

Miscellaneous Web Services

This section includes some additional web services that you can use.

Testing Functions

The following web services can be used for testing or other basic functions:

This function is used to perform a test call to check if the Yellowfin server is active. It simply returns a Success message if you are able to talk to the server, otherwise a Failure message is returned.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
LoginId	String	An administrator account to connect to the Yellowfin web services. This can either be the user ID or the email address, depending on the Logon ID method. This account must have the “web services” role enabled, and must belong to the default (i.e. primary) org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "TEST".

Request Example

Below is a SOAP XML example for this request:

```
<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>TEST</function>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The returned response will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: <ul style="list-style-type: none">SUCCESSFAILURE

Response Example

The service will return the below response, according to our SOAP example:

```
<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>0c411f82a9f04ddc6c3336a6da294761</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Define the request for this function, which includes logging in as the admin user and specifying the web service call to perform:

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();
rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);
rsr.setFunction("TEST");
```

- Once the request is configured, simply perform the call to test the server:

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

Initialize the Administration web service. Click [here](#) to learn how to do this.

- Add the following code to retrieve the response, which will contain the StatusCode.

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

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following steps:

1. Copy the code and save it as `ws_test.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.

3. Adjust the host, port, and admin user details according to your environment.
4. Run `http://<host>:<port>/ws_test.jsp` from your Internet browser.

```
<%
/*
    ws_test.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 according to your setting

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("TEST");

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

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

*/
%>
```

This is another web service that can be used to test whether the Yellowfin server is active or not. It simply attempts to talk to the server and returns either a Success or Failure response.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
LoginId	String	An administrator account to connect to the Yellowfin web services. This can either be the user ID or the email address, depending on the Logon ID method. This account must have the "web services" role enabled, and must belong to the default (i.e. primary) org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "INFO".

Request Example

Below is a SOAP XML example for this request:

```
<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>INFO</function>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The returned response will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: <ul style="list-style-type: none">SUCCESSFAILURE

Response Example

The service will return the below response, according to our SOAP example:

```
<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>45a10389d35349476efd2c8246d72b69</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Define the request for this function, which includes logging in as the admin user and specifying the web service call to perform:

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();
rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);
rsr.setFunction("INFO");
```

- Once the request is configured, simply perform the call to test the server:

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

Initialize the Administration web service. Click [here](#) to learn how to do this.

- Add the following code to retrieve the response, which will contain the StatusCode.

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

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following the steps:

1. Copy the code and save it as ws_info.jsp.
2. Put the file in the root folder: *Yellowfin/appserver/webapps/ROOT*.
3. Adjust the host, port, and admin user details according to your environment.
4. Run *http://<host>:<port>/ws_info.jsp* from your Internet browser.

```

<%
/*
ws_info.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 according to your setting

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("INFO");

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

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

*/
%>

```

SQL Query Functions

This function is used to manage the Yellowfin database by using SQL query. For instance, you can retrieve information from the database by using the SELECT command, as well as manage the database tables by using DELETE and INSERT. Note that this function requires the password for the database, however this can be disabled by changing the setting, discussed below.

Click below to see how to enable this functionality for your environment, and change its setting.

1. Open the web.xml file from the WEB-INF directory (appserver\webapp\ROOT\WEB-INF\web.xml).
2. Then add the following code in that file:

```

<init-param>
    <param-name>EnableQueryWebservice</param-name>
    <param-value>TRUE</param-value>
</init-param>

```

3. By default this function will require the database password, but you can add the following code to the same file to disable this requirement. Otherwise skip this step.

```

<init-param>
  <param-name>DisableQueryWebservicePassword</param-name>
  <param-value>TRUE</param-value>
</init-param>

```

4. Save the web.xml file, and restart Yellowfin, if you had it running.
5. If this web service was not enabled, you will see the "Query Webservice not enabled" error in the Yellowfin logs after performing the METADATASQLQUERY call.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
LoginId	String	An administrator account to connect to the Yellowfin web services. This can either be the user ID or the email address, depending on the Logon ID method. This account must have the "web services" role enabled, and must belong to the default (i.e. primary) org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "METADATASQLQUERY".
Query	String	The SQL query that you want to perform on the Yellowfin database.
Parameters	String[]	Use this to pass the Yellowfin database password. Only the first element of the array is required (which you can set as the password). However, if this requirement has been disabled (as mentioned in the steps above, then this parameter will not be required.

Request Example

Below is a SOAP XML example for this request:

```

<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>METADATASQLQUERY</function>
        <query>SELECT * FROM person</query>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Parameters

The returned response will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: <ul style="list-style-type: none"> SUCCESS FAILURE
QueryResults	ReportRow	Array of the result set rows if SELECT was the request query.

Response Example

The service will return the below response, according to our SOAP example:

```
<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>5</dataValue>
          <dataValue>Administrator</dataValue>
          <dataValue>System</dataValue>
          <dataValue/>
          <dataValue/>
          <dataValue xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" />
          <dataValue>ATMN SSTM</dataValue>
          <dataValue/>
          <dataValue>System Administrator</dataValue>
          <dataValue>System Administrator</dataValue>
          <dataValue>EN</dataValue>
          <dataValue>PRIVATE</dataValue>
          <dataValue>4</dataValue>
          <dataValue>EMAIL</dataValue>
          <dataValue xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" />
          <dataValue xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" />
          <dataValue xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" />
          <dataValue>AUSTRALIA/SYDNEY</dataValue>
          <dataValue xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" />
          <dataValue xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" />
          <dataValue xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" />
          <dataValue>ACTIVE</dataValue>
        </queryResults>
        <sessionId>804451ca30ad4a3065e4b67a9293440c</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Define the request for this function, which includes logging in as the admin user and specifying the web service call to perform:


```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();
rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);
rsr.setFunction("METADATASQLQUERY");
```

- Provide the SQL query that you wish to perform:

```
rsr.setQuery("SELECT * FROM configuration WHERE ConfigTypeCode = 'SYSTEM'");
```

- Once the request is configured, simply perform the call to test the server:

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

Initialize the Administration web service. Click [here](#) to learn how to do this.

- Add the following code to retrieve the response. (See the Response Parameter table above for details.)

```
if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
    ReportRow[] rows = rs.getQueryResults();
    if (rows != null)
        for (ReportRow rr: rows){
            String[] str = rr.getDataValue();
            out.write("<br>");
            for (String s: str)
                out.write

(" <br>" + s);

        }
    }
else {
    out.write("<br>Failure");
}
}
```

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following the steps:

1. Copy the code and save it as ws_metadatasqlquery.jsp.
2. Put the file in the root folder: *Yellowfin/appserver/webapps/ROOT*.
3. Adjust the host, port, and admin user details according to your environment.
4. Run *http://<host>:<port>/ws_metadatasqlquery.jsp* from your Internet browser.

```

<%
/*          ws_metadatasqlquery.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");          // change to the password of the account above
rsr.setOrgId(1);
rsr.setFunction("METADATASQLQUERY");

rsr.setQuery("SELECT * FROM configuration WHERE ConfigTypeCode = 'SYSTEM'");

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
    ReportRow[] rows = rs.getQueryResults();
    if (rows != null)
        for (ReportRow rr: rows){
            String[] str = rr.getDataValue();
            out.write("<br>");
            for (String s: str)
                out.write("<br>"

+ s);

        }

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

```

Folder Management Functions

The below web services are used to manage the Yellowfin folders or sub-folders (previously known as categories/subcategories) in which reports, dashboards, and other content can be saved:

This web service returns all the categories and subcategories (that is, folders and sub-folders) in Yellowfin's Browse page.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
-----------------	-----------	-------------

LoginId	String	An administrator account to connect to the Yellowfin web services. This can either be the user ID or the email address, depending on the Logon ID method. This account must have the “web services” role enabled, and must belong to the default (i.e. primary) org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "GETCATEGORIES".
OrgRef	String	Client org reference ID to use this service on a particular client org. If this is not specified, the default org. will be selected.

Request Example

Below is a SOAP XML example for this request:

```
<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>GETCATEGORIES</function>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The returned response will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: <ul style="list-style-type: none"> SUCCESS FAILURE
ContentResources	ContentResource[]	Object array containing details of folders and subfolders.

Response Example

The service will return the below response, according to our SOAP example:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <contentResources>
          <resourceCode>AUDITREPORTS</resourceCode>
          <resourceDescription>Audit Reports</resourceDescription>
          <resourceId>56339</resourceId>
          <resourceName>Audit Reports</resourceName>
          <resourceOrgId>1</resourceOrgId>
        </contentResources>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

```
<resourceType>RPTCATEGORY</resourceType>
<resourceUUID>a6bdc6b5-a832-42a2-98c7-18273900d0aa</resourceUUID>
</contentResources>
<contentResources>
  <resourceCode>ADMINREPORTS</resourceCode>
  <resourceDescription>Admin Reports</resourceDescription>
  <resourceId>56340</resourceId>
  <resourceName>Admin Reports</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>RPTSUBCATEGORY</resourceType>
  <resourceUUID>f7fb32b7-1573-4899-916f-c34afb9a865d</resourceUUID>
</contentResources>
<contentResources>
  <resourceCode>CONTENTUSAGE</resourceCode>
  <resourceDescription>Content Usage</resourceDescription>
  <resourceId>56341</resourceId>
  <resourceName>Content Usage</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>RPTSUBCATEGORY</resourceType>
  <resourceUUID>6bae5230-clf9-4491-8a8b-f14blae660d7</resourceUUID>
</contentResources>
<contentResources>
  <resourceCode>USERACCESS</resourceCode>
  <resourceDescription>User Access</resourceDescription>
  <resourceId>56342</resourceId>
  <resourceName>User Access</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>RPTSUBCATEGORY</resourceType>
  <resourceUUID>0c7ddde4-fa03-4e88-b37b-7b5e4aad5e1d</resourceUUID>
</contentResources>
<contentResources>
  <resourceCode>TUTORIAL</resourceCode>
  <resourceDescription>Tutorial</resourceDescription>
  <resourceId>60706</resourceId>
  <resourceName>Tutorial</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>RPTCATEGORY</resourceType>
  <resourceUUID>a23c2ec6-a2fa-45c7-b5da-dcf3f02e6633</resourceUUID>
</contentResources>
<contentResources>
  <resourceCode>ATHLETES</resourceCode>
  <resourceDescription>Athletes</resourceDescription>
  <resourceId>60707</resourceId>
  <resourceName>Athletes</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>RPTSUBCATEGORY</resourceType>
  <resourceUUID>72e4b4bd-a482-4a01-a031-c6ab76dbb3a5</resourceUUID>
</contentResources>
<contentResources>
  <resourceCode>CAMP</resourceCode>
  <resourceDescription>Camp</resourceDescription>
  <resourceId>60708</resourceId>
  <resourceName>Camp</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>RPTSUBCATEGORY</resourceType>
  <resourceUUID>465411e5-594b-478e-af64-c0f59fc4546f</resourceUUID>
</contentResources>
<contentResources>
  <resourceCode>KPIS</resourceCode>
  <resourceDescription>KPIs</resourceDescription>
  <resourceId>60709</resourceId>
  <resourceName>KPIs</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>RPTSUBCATEGORY</resourceType>
  <resourceUUID>d514c643-dc01-4781-8905-d34e761ccd19</resourceUUID>
</contentResources>
<contentResources>
  <resourceCode>MARKETINGBOOKING</resourceCode>
  <resourceDescription>Marketing & Booking</resourceDescription>
  <resourceId>60710</resourceId>
  <resourceName>Marketing & Booking</resourceName>
```

```

        <resourceOrgId>1</resourceOrgId>
        <resourceType>RPTSUBCATEGORY</resourceType>
        <resourceUUID>db6d0a3-c088-4d71-b65a-f383aaa54be9</resourceUUID>
    </contentResources>
    <contentResources>
        <resourceCode>TRAINING</resourceCode>
        <resourceDescription>Training</resourceDescription>
        <resourceId>60711</resourceId>
        <resourceName>Training</resourceName>
        <resourceOrgId>1</resourceOrgId>
        <resourceType>RPTSUBCATEGORY</resourceType>
        <resourceUUID>c503ea57-cc69-43a9-98bc-a90ebbelc864</resourceUUID>
    </contentResources>
    <errorCode>0</errorCode>
    <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
    <messages>Web Service Request Complete</messages>
    <sessionId>fd3afecb73fe48578501f29e4d00065b</sessionId>
    <statusCode>SUCCESS</statusCode>
    </return>
</ns2:remoteAdministrationCallResponse>
</S:Body>
</S:Envelope>

```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Define the request for this function, which includes logging in as the admin user and specifying the web service call to perform:

```

AdministrationServiceRequest rsr = new AdministrationServiceRequest();
rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);
rsr.setFunction("GETCATEGORIES");

```

- Once the request is configured, simply perform the call to test the server:

```

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

```

Initialize the Administration web service. Click [here](#) to learn how to do this.

- Add the following code to retrieve the response that includes details of the folders/subfolders. (See the Response Parameter table above for details.)

```

if ( "SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
    ContentResource[] cr = rs.getContentResources();
    for (ContentResource x: cr){
        out.write("resourceCode: " + x.getResourceCode() + "<br>");
        out.write("resourceDescription: " + x.getResourceDescription() +
"<br>");
        out.write("resourceId: " + x.getResourceId() + "<br>");
        out.write("resourceName: " + x.
getResourceName() + "<br>");
        out.write("resourceOrgId: " + x.
getResourceOrgId() + "<br>");
        out.write("resourceType: " + x.
getResourceType() + "<br>");
        out.write("resourceUUID: " + x.
getResourceUUID() + "<br><br>");
    }
}
else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
}

```

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following steps:

1. Copy the code and save it as `ws_getcategories.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
3. Adjust the host, port, and admin user details according to your environment.
4. Run `http://<host>:<port>/ws_getcategories.jsp` from your Internet browser.

```

<%
/*                                ws_getcategories.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 webservices admin account
rsr.setPassword("test");                          // change to be the password of the account above
rsr.setOrgId(1);
rsr.setFunction("GETCATEGORIES");
//rsr.setOrgRef("org1");

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
    ContentResource[] cr = rs.getContentResources();
    for (ContentResource x: cr){
        out.write("resourceCode: " + x.getResourceCode() + "<br>");
        out.write("resourceDescription: " + x.getResourceDescription() +
"<br>");
        out.write("resourceId: " + x.getResourceId() + "<br>");
        out.write("resourceName: " + x.
getResourceName() + "<br>");
        out.write("resourceOrgId: " + x.
getResourceOrgId() + "<br>");
        out.write("resourceType: " + x.
getResourceType() + "<br>");
        out.write("resourceUUID: " + x.
getResourceUUID() + "<br><br>");
    }
}
else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>

```

This web service returns all the categories and subcategories (folders and sub-folders) in Yellowfin's Browse page, which are in the draft mode (that is not active yet).

Note: This function is only relevant for older versions of Yellowfin, as starting from Yellowfin 7.3, categories/subcategories cannot be saved as drafts.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
-----------------	-----------	-------------

LoginId	String	An administrator account to connect to the Yellowfin web services. This can either be the user ID or the email address, depending on the Logon ID method. This account must have the “web services” role enabled, and must belong to the default (i.e. primary) org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "GETDRAFTCATEGORIES".
OrgRef	String	Client org reference ID to use this service on a particular client org. If this is not specified, the default org. will be selected.

Request Example

Below is a SOAP XML example for this request:

```
<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>GETDRAFTCATEGORIES</function>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The returned response will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: <ul style="list-style-type: none"> SUCCESS FAILURE
ContentResources	ContentResource []	Object array containing details of draft folders and subfolders.

Response Example

The service will return the below response, according to our SOAP example:


```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <contentResources>
          <resourceCode>REPORTS</resourceCode>
          <resourceDescription>Reports</resourceDescription>
          <resourceId>73674</resourceId>
          <resourceName>Reports</resourceName>
          <resourceOrgId>13004</resourceOrgId>
          <resourceType>RPTCATEGORY</resourceType>
          <resourceUUID>3bc780d7-6638-4520-b233-77ad6e24ae3d</resourceUUID>
        </contentResources>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>3f09ab77656b3632ab05786aa0fa4570</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>

```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Define the request for this function, which includes logging in as the admin user and specifying the web service call to perform:

```

AdministrationServiceRequest rsr = new AdministrationServiceRequest();
rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);
rsr.setFunction("GETDRAFTCATEGORIES");

```

- Once the request is configured, simply perform the call to test the server:

```

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

```

Initialize the Administration web service. Click [here](#) to learn how to do this.

- Add the following code to retrieve the response that includes details of the folders/subfolders. (See the Response Parameter table above for details.)

```

if ( "SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
    ContentResource[] cr = rs.getContentResources();
    for (ContentResource x: cr){
        out.write("resourceCode: " + x.getResourceCode() + "<br>");
        out.write("resourceDescription: " + x.getResourceDescription() +
"<br>");
        out.write("resourceId: " + x.getResourceId() + "<br>");
        out.write("resourceName: " + x.
getResourceName() + "<br>");
        out.write("resourceOrgId: " + x.
getResourceOrgId() + "<br>");
        out.write("resourceType: " + x.
getResourceType() + "<br>");
        out.write("resourceUUID: " + x.
getResourceUUID() + "<br><br>");
    }
}
else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
}

```

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following steps:

1. Copy the code and save it as `ws_getdraftcategories.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
3. Adjust the host, port, and admin user details according to your environment.
4. Run `http://<host>:<port>/ws_getdraftcategories.jsp` from your Internet browser.

```

<%
/*          ws_getdraftcategories.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");          // change to the password of the account above
rsr.setOrgId(1);
rsr.setFunction("GETDRAFTCATEGORIES");
//rsr.setOrgRef("org1");

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
    ContentResource[] cr = rs.getContentResources();
    for (ContentResource x: cr){
        out.write("resourceCode: " + x.getResourceCode() + "<br>");
        out.write("resourceDescription: " + x.getResourceDescription() +
"<br>");
        out.write("resourceId: " + x.getResourceId() + "<br>");
        out.write("resourceName: " + x.
getResourceName() + "<br>");
        out.write("resourceOrgId: " + x.
getResourceOrgId() + "<br>");
        out.write("resourceType: " + x.
getResourceType() + "<br>");
        out.write("resourceUUID: " + x.
getResourceUUID() + "<br><br>");
    }
}
else {
    out.write("<br>Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>

```

This web service deletes a specific category or subcategory (folder/sub-folder) from Yellowfin's Browse page.

Request Parameters

The following parameters should be passed with this request:

Request Element	Data Type	Description
-----------------	-----------	-------------

LoginId	String	An administrator account to connect to the Yellowfin web services. This can either be the user ID or the email address, depending on the Logon ID method. This account must have the "web services" role enabled, and must belong to the default (i.e. primary) org.
Password	String	Password of the above account.
OrgId	Integer	Default (i.e. primary) organization ID within Yellowfin. Always set this to 1.
Function	String	Web service function. Set this to "DELETECATEGORY".
ContentResources	ContentResource[]	Object array of Content Resources containing the description of the category (folder) to be deleted. See the table below.
OrgRef	String	Client org reference ID to use this service on a particular client org. If this is not specified, the default org. will be selected.

The below parameters are mandatory to be provided in the **ContentResource** object.

ContentType	FavouriteType	CreationCode
ResourceType	String	Specifies the content type. Set this to either RPTCATEGORY or RPTSUBCATEGORY.
ResourceCode	String	Unique code for the content type. Tip: If you don't know ResourceCode, you can find it out by calling the GETCATEGORIES call, or directly look for it in Yellowfin's database. (Filter the orgReferenceCodeDesc database table by RefTypeCode IN ('RPTCATEGORY', 'RPTSUBCATEGORY') where, <ul style="list-style-type: none"> • <i>ShortDescription</i> is folder/subfolder name; • <i>RefTypeCode</i> is a resourceType; • <i>RefCode</i> is a resourceCode.

Request Example

Below is a SOAP XML example for this request:

```
<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>DELETECATEGORY</function>
        <contentResources>
          <resourceTyp>RPTSUBCATEGORY</resourceTyp>
          <resourceCode>SUBCAT2ORGL</resourceCode>
        </contentResources>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Parameters

The returned response will contain these parameters:

Response Element	Data Type	Description
StatusCode	String	Status of the web service call. Possible values include: <ul style="list-style-type: none"> SUCCESS FAILURE

Response Example

The service will return the below response, according to our SOAP example:

```
<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>359f5fce5ce26028acb2432720995a62</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Instructions

See below for step-by-step instructions on how to perform this call, using a Java example:

- Define the request for this function, which includes logging in as the admin user and specifying the web service call to perform:

```
AdministrationServiceRequest rsr = new AdministrationServiceRequest();
rsr.setLoginId("admin@yellowfin.com.au");
rsr.setPassword("test");
rsr.setOrgId(1);
rsr.setFunction("DELETECATEGORY");
```

- Pass the details of the category or folder you want to delete:

```
ContentResource[] cr = new ContentResource[1];
cr[0] = new ContentResource();
cr[0].setResourceType("RPTSUBCATEGORY");
cr[0].setResourceCode("SUBCAT2ORG1"); // my sub category or subcategory code

rsr.setContentResources(cr);
```

- Once the request is configured, simply perform the call to test the server:

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

Initialize the Administration web service. Click [here](#) to learn how to do this.

- Add the following code to retrieve the response, containing the StatusCode. (See the Response Parameter table above for details.)

```

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

```

Complete Example

Below is a full example of this web service call. To use it for yourself, carry out the following the steps:

1. Copy the code and save it as `ws_deletecategory.jsp`.
2. Put the file in the root folder: `Yellowfin/appserver/webapps/ROOT`.
3. Adjust the host, port, and admin user details according to your environment.
4. Run `http://<host>:<port>/ws_deletecategory.jsp` from your Internet browser.

```

<%
/*          ws_deletecategory.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 webservices admin account
rsr.setPassword("test");          // change to the password of the account above
rsr.setOrgId(1);
rsr.setFunction("DELETECATEGORY");
rsr.setOrgRef("org1");

ContentResource[] cr = new ContentResource[1];
cr[0] = new ContentResource();
cr[0].setResourceType("RPTSUBCATEGORY");
cr[0].setResourceCode("SUBCAT2ORG1");          // my sub category or subcategory code

rsr.setContentResources(cr);

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

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

%>

```

