

Useful shell scripts



✂

Your Rating: ☆☆☆☆☆ Results: ★★★★★ 157 rates

The following are a collection of potentially useful scripts to be used with the [Magnolia shell module](#).

Also see [Groovy Shell Scripts](#) (some duplication, feel free to merge)

- [Run a script from your IDE](#)
- [Delete same-name siblings](#)
- [Bootstrap](#)
- [Execute module installation tasks](#)
- [Replace some value in multiple nodes](#)
- [Search and replace](#)
- [Update paragraph template](#)

There are also some more scripts at <http://svn.magnolia-cms.com/view/community/projects/magnolia-documentation/trunk/tools-scripts/> - unsorted, undocumented, comes with no warranty. Use at your own risk, but feel to use, copy, modify, and share !

Run a script from your IDE

```
bsh.Remote http://localhost:8080/webapp/.magnolia/shell?mgnlUserId=superuser&mgnlUserPSWD=superuser src
/scripts/renameBkToStk.bsh
```

Delete same-name siblings

```
name = "pagename";
for(node : parent.JCRNode.getNodes(name)){
    print("will delete " + node.path);
}

while(parent.hasContent(name)){
    root.delete(name);
    print("delete sibling");
}
hm.save();
```

Bootstrap

```
try{
  ModuleUtil.bootstrap(new String[]{"mgnl-bootstrap/core/config.server.filters.xml"}, true, 1);
}
catch(e){
  e.printStackTrace();
}
```

Execute module installation tasks

(in this example, adding a servlet)

```
registry = ModuleRegistry.Factory.getInstance();

def = registry.getDefinition("adminInterface");

installCtx = new InstallContext(){
  getConfigHierarchyManager(){
    print(hm);
    return hm;
  }
  invoke(method, args){
    print(method + "not implemented");
  }
};

for(servlet: def.servlets){
  task = new RegisterServletTask(servlet);
  task.execute(installCtx);
  print(servlet.name);
}
hm.save();
```

Replace some value in multiple nodes

(e.g. replace old dialog names with new ones)



Any kind of bulk operations can render your repository useless. Make sure you run script just printing out the values of what you want to change before having it to save the results of your changes.

```
import info.magnolia.cms.core.*;
import info.magnolia.cms.util.*;
import info.magnolia.context.*;

ctx = MgnlContext.getSystemContext();
MgnlContext.setInstance(ctx);
hm = ctx.getHierarchyManager("config");

pars = hm.getContent("/modules/yourmodule/paragraphs");
ContentUtil.visit(pars, new ContentUtil.Visitor() {
  visit(Content node) {
    dialog = node.getNodeData("dialogPath");
    print("old: " + dialog.string);
    dialog.value = "new" + dialog.string;
    print("new: " + dialog.string);
  }
});

//hm.save(); // uncomment when sure that you want to proceed.
```

Search and replace

```
import info.magnolia.cms.core.*;
import info.magnolia.context.MgnlContext;
import info.magnolia.cms.util.ContentUtil;
import org.apache.commons.lang.StringUtils;
import java.util.regex.*;

final String path = "/";
final String propertyName = "foo";
final String search = "bar"; // regex
final String replace = "baz";
final boolean save = false;

final HierarchyManager hm = MgnlContext.getHierarchyManager("website");
final Content root = hm.getContent(path);

if (root == null) {
    System.out.println("Node wasn't found for " + root);
    return;
}
System.out.println("");

final Pattern pattern = Pattern.compile(search);//, Pattern.LITERAL);

ContentUtil.visit(root, new ContentUtil.Visitor() {
    public void visit(Content node) throws Exception {
        final NodeData prop = node.getNodeData(propertyName);
        if (prop.isExist()) {
            final String oldValue = prop.getString();
            final StringBuffer newValue = new StringBuffer();
            final Matcher matcher = pattern.matcher(oldValue);
            int count = 0;
            while (matcher.find()) {
                final String group = matcher.group();
                matcher.appendReplacement(newValue, replace);
                count++;
            }
            matcher.appendTail(newValue);
            System.out.println(String.format("Found %d occurrences of %s at %s", count, search, prop.getHandle()));
            System.out.println(String.format("    old value:\n %s\n\nnew value:\n%s", oldValue, newValue.toString()));
            prop.setValue(newValue.toString());
        }
    }
});

System.out.println("");
System.out.println("Done !");
if (save) {
    System.out.println("Saving ...");
    hm.save();
    System.out.println("... Saved :)");
}
```

Update paragraph template

```
import info.magnolia.cms.core.search.*;
import info.magnolia.cms.util.*;
import javax.jcr.*;

update(oldName, newName, processor){
    query = "select * from nt:base where mgnl:template = '" + oldName + "'";

    nodes = QueryUtil.query("website",query , Query.SQL, "mgnl:contentNode");

    for(node:nodes){
        node.getMetaData().setTemplate(newName);
        if(processor!=null){
            processor.process(node);
        }
        node.save();
        print("updated: " + node);
    }
}

// example processor
AddListProcessor(){
    process(node){
        nd = node.createNodeData("hasLinkList", PropertyType.BOOLEAN);
        nd.setValue(true);
    }
    return this;
}

// examples
update("bkPromosTeaserInternal", "bkPromo", null);
update("bkTeaserLinkList", "bkTeaser", AddListProcessor());
```