

## JAVASCRIPT examples

## 1 – getting script engine info, &amp; hello world

## Java game application:

```
import javax.script.ScriptEngine;
import javax.script.ScriptEngineFactory;
import javax.script.ScriptEngineManager;
import javax.script.ScriptException;
import java.io.*;
import java.util.*;

public class MyGame
{
    public static void main(String[] args)
    {
        MyGame m = new MyGame();

        ScriptEngineManager factory = new ScriptEngineManager();
        String scriptFileName = "hello.js";

        // get a list of the script engines on this platform
        List<ScriptEngineFactory> list = factory.getEngineFactories();

        System.out.println("Script Engine Factories found:");
        for (ScriptEngineFactory f : list)
        {
            System.out.println("  Name = " + f.getEngineName()
                               + "  language = " + f.getLanguageName()
                               + "  extensions = " + f.getExtensions());
        }

        // get the JavaScript engine
        ScriptEngine jsEngine = factory.getEngineByName("js");

        // run the script
        m.executeScript(jsEngine, scriptFileName);
    }

    private void executeScript(ScriptEngine engine, String scriptFileName)
    {
        try
        {
            FileReader fileReader = new FileReader(scriptFileName);
            engine.eval(fileReader); //execute the script statements in the file
            fileReader.close();
        }
        catch (FileNotFoundException e1)
        {
            System.out.println(scriptFileName + " not found " + e1);
        }
        catch (IOException e2)
        {
            System.out.println("IO problem with " + scriptFileName + e2);
        }
        catch (ScriptException e3)
        {
            System.out.println("ScriptException in " + scriptFileName + e3);
        }
        catch (NullPointerException e4)
        {
            System.out.println("Null ptr exception in " + scriptFileName + e4);
        }
    }
}
```

## “hello.js” (javascript file)

```
print("hello world!");
```

Java game application - in *setUpScene()*:

```
...
// prepare the script engine
ScriptEngineManager factory = new ScriptEngineManager();
java.util.List<ScriptEngineFactory> list = factory.getEngineFactories();
jsEngine = factory.getEngineByName("js");

// use spin speed setting from the first script to initialize dolphin rotation
scriptFile1 = new File("InitParams.js");
this.executeScript(jsEngine, scriptFile1);
rc = new RotationController(Vector3f.createUnitVectorY(),
                           ((Double)jsEngine.get("spinSpeed")).floatValue());
rc.addNode(dolphinN);
sm.addController(rc);

// add the light specified in the second script to the game world
scriptFile2 = new File("CreateLight.js");
jsEngine.put("sm", sm);
this.executeScript(jsEngine, scriptFile2);
SceneNode plightNode =
    sm.getRootSceneNode().createChildSceneNode("plightNode");
plightNode.attachObject((Light)jsEngine.get("plight"));
...

protected void update(Engine engine)
{
    ...
    // same as before, plus the following:

    // run script again in update() to demonstrate dynamic modification
    long modTime = scriptFile1.lastModified();
    if (modTime > fileLastModifiedTime)
    {
        fileLastModifiedTime = modTime;
        this.runScript(scriptFile1);
        rc.setSpeed(((Double)jsEngine.get("spinSpeed")).floatValue());
    }
}
```

## “InitParams.js” (javascript file)

```
// sets a simple parameter - in this case the spin speed of the dolphin
var spinSpeed = 0.02;
```

## “CreateLight.js” (javascript file)

```
var JavaPackages = new JavaImporter(
    Packages.rage.scene.SceneManager,
    Packages.rage.scene.Light,
    Packages.rage.scene.Light.Type,
    Packages.rage.scene.Light.Type.POINT,
    Packages.java.awt.Color
);

// creates a RAGE object - in this case a light
with (JavaPackages)
{
    var plight = sm.createLight("testLamp1", Light.Type.POINT);
    plight.setAmbient(new Color(.3, .3, .3));
    plight.setDiffuse(new Color(.7, .7, .7));
    plight.setSpecular(new Color(1.0, 1.0, 1.0));
    plight.setRange(5);
}
```