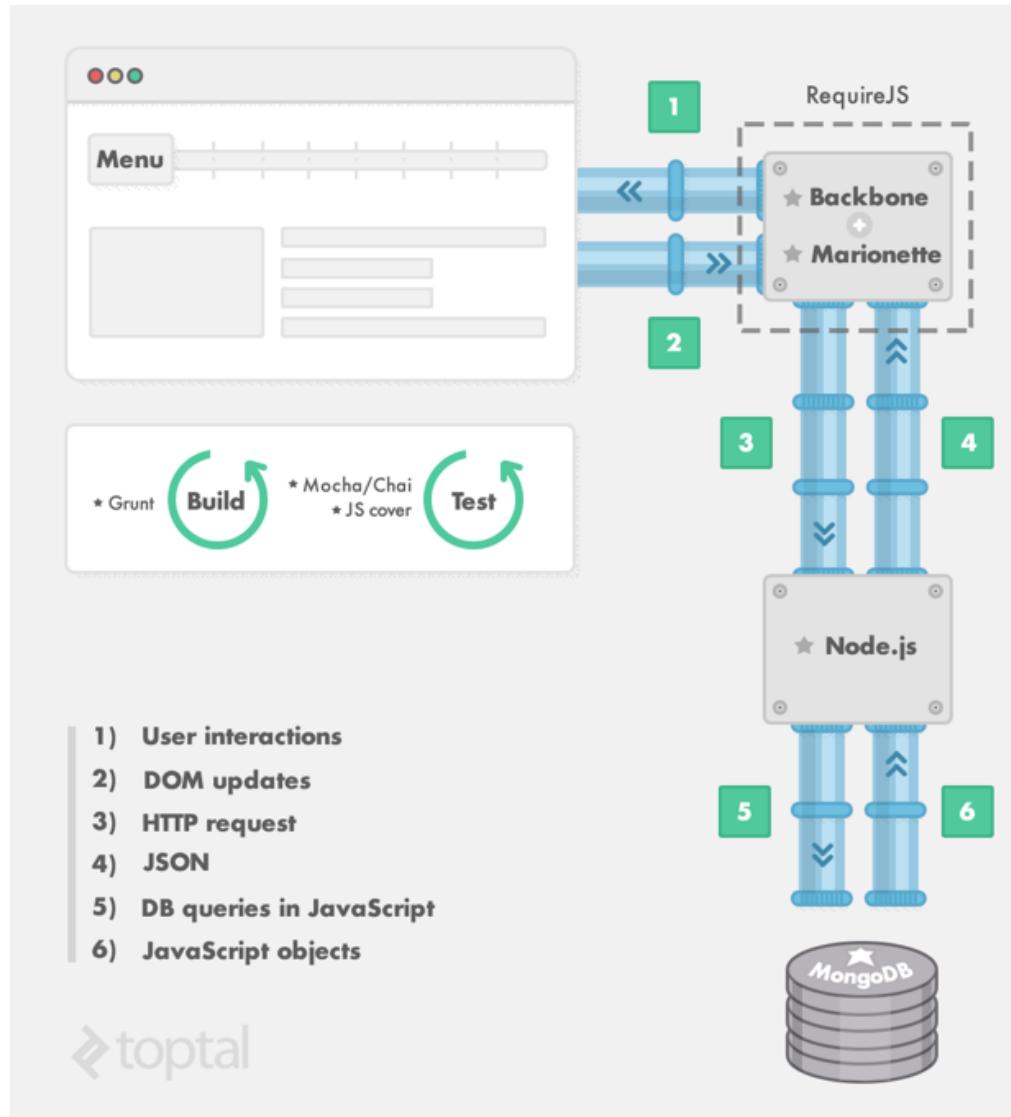


Lens
transparency

50%



```
33 <html>
34
35 <head>
36   <meta charset="UTF-8" />
37   <title>Panning Images</title>
38   <link rel="stylesheet" type="text/css" href="../cssFiles/pan.css"
39 </head>
40
41 <body id='body'>
42   <div id='intro'>
43     An example of an HTML5 canvas and related capabilities
44   </div>
45   <div id='credits'>
46     An example originally created by David Geary. This code has been modified
47     by D. Menasce from the book <br> Core HTML5 Canvas,
48     published by Prentice-Hall in 2012.
49 </div>
50
51   <div id='controls'      >
52     <span id='sizeText'    >Lens size          </span>
53     <span id='sizeSpan'   >0                  </span>
54     <div id='alphaSliderDiv'>                    </div>
55
56     <span id='alphaText'   >Lens transparency</span>
57     <span id='alphaSpan'  >0                  </span>
58     <div id='sizeSliderDiv' >                    </div>
59   </div>
60   <canvas id='canvas' width='1000' height='600'>
61     Canvas is not supported by this browser!...
62   </canvas>
63   <script src='../jsFiles/panRoundedRectangle.js'></script>
64   <script src='../jsFiles/panSlider.js'           ></script>
65   <script src='../jsFiles/pan.js'                ></script>
66   <script src='../jsFiles/panExample.js'         ></script>
67 </body>
68 </html>
```



Consideriamo il caso di un server che deve fornire in output il nome di una persona in seguito alla *query* ad un DB, e vediamo una tipica applicazione implementata in modo tradizionale, ossia sequenziale:

Il processo rimane bloccato fintanto che l'operazione di query non ha avuto luogo (con tempi di latenza possibilmente lunghi).

Durante tutto quel tempo il server non può fare altro e qualsiasi altra richiesta rimarrà bloccata, forzando ad usare un server in un thread parallelo.

1
2
3
4
5
6

```
console.log('Name: ' + getUser(432).name);
```

L'utilizzo di funzioni di tipo call-back, invece, amplia notevolmente l'efficienza di esecuzione di un programma, rimuovendo i blocchi.

La funzione callback in riga 2 verrà chiamata asincronamente solo al momento in cui la query avrà avuto successo. Durante tutto quel tempo il server sarà libero di rispondere ad altre richieste.

```
1  function getUser(id, callback) {  
2      db.query(id, callback);  
3  }  
4  
5  function display(user) {  
6      console.log(user.name);  
7  }  
8  
9  getUser(432, display);
```

La sintassi di Javascript per le funzioni anonime permette poi di riscrivere il codice in maniera più compatta:

```
1 function getUser(id, callback) {  
2     db.query(id, callback);  
3 }  
4  
5 getUser(432, function (user) {  
6     console.log(user.name);  
7 });
```

Occorre però fare molta attenzione alla sequenza:

Done può infatti apparire nello STDOUT prima del nome dello user!!!

```
1 getUser(432, function (user) {  
2     console.log(user.name);  
3 });  
4  
5 console.log('Done');
```


Fine quarta parte