

RTReport

Testprojekt

JART Playground aufm HAN

Hands On

Tabelle anlegen

Testdaten eingefügt

DB APP Bulder

Report spezifischer pfad ist immer nach `"/prj3/.../appdes/<report-spezifischer-pfad>"`

besitzt ein feld namens "disable on key", um eine condition zu disablen!!

Der RT Report

Alles unter dbadmin_reports --> reports Im Feld App Builders können beistrichgetrennt die "report-spezifischer-pfad"e eingefügt werden

Zuerst muss man den Report mit File>Save speichern, damit man alle applikationen die zur verfügung stehen angezeigt werden. Speicherpfad muss folder indexes/<index-name>/<report-folder>/ sein, also zb indexes/main/...

Funktionen

Sind definiert in prj3/jart-tools/js-apps/rt-report/run/web.cjs, welches zb auch global-handlebars.js importiert.
<http://han.jart.at/jart/prj3/jart-tools/resources/developer-apps/text-editor/c-editor.jart?fn=/prj3/jart-tools/js-apps/>

Handlebars:

- `_date`: formatiert das datum; zb `{{_date termindatum 'EEE dd.MMM.yyyy'}}`
- `_link`:
- `_media_img`: `{{_media_img termin_bid '&v=Box&width=300'}}`
- `_rt_value` : gibt HTML inhalt aus und macht security überprüfungen

Bilder

Feld vom Typ "Upload", subtyp "mediadb"

Detailseite

- Bei der übersichtsseite stellt man "disable on key" zb auf "termin_id", und vergibt in der liste einen

- link um auf die detailseite zu kommen
- 2. Report für die detailseite machen welche einen DBApp Builder hat, der die details ausgibt. report natürlich noch auf die selbe seite befüllen.

TAB: Code

Der richtext editor vermurkst teilweise den html/handlebars code. Falls das passiert muss man im Code weiterarbeiten und den editor disable. einfach als oberste zeile: Generate: "Disable Richtext"

man kann aber auch nur teile dem RT-Editor "verstecken" und zwar mit einem html Comment tag mit 3 Bindestrichen

TAB: CSS

Hier kann man in wirklichkeit LESS verwenden!

TAB: Javascriptp (Server Side)

Zugriff auf jart dateien (jartrequest) bzw jart funktionen (jartcall). Außerdem kann man auf so ziemlich alle funktionen zugreifen, die auch bei handlebars mit underscore '_' aufgerufen werden; zb "lang()"

Debugging

\$.log(object), oder \$.debug(object); beide funtkionen sind die selben logobject(object); logt alle properties
formatJsonToHtml(object); loggt object-graph

STRG+M zeigt ein popup was beim report generieren eigentlich passiert.

Navigation Generate

- onData: wenn daten da sind, soll NUR EINMAL verwendet werden. darin dann die jeweiligen funktionen programmieren

Tabs

Structure

Zeigt die struktur der DBApp Builder ergebnisse an. Klickt man den Bleistift, fügt sich das gewünschte feld automatisch im report ein (Handlebar Syntax)

Ausgabe filtern

Ausgabe soll zum beispiel nach datum oder "kategorie" oder whatever gefiltert werden. Dazu einfach im DBApp builder eine condition einfügen. Bei der befüllten Applikation auf der Seite in den "Einstellungen" kann man nun die conditionen mit werten befüllen.

Wie kommt der Report auf die Webseite?

Im editiermode "Applikation" einfügen. Hier wählt man den Report aus u speichern. Tadaaaa, er ist da

Debugging

Wenn bei der Seite nix oder das falsche angezeigt wird kann man folgendes überprüfen

- query parameter debug-sql=y an seite anhängen: zeigt welches SQL ausgeführt wurde

Anhang

```
//@import /prj3/jart-tools/js-apps/db-admin/resources/jquery/handlebars-v1.3.0.js
//@import xml-to-json.js
//@import global-handlebars.js

function getJartRequest(url){
    var ret = null;
    ext.strFnc($.filter, "getJSJartRequest", url);
    var node = $.filter.selectSingleNode("/data/fnc-data/getJSJartRequest", $.outNode);
    if(node){
        ret = {};
        convertXToJ(node, ret);
    }
    return ret;
}

function subReport(call, opts){
    var ret = null;
    if(opts){
        for(var n in opts){
            $.filter.params.put(n, opts[n]);
        }
    }
    return "" + ext.strFnc($.filter, "getSubReport", call);
}

function getJartCall(call, opts){
    var ret = null;
    if(opts){
        for(var n in opts){
            $.filter.params.put(n, opts[n]);
        }
    }
    ext.strFnc($.filter, "getJSJartCall", call);
    var node = $.filter.selectSingleNode("/data/fnc-data/getJSJartCall", $.outNode);
    if(node){
        ret = {};
        convertXToJ(node, ret);
    }
    return ret;
}

function logObject(obj){
    if(obj){
        $.log(JSON.stringify(obj).replace(/"/g, "'", ''));
    } else {
        $.log("null");
    }
}

function cleanObject(obj){
    if(typeof obj != "object") return null;
```

```

var ret = {};
for(var i in obj){
    if(typeof obj[i] == "object"){
        if(Array.isArray(obj[i])){
            if(obj[i].length === 1 && obj[i][0]._text !== null){
                ret[i] = obj[i][0]._text;
            }
        } else {
            ret[i] = [];
            for(var i2 = 0; i2 < obj[i].length; i2++){
                ret[i].push(cleanObject(obj[i][i2]));
            }
        }
    } else {
        ret[i] = obj[i];
    }
}
return ret;
}

function formatJsonToHtml(obj, isSub){
    var ret = "";
    if(typeof obj == "object"){
        ret += "<ul" + (isSub ? ' style="display: none;" ' : '') + ">";
        for(var i in obj){
            ret += "<li style='cursor: pointer;' onclick='$(\">ul\", this).toggle(); event.st";
            ret += i + ": ";
            ret += formatJsonToHtml(obj[i], true);
            ret += "</li>";
        }
        ret += "</ul>";
    } else {
        ret = "" + obj;
    }
    return ret;
}

function getMediaInfo(img, vc){
    var mUrl = "/pub-prc/img.jart?base=/prj3/" + data["j-project"] + "&img=/images/img-db/"
    return getJartRequest(mUrl);
}

function each(a, cb){
    if(!a) return;
    for(var i = 0; i < a.length; i++){
        cb(i, a[i]);
    }
}

Handlebars.registerHelper('_subReport', function(conf, params) {
    if(!conf) return "";
    if(params && typeof params == "string"){
        var p = params.replace(/, /g, ",").split(",");
        var me = this;
        each(p, function(i, v){
            var val = me[v];
            if(val) $.filter.params.put(v, val);
        });
    }
    return new Handlebars.SafeString(subReport(conf));
});

Handlebars.registerHelper('_json2ul', function(ob) {
    if(!ob) return "";
    return new Handlebars.SafeString(formatJsonToHtml(ob));
});

```

```

});

Handlebars.registerHelper('_rt_value', function(val) {
    if(!val) return "";
    return new Handlebars.SafeString(val.replace(/<[/]{0,1}body>/gi, ""));
});

Handlebars.registerHelper('_media_img_src', function(img, vc) {
    var ret = "";
    var imgInfo = getMediaInfo(img, vc);
    if(imgInfo && imgInfo["image-info"] && imgInfo["image-info"].length > 0){
        return imgInfo["image-info"][0].src;
    }
    return new Handlebars.SafeString(ret);
});

Handlebars.registerHelper('_media_img', function(img, vc, _alt) {
    var ret = "";
    if(!img) return ret;
    var alt = _alt;
    var imgInfo = getMediaInfo(img, vc);
    if(imgInfo && imgInfo["image-info"] && imgInfo["image-info"].length > 0){
        var src = imgInfo["image-info"][0].src
        if(!alt){
            if(imgInfo["image-info"][0].title && imgInfo["image-info"][0].title[0]._text){
                alt = imgInfo["image-info"][0].title[0]._text;
            }
            if(imgInfo["image-info"][0].alt && imgInfo["image-info"][0].alt[0]._text){
                alt = (alt ? " " : "") + imgInfo["image-info"][0].alt[0]._text;
            }
        }
        ret = '';
    }
    return new Handlebars.SafeString(ret);
});

function _lang(str) {
    var ret = '';
    var p = str.split("??");
    var found = false;
    for(var i = 0; i < p.length && !found; i++){
        if(i===0 || data['j-db-lang'] && p[i].indexOf(data['j-db-lang'] + "=") === 0){
            ret = p[i].substr(p[i].indexOf("=") + 1);
        }
    }
    return ret;
}

Handlebars.registerHelper('_lang', function(str) {
    return _lang(str);
});

Handlebars.registerHelper('_log', function(str) {
    $.log(str);
    return "";
});

function link(ext, cid) {
    var lnk = "";
    if(cid){
        lnk = data["j-rlink"] + "&content-id=" + cid + ext;
    } else {
        lnk = data["j-slink"] + ext;
    }

    return "" + Packages.jart.Extension.strFnc($.filter, "getNiceUrlV4", lnk);
}

```

```

}

Handlebars.registerHelper('_link', function(_ext, _cid) {
  var ext = "";
  if(_ext){
    var p = _ext.split(",");
    for(var i = 0; i < p.length; i++){
      ext += "&" + p[i] + "=" + this[p[i]];
    }
  }
  var cid = typeof _cid === "string" ? _cid : null;
  return link(ext, cid);
});

Handlebars.registerHelper('_j_r_link', function(img, vc) {
  var ret = "";
  ret += data["j-rlink"];
  return new Handlebars.SafeString(ret);
});

Handlebars.registerHelper('_j_s_link', function(img, vc) {
  var ret = "";
  ret += data["j-slink"];
  return new Handlebars.SafeString(ret);
});

//@import runner.js

```