Contents

About this widget

This is a widget for testing filters with the HTTP service.

Xpath used in this example. Regex filters are also available.

WidSets Scripting Language code: filter_test.he

```
class
{
  const int     CMD_BACK   = 1;
  const int     CMD_CANCEL = 2;

  MenuItem      BACK   = new MenuItem(CMD_BACK, "Back");
  MenuItem      CANCEL = new MenuItem(CMD_CANCEL, "Cancel");

  String        URL = "http://www.deftit.com/data.xml";

  Flow          content;
  Prompt        prompt;

  Component createElement(String view, String id, Style style, Object ctx)
  {
    if (id.equals("max")) {
      content = new Flow(style);
      content.setPreferredSize(-100, -100);
      content.add(createLabel("Loading"));
      return content;

    } else {
      return null;
    }
  }


  void startWidget()
  {
    setMinimizedView(createMinimizedView("viewMini", getStyle("white")));
```

```
}


Shell openWidget()
{
  Flow view = createMaximizedView("viewMaxi", getStyle("maxi"));
  Shell shell = new Shell(view);

  fetchData();
  return shell;
}


Label createLabel(String text)
{
  Label label = new Label(getStyle("label"), text);
  label.setPreferredWidth(-100);
  return label;
}


void fetchData()
{
  printf("Fetching data");
  call(null, "httpService" , "get", ["url" => URL], success, failure);
  prompt = new Prompt(null, "Downloading...", null, CANCEL);
  prompt.push();


  void success(Object state, Value ret)
  {
    printf("ret: "+ret);
    closePrompt();

    content.clear();
    foreach (Value item : ret) {
      content.add(createLabel(String(item[1])));
    }
  }

  void failure(Object state, String error)
  {
    printf("error: "+error);
    closePrompt();
  }

  void closePrompt()
  {
    if (prompt != null) {
      prompt.pop();
      prompt = null;
    }
  }
}


MenuItem getSoftKey(Shell shell,
                    Component focused,
                    int key)
{
  if (key == SOFTKEY_OK) {
    return null;
```

```
    } else if (key == SOFTKEY_BACK) {
      return BACK;
    }
    return null;
  }


  void actionPerformed(Shell shell, Component source, int action)
  {
    switch(action)
    {
      case CMD_BACK:
      {
        content = null;
        popShell(shell);
      }
      break;

      case CMD_CANCEL:
      {
        if (prompt != null) {
          prompt.pop();
          prompt = null;
        }
      }
      break;
    }
  }

}
```

Widget.xml

```
<?xml version="1.0" encoding="utf-8"?>

<widget spec_version="2.0">

  <info>
    <name>filter_test</name>
    <version>0.1</version>
    <author>render</author>
    <clientversion>0.98</clientversion>
    <shortdescription>Example widget demonstrating use of xpath filters</shortdescription>
    <longdescription>Example widget demonstrating use of xpath filters to fetch XML data from Web
    <tags>test example xpath filter</tags>
  </info>

  <services>
    <service type="http" id="httpService">
     <filter id="xpathfilter"/>
    </service>
  </services>

  <filters>
    <filter id="xpathfilter">
      <list>
        <foreach>
          <xpath>/record/*</xpath>
```

```
        <item>
          <name><xpath>name()</xpath></name>
          <value><xpath>text()</xpath></value>
        </item>
      </foreach>
    </list>
  </filter>
</filters>

<parameters>
  <parameter name="widgetname">filter_test</parameter>
</parameters>

<resources>
  <code src="filter_test.he"/>

  <stylesheet>
    white {
      background: solid white;
    }

    minbg {
      background: solid white;
    }

    titleLabel {
      padding: 0 0 10 0;
      font-1: small bold;
      color-1: #000000;
      align: hcenter vcenter;
    }

    maxi {
      color-1: black;
      background: solid white;
      align: hcenter vcenter;
      border: 1 1 1 1;
      border-type: rectangle black;
    }

    label {
      font-1: large;
      color-1: black;
      align: hcenter vcenter;
    }
  </stylesheet>
</resources>

<layout minimizedheight="65sp">
  <view id="viewMini" class="minbg">
    <label class="titleLabel">${widgetname}</label>
  </view>

  <view id="viewMaxi" class="maxi">
    <script id="max" class="maxi"/>
  </view>

  <webview>
    <weblabel class="top:0px;left:10px;">${widgetname}</weblabel>
  </webview>
</layout>
```

```
</widget>
```

Note

Due to *feature* in Xpath-filter implementation,

```
name()
```

function returns extra brackets, and so to use it, you need to do some substringing.

See also

- WidSets SDK
- WidSets Client
- WidSets Scripting Language
- Widget examples
    - WidClock
    - Memory Game
    - **Filter test**
    - Hello World
    - UITest