

Example_Http_Service

This article is archived because it is not considered relevant for third-party developers creating commercial solutions today. The article is believed to be still valid for the original topic scope.



This script code does a HTTP GET search to Google and prints the resulting HTML content to a log.

http_service.he

```
class
{
    const String URL = "http://www.google.com/search";

    void startWidget()
    {
        setMinimizedView(createMinimizedView("viewMini", getStyle("default")));
    }

    Shell openWidget()
    {
        fetch();

        return null;
    }

    void fetch()
    {
        //http parameters to be used in get-query
        Value params = [
            "q" => "widsets",
            "meta" => ""
        ];

        //widsets http service parameters
        Value arg = [
            "url" => URL,
            "params" => params
        ];
        call(null, "httpService", "get", arg, ok, nok);
    }

    void ok(Object state, Value ret)
    {
        setBubble(null, "Got "+ret.size()+" bytes of html");
        //printing html to log
        printf(ret);
    }

    void nok(Object state, String error)
    {
        setBubble(null, "Http get failed: "+error);
    }
}
```

Example_Http_Service

```
} //class
```

widget.xml

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

<widget spec_version="2.0">
  <info>
    <name>example_http_service</name>
    <version>1.0</version>
    <author>example</author>
    <clientversion>1.0</clientversion>
    <shortdescription>Http Service Example</shortdescription>
    <longdescription>Http Service Example</longdescription>
    <tags>example httpservice</tags>
  </info>

  <parameters>
    <parameter name="widgetname">Http Service</parameter>
  </parameters>

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

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

    <stylesheet>
      mini {
        background: solid white;
        color: black;
        align: vcenter hcenter;
      }

      maxi {
        background: solid white;
        padding: 5 5 5 5;
      }

      flow {
        background: solid white;
      }
    </stylesheet>
  </resources>

  <layout minimizedheight="2em">
    <view id="viewMini" class="mini">
      <label class="mini">${widgetname}</label>
    </view>
  </layout>
</widget>
```

Note. You can search for certain parts from html-pages by defining Regular Expression -filters to your http-service.