

# web

webYellowfin

webYellowfin

YellowfinYellowfinYellowfin**ConfigTypeCodeSYSTEM**

Yellowfin

LoginId	String	Yellowfin WebIDID Yellowfin <b>Yellowfin Web</b>
Password	String	
OrgId	Integer	YellowfinID1
Function	String	Web <b>SAVECONFIGURATION</b>
Parameter	String[]	<b>configCodeconfigData</b>
Client	<a href="#">AdministrationClientOrg</a>	

**AdministrationClientOrg**

ClientReferenceId	String	ID

SOAP XML

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>SAVECONFIGURATION</function>
        <parameters>Simple_Authentication</parameters>
        <parameters>true</parameters>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

StatusCode	String	Web <ul style="list-style-type: none"> <li>• SUCCESS</li> <li>• FAILURE</li> </ul>

## SOAP

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>15359eb5b3d7d2c63c6f43affe5f3b45</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

## Java

- web

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);
rsr.setFunction("SAVECONFIGURATION");
```

- Yellowfin**Simple**

```
rsr.setParameters(new String[]{"SIMPLE_AUTHENTICATION","TRUE"});
```

- 

```
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
```

web

- StatusCode

```

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
}
else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
}

```

web

1. **ws\_saveconfiguration.jsp**
2. rootYellowfin/appserver/webapps/ROOT
- 3.
4. **http://<host>:<port>/ws\_saveconfiguration.jsp**

```

<%
/*          ws_SAVECONFIGURATION.jsp          */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
/*
This example sets Yellowfin authentication method to Simple.
That means, once Yellowfin has been restarted, the
LOGINUSERNOPASSWORD call can be used to log users into Yellowfin with no password provided.
*/

AdministrationServiceService s_adm = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_adm.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin webservices admin account
rsr.setPassword("test");                          // set to the password of the account above
rsr.setOrgId(1);
rsr.setFunction("SAVECONFIGURATION");

rsr.setParameters(new String[]{"SIMPLE_AUTHENTICATION","TRUE"});
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
}
else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>

```

YellowfinYellowfinYellowfin**ConfigTypeCodeCUSTOM**

Yellowfin

LoginId	String	Yellowfin WebIDID Yellowfin <b>Yellowfin Web</b>
Password	String	
OrgId	Integer	YellowfinID1
Function	String	Web <b>SAVECUSTOMPARAMETER</b>
Parameters	String[]	<b>configCodeconfigData</b>
Client	<a href="#">AdministrationClientOrg</a>	

**AdministrationClientOrg**

ClientReferenceId	String	ID

SOAP XML

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>SAVECUSTOMPARAMETER</function>
        <parameters>Simple_Authentication</parameters>
        <parameters>true</parameters>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

StatusCode	String	Web <ul style="list-style-type: none"> <li>• SUCCESS</li> <li>• FAILURE</li> </ul>

SOAP

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>471080457c1359cbf09c29cd348f97d1</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

## Java

- web

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);
rsr.setFunction("SAVECUSTOMPARAMETER");
```

- Yellowfin**Simple**

```
rsr.setParameters(new String[]{"SIMPLE_AUTHENTICATION", "TRUE"});
```

- 

```
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
```

## web

- StatusCode

```
if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
}
else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
}
```

## web

1. **ws\_savecustomparameter.jsp**
2. rootYellowfin/appserver/webapps/ROOT
- 3.
4. **http://<host>:<port>/ws\_savecustomparameter.jsp**

```

<%
/*          ws_savecustomparameter.jsp          */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%

/*
    This example set Yellowfin authentication method to Simple.
    That means, once Yellowfin has been restarted,
    LOGINUSERNOPASSWORD call can be used to log users into Yellowfin with no password provided.
*/

AdministrationServiceService s_adm = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);    // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_adm.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");           // provide your Yellowfin web services admin account
rsr.setPassword("test");                           // change to the password of the account above
rsr.setOrgId(1);
rsr.setFunction("SAVECUSTOMPARAMETER");

rsr.setParameters(new String[] {"SIMPLE_AUTHENTICATION","TRUE"});
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
    }
    else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
    }

%>

```

webYellowfinConfigCodeYellowfinconfigDataYellowfinConfigTypeCodeSYSTEM

simple

LoginId	String	Yellowfin WebIDID  Yellowfin <b>Yellowfin Web</b>
Password	String	
OrgId	Integer	YellowfinID1
Function	String	Web <b>LOADCONFIGURATION</b>
Parameter	String[]	ConfigCode
Client	<a href="#">AdministrationClientOrg</a>	

AdministrationClientOrg

ClientReferenceId	String	ID

StatusCode	String	Web <ul style="list-style-type: none"><li>SUCCESS</li><li>FAILURE</li></ul>
BinaryData	String	YellowfinConfiguration.configCodeconfigDataConfigCodeNull

Java

- web

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);
rsr.setFunction("LOADCONFIGURATION");
```

- YellowfinSimple

```
rsr.setParameters(new String[]{"SIMPLE_AUTHENTICATION"});
```

- 

```
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
```

web

- 

```
if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
    String data = rs.getBinaryData();
    out.write("<br>ConfigData: " + data);
}

else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
}
```

web

1. **ws\_loadconfiguration.jsp**
2. rootYellowfin/appserver/webapps/ROOT
- 3.
4. **http://<host>:<port>/ws\_loadconfiguration.jsp**

```
<%
/*                                ws_loadconfiguration.jsp                                */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
AdministrationServiceService s_admin = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_admin.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin webservices admin account
rsr.setPassword("test");                          // change to be the password of
the account above
rsr.setOrgId(1);
rsr.setFunction("LOADCONFIGURATION");

rsr.setParameters(new String[]{"SIMPLE_AUTHENTICATION"});
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
String data = rs.getBinaryData();
    out.write("<br>ConfigData: " + data);
}
else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>
```

webYellowfinConfigCodeYellowfinconfigDataYellowfinConfigTypeCode**CUSTOM**

simple

LoginId	String	Yellowfin WebIDID Yellowfin <b>Yellowfin Web</b>
Password	String	
OrgId	Integer	YellowfinID1
Function	String	Web <b>LOADCUSTOMPARAMETER</b>
Parameter	String[]	ConfigCode
Client	<a href="#">AdministrationClientOrg</a>	

**AdministrationClientOrg**



ClientReferenceId	String	ID

StatusCode	String	Web <ul style="list-style-type: none"> <li>• SUCCESS</li> <li>• FAILURE</li> </ul>
BinaryData	String	YellowfinConfiguration.configCodeconfigDataConfigCodeNull

Java

- web

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);
rsr.setFunction("LOADCUSTOMPARAMETER");
```

- Yellowfin**Simple**

```
rsr.setParameters(new String[]{"SIMPLE_AUTHENTICATION"});
```

- 

```
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
```

web

- 

```
if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
    String data = rs.getBinaryData();
    out.write("<br>ConfigData: " + data);
}

else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
}
```

web

1. **ws\_loadcustomparameter.jsp**
2. rootYellowfin/appserver/webapps/ROOT
- 3.
4. **http://<host>:<port>/ws\_loadcustomparameter.jsp**

```

/*          ws_loadcustomparameter.jsp          */
%>
<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>

<%
AdministrationServiceService s_adm = new AdministrationServiceServiceLocator("localhost",8080, "/services
/AdministrationService", false);          // adjust host and port number
AdministrationServiceSoapBindingStub adminService = (AdministrationServiceSoapBindingStub) s_adm.
getAdministrationService();
AdministrationServiceRequest rsr = new AdministrationServiceRequest();

rsr.setLoginId("admin@yellowfin.com.au");          // provide your Yellowfin web services admin account
rsr.setPassword("test");          // set to the password of the account above
rsr.setOrgId(1);
rsr.setFunction("LOADCUSTOMPARAMETER");

rsr.setParameters(new String[] {"SIMPLE_AUTHENTICATION"});
AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
String data = rs.getBinaryData();
    out.write("<br>ConfigData: " + data);
}
else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>

```

web/

webYellowfin

**report consumer**Yellowfin

DISABLEADMINMODE

LoginId	String	Yellowfin WebIDID
		Yellowfin <b>Yellowfin Web</b>

Password	String	
OrgId	Integer	YellowfinID1
Function	String	Web <b>ENABLEADMINMODE</b>
RoleFunctionList	Map<String, String>	CRUDAdministrationFunction
Parameter	String[]	trueTRUEfalse

SOAP

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <orgRef>org1</orgRef>
        <function>ENABLEADMINMODE</function>
        <parameters>true</parameters>
        <roleFunctionList>
          <key>MIREPORT</key>
          <value>CRUD</value>
        </roleFunctionList>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

StatusCode	String	Web <ul style="list-style-type: none"><li>SUCCESS</li><li>FAILURE</li></ul>

SOAP

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>517ce15c32a079dc1528424c03d86691</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Java

- web

```
AdministrationServiceRequest asr = new AdministrationServiceRequest();

asr.setLoginId("admin@yellowfin.com.au");
asr.setPassword("test");
asr.setOrgId(new Integer(1));
asr.setFunction("ENABLEADMINMODE");
```

- 

```
asr.setParameters(new String[] { "true" });
```

- 

```
Map<String, String> roleFunctionList = new HashMap<>();

roleFunctionList.put("SYSTEMINFO", "");
roleFunctionList.put("REPORTDASHBOARD", "R");
roleFunctionList.put("DASHPUBLIC", "R");
roleFunctionList.put("MIREPORT", "R");
roleFunctionList.put("STORYBOARD", "R");
```

- 

```
asr.setRoleFunctionList(roleFunctionList);
```

- 

```
AdministrationServiceResponse rs = rssbs.remoteAdministrationCall(asr);
```

web

- StatusCode

web

1. **ws\_enableadminmode.jsp**
2. rootYellowfin/appserver/webapps/ROOT
- 3.
4. **http://<host>:<port>/ws\_enableadminmode.jsp**

```

<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
/*
Enable Admin Mode
Enables administrative mode with the list of functions in roleFunctionList.
A parameter is set to indicate this should notify the cluster.
*/

AdministrationServiceResponse rs = null;
AdministrationServiceRequest asr = new AdministrationServiceRequest();
AdministrationServiceService ts = new AdministrationServiceServiceLocator("localhost", 8080,
"/services/AdministrationService", false);
AdministrationServiceSoapBindingStub rssbs = (AdministrationServiceSoapBindingStub) ts
.getAdministrationService();

asr.setLoginId("admin@yellowfin.com.au");
asr.setPassword("test");
asr.setOrgId(new Integer(1));
asr.setFunction("ENABLEADMINMODE");
asr.setParameters(new String[] { "true" });

Map<String, String> roleFunctionList = new HashMap<>();
roleFunctionList.put("SYSTEMINFO", "");
roleFunctionList.put("REPORTDASHBOARD", "R");
roleFunctionList.put("DASHPUBLIC", "R");
roleFunctionList.put("MIREPORT", "R");
roleFunctionList.put("STORYBOARD", "R");

asr.setRoleFunctionList(roleFunctionList);

rs = rssbs.remoteAdministrationCall(asr);

if ("SUCCESS".equals(rs.getStatusCode())) {
for (String message : rs.getMessages()) {
out.write(message + "<BR>");
}
} else {
out.write("Failure");
out.write(" Code: " + rs.getErrorCode());
}
}%>

```

webYellowfintrueTRUEfalse/FALSE

LoginId	String	Yellowfin WebIDID YellowfinYellowfin Web
Password	String	
OrgId	Integer	YellowfinID1
Function	String	WebDISABLEADMINMODE
Parameter	String[]	trueTRUEfalse

## SOAP

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <orgRef>org1</orgRef>
        <function>DISABLEADMINMODE</function>
        <parameters>true</parameters>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

StatusCode	String	Web <ul style="list-style-type: none"><li>• SUCCESS</li><li>• FAILURE</li></ul>

## SOAP

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>b2aedaafa1350f99c904b88c5daa14e0</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

## Java

- web

```
AdministrationServiceRequest asr = new AdministrationServiceRequest();

asr.setLoginId("admin@yellowfin.com.au");
asr.setPassword("test");
asr.setOrgId(new Integer(1));
asr.setFunction("DISABLEADMINMODE");
```

- 

```
asr.setParameters(new String[] { "true" });
```

- 

```
AdministrationServiceResponse rs = rssbs.remoteAdministrationCall(asr);
```

web

- StatusCode

web

1. **ws\_disableadminmode.jsp**
2. rootYellowfin/appserver/webapps/ROOT
- 3.
4. **http://<host>:<port>/ws\_disableadminmode.jsp**

```

<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
/*
Disable Admin Mode
Disables administrative mode. A parameter is set to indicate this should notify the cluster.
*/
AdministrationServiceResponse rs = null;
AdministrationServiceRequest asr = new AdministrationServiceRequest();
AdministrationServiceService ts = new AdministrationServiceServiceLocator("localhost", 8080,
"/services/AdministrationService", false);
AdministrationServiceSoapBindingStub rssbs = (AdministrationServiceSoapBindingStub) ts
.getAdministrationService();
asr.setLoginId("admin@yellowfin.com.au");
asr.setPassword("test");
asr.setOrgId(new Integer(1));
asr.setFunction("DISABLEADMINMODE");
asr.setParameters(new String[] { "true" });
rs = rssbs.remoteAdministrationCall(asr);
if ("SUCCESS".equals(rs.getStatusCode())) {
for (String message : rs.getMessages()) {
out.write(message + "<BR>");
}
} else {
out.write("Failure");
out.write(" Code: " + rs.getErrorCode());
}
}%>

```

webYellowfin

- **Function Code**
- **Function Name**
- **Function Description**
- **Uses CRUDCRUD**

LoginId	String	Yellowfin WebIDID Yellowfin <b>Yellowfin Web</b>
Password	String	
OrgId	Integer	YellowfinID1
Function	String	Web <b>GETROLEFUNCTIONS</b>
Parameter	String[]	

SOAP



```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://webservices.web.
mi.hof.com/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:remoteAdministrationCall>
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <orgRef>org1</orgRef>
        <function>GETROLEFUNCTIONS</function>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>

```

StatusCode	String	Web <ul style="list-style-type: none"> <li>• SUCCESS</li> <li>• FAILURE</li> </ul>
QueryResults	ReportRow[]	

#### ReportRow

ReportRow		
Function Code	String	
Function Name	String	
Function Description	String	
Uses CRUD	String	CRUD

#### SOAP

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <queryResults>
          <dataValue>ACCESSFILTER</dataValue>
          <dataValue>Access Filter</dataValue>
          <dataValue>Allows users to set or change the access filter on a report</dataValue>
          <dataValue>0</dataValue>
        </queryResults>
        <queryResults>
          <dataValue>CONTENTACCESS</dataValue>
          <dataValue>Folder Access</dataValue>
          <dataValue>Allow users to configure content folders.</dataValue>
          <dataValue>0</dataValue>
        </queryResults>
        .
        .
        .
        <queryResults>
          <dataValue>SHOWHEADERFOOTERADMIN</dataValue>
          <dataValue>Header/Footer Admin</dataValue>
          <dataValue>Allow the user to edit the Header and Footer displayed throughout the system.<
/dataValue>
          <dataValue>0</dataValue>
        </queryResults>
        <sessionId>31e386b7f19b97d6dcdea223f6b2f737</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>

```

## Java

- web

```

AdministrationServiceRequest asr = new AdministrationServiceRequest();

asr.setLoginId("admin@yellowfin.com.au");
asr.setPassword("test");
asr.setOrgId(new Integer(1));
asr.setFunction("GETROLEFUNCTIONS");

```

- 

```

asr.setParameters(new String[] { "root" });

```

- 

```

AdministrationServiceResponse rs = rssbs.remoteAdministrationCall(asr);

```

web

- StatusCode

```
for (String message : rs.getMessages()) {  
    out.write(message + "<BR>");  
}
```

- HTML

```
out.write("<table>");  
out.write("<th>Function Code</th>");  
out.write("<th>Function Name</th>");  
out.write("<th>Function Description</th>");  
out.write("<th>Uses CRUD</th>");  
for (ReportRow row : rs.getQueryResults()) {  
    out.write("<tr>");  
    for (String data : row.getDataValue()) {  
        out.write("<th>" + data + "</th>");  
    }  
    out.write("</tr>");  
}  
out.write("</table>");
```

web

1. **ws\_getrolefunctions.jsp**
2. rootYellowfin/appserver/webapps/ROOT
- 3.
4. **http://<host>:<port>/ws\_getrolefunctions.jsp**

```

<%@ page language="java" contentType="text/html; charset=UTF-8" %>
<%@ page import="com.hof.util.*, java.util.*, java.text.*" %>
<%@ page import="com.hof.web.form.*" %>
<%@ page import="com.hof.mi.web.service.*" %>
<%
/*
Get Role Functions          ws_getrolefunctions.jsp
Returns all of the role functions available from the OrgFunction table with descriptions from the
OrgReferenceCodeDesc table. Also includes whether the function uses CRUD level access.
*/
AdministrationServiceResponse rs = null;
AdministrationServiceRequest asr = new AdministrationServiceRequest();
AdministrationServiceService ts = new AdministrationServiceServiceLocator("localhost", 8080, "/services
/AdministrationService", false);
AdministrationServiceSoapBindingStub rssbs = (AdministrationServiceSoapBindingStub) ts
.getAdministrationService();
asr.setLoginId("admin@yellowfin.com.au");
asr.setPassword("test");
asr.setOrgId(new Integer(1));
asr.setFunction("GETROLEFUNCTIONS");
asr.setParameters(new String[] { "root" }); // Database password

rs = rssbs.remoteAdministrationCall(asr);
if ("SUCCESS".equals(rs.getStatusCode())) {
for (String message : rs.getMessages()) {
out.write(message + "<BR>");
}
// Create a simple table with the results
out.write("<table>");
out.write("<th>Function Code</th>");
out.write("<th>Function Name</th>");
out.write("<th>Function Description</th>");
out.write("<th>Uses CRUD</th>");
for (ReportRow row : rs.getQueryResults()) {
out.write("<tr>");
for (String data : row.getDataValue()) {
out.write("<th>" + data + "</th>");
}
out.write("</tr>");
}
out.write("</table>");
} else {
out.write("Failure");
out.write(" Code: " + rs.getErrorCode());
}
%>

```