

Beanshell vs Javascript

Description

Soffid 4 configures the smart engine with **Javascript** scripting language as the default. See [Smart engine settings](#).

Previously, the default engine was Beanshell, and many scripts will need to be adapted.

This page lists these differences.

Related objects

- [Smart engine settings](#) : where the engine is configured.
- Where we can use scripts:
 - [Agents](#) : properties, mappings and triggers
 - [Custom scripts](#) : all the scripts
 - [Account naming rules](#) : script to validate and set name
 - [Role assignment rules](#) : script to validate
 - [BPM editor](#) : visualization, triggers, transitions
 - [Password policies](#) : optional script

Table of differences

Topic	Beanshell	Javascript
-------	-----------	------------

variable	s = "text";	// The use of var should be mandatory, //but it almost always works without using it. var s = "text"; or s = "text";
function	public void doSomething(String system) { ... } doSomething("APP_USERS");	function doSomething(system) { ... } doSomething("APP_USERS");
for	for (user : listOfUsers) { ... }	for (var i=0; i<listOgUsers.size(); i++) { ... }
equals	user == null	user === null
equals	userName.equals("myName")	userName === "myName"
java class	es.caib.seycon.ng.comu.AccountType	Java.type("es.caib.seycon.ng.comu.AccountType") // We can also generate objects without Java.type query = new com.soffid.zkdb.api.Query(); or Query = Java.type("com.soffid.zkdb.api.Query"); query = new Query();
catch / printStackTrace	} catch(Exception e) { e.printStackTrace(out); }	} catch(e) { out.println(e.message); }
	e.printStackTrace(out)	out.println(e.message) + out.println(e.stack)

Search in Soffid 4

In Soffid 4 the findObjectByJsonQuery method no longer exists, it has been replaced by findObjects with or without pagination.

List users

```
q = new com.soffid.zkdb.api.Query();
pr = serviceLocator.getUserService().findUsers(q);
lu = pr.getResources();
for (i=0; i<lu.size(); i++) {
    u = lu.get(i);
    out.println(u.userName);
}
```

List users with pagination

```
us = serviceLocator.getUserService();
q = new com.soffid.zkdb.api.Query();
q.startIndex = 0;
q.pageSize = 2;
do {
    out.println("Searching...");
    pr = us.findUsers(q);
    lu = pr.getResources();
    for (i=0; i<lu.size(); i++) {
        u = lu.get(i);
        out.println(" "+u.userName);
    }
    q.startIndex = q.startIndex+pr.itemsPerPage;
} while (q.startIndex<pr.totalResults);
```

Revision #15

Created 13 August 2025 14:01:28 by Sion Vives

Updated 4 March 2026 16:14:02 by David Gómez