

webwebAPI

- o
- o
- o
- o
- o
- o
- o

web

web

- 1. YellowfinGETCONTENT
- 2. YellowfinContentResource
- 3. ContentResourceYellowfin
- 4. YellowfinresorceType

	ContentResource resourceType
	RPTCATEGORY
	RPTSUBCATEGORY
	DATASOURCE
	VIEW
	GROUP
	REPORT
	ETLPROCESS

- 5. GETEXPORTDEPENDRNCIESContentResource
- 6. ContentResourceContentResource
- 7. YellowfinXMLContentResourceEXPORTCONTENTYellowfin
- 8. GETEXPORTDEPENDENCIESEXPORTCONTENTweb

Yellowfin

LoginId	String	Yellowfin WebIDIDID Yellowfin Yellowfin Web
Password	String	
OrgId	Integer	YellowfinID1
Function	String	Web GETCONTENT
OrgRef	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>GETCONTENT</function>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

StatusCode	String	Web <ul style="list-style-type: none">SUCCESSFAILURE
ContentResources	ContentResource[]	Yellowfin

SOAP

```
<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>
          <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-c1f9-4491-8a8b-f14b1ae660d7</resourceUUID>
        </contentResources>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

```

</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>dbe6d0a3-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>
<contentResources>
  <resourceDescription/>
  <resourceId>70101</resourceId>
  <resourceName>Oracle database</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>DATASOURCE</resourceType>
</contentResources>

```

```

<contentResources>
  <resourceDescription/>
  <resourceId>70109</resourceId>
  <resourceName>Oracle</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>DATASOURCE</resourceType>
</contentResources>
<contentResources>
  <resourceDescription>Ski Team is the demonstration and tutorial database.</resourceDescription>
  <resourceId>54700</resourceId>
  <resourceName>Ski Team</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>DATASOURCE</resourceType>
</contentResources>
<contentResources>
  <resourceDescription/>
  <resourceId>54701</resourceId>
  <resourceName>Yellowfin Configuration Database</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>DATASOURCE</resourceType>
</contentResources>
<contentResources>
  <resourceDescription/>
  <resourceId>70108</resourceId>
  <resourceName>c</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>DATASOURCE</resourceType>
</contentResources>
<contentResources>
  <resourceDescription/>
  <resourceId>70110</resourceId>
  <resourceName>f</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>DATASOURCE</resourceType>
</contentResources>
<contentResources>
  <resourceDescription>This is a tutorial view for training and demo purposes.<
/resourceDescription>
  <resourceId>60543</resourceId>
  <resourceName>Ski Team</resourceName>
  <resourceType>VIEW</resourceType>
  <resourceUUID>e3632adb-5194-460c-a172-c085416f493f</resourceUUID>
</contentResources>
<contentResources>
  <resourceDescription>This view should be used to monitor usage of Yellowfin by User and Content
Type.</resourceDescription>
  <resourceId>56169</resourceId>
  <resourceName>Yellowfin Usage Audit</resourceName>
  <resourceType>VIEW</resourceType>
  <resourceUUID>Fb6416c4-441e-42b3-a442-e7426f25f6b4</resourceUUID>
</contentResources>
<contentResources>
  <resourceDescription>This dashboard contains a set of reports covering general system and admin
information, including performance, sessions, data source & view usage.</resourceDescription>
  <resourceId>57438</resourceId>
  <resourceName>Admin</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>GROUP</resourceType>
  <resourceUUID>33827292-cda6-4071-965f-730ccbc53519</resourceUUID>
</contentResources>
<contentResources>
  <resourceDescription>This is an analytic tab that is used to understand examine metrics split by
various demographics and filters.</resourceDescription>
  <resourceId>61195</resourceId>
  <resourceName>Analysis</resourceName>
  <resourceOrgId>1</resourceOrgId>
  <resourceType>GROUP</resourceType>
  <resourceUUID>f19e63f5-7175-4c57-897d-ed865aba8972</resourceUUID>
</contentResources>
</contentResources>
<errorCode>0</errorCode>

```

```

        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>92029c8ae4f1db0f33bf0b7370c1088a</sessionId>
        <statusCode>SUCCESS</statusCode>
    </return>
</ns2:remoteAdministrationCallResponse>
</S:Body>
</S:Envelope>

```

Java

- web

```

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

```

-

```

rsr.setOrgRef("org1");

```

-

```

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

```

web

- StatusCodeContentResource

web

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

```

<%
/*      ws_getcontent.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("GETCONTENT");

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);
    if ("SUCCESS".equals(rs.getStatusCode()) ) {
        out.write("<br>Success");
        ContentResource[] crs = rs.getContentResources();
        out.write("<table>");
        out.write("<tr><td> id </td><td> type </td><td> UUID </td></tr>");
        for (ContentResource c: crs) {
            out.write("<tr>");
            out.write("<td>" + c.getResourceId() + "</td><td>" + c.getResourceType() +
"</td><td>" + c.getResourceUUID() + "</td>");
            out.write("</tr>");
        }
    }
    else {
        out.write("<br>Failure");
        out.write(" Code: " + rs.getErrorCode());
    }
%>

```

YellowfinXML

LoginId	String	Yellowfin WebIDIDID YellowfinYellowfin Web
Password	String	
OrgId	Integer	YellowfinID1
Function	String	WebEXPORTCONTENT
OrgRef	String	ID
ContentResources	ContentResource	

ContentResource

ContentResource		
------------------------	--	--

ResourceID	Integer	ID
ResourceType	String	<ul style="list-style-type: none"> • RPTCATEGORY • RPTSUBCATEGORY • DATASOURCE • VIEW • GROUP • REPORT • ETLPROCESS
ResourceUUID	String	UUID

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>EXPORTCONTENT</function>
        <contentResources>
          <resourceId>56169</resourceId>
          <resourceType>VIEW</resourceType>
        </contentResources>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

StatusCode	String	Web <ul style="list-style-type: none"> • SUCCESS • FAILURE
BinaryAttachments	ReportBinaryObject[]	Yellowfin

ReportBinaryObject

ReportBinaryObject		
Key	String	EXPORT/XML
ContentType	String	MIMEtext/XML
Data	Byte[]	XML

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>
        <function>EXPORTCONTENT</function>
        <contentResources>
          <resourceId>56169</resourceId>
          <resourceType>VIEW</resourceType>
        </contentResources>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

Java

- web

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

-

```
ContentResource[] cr = new ContentResource[1];
cr[0] = new ContentResource();
cr[0].setResourceId(70058);
cr[0].setResourceType("GROUP");
cr[0].setResourceOrgId(1);
```

-

```
rsr.setContentResources(cr);
```

-

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

web

- StatusCodeReportBinaryObject

web

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

```
<%
/*      ws_exportcontent.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.*" %>
<%@ page import="java.nio.file.Files" %>
<%@ page import="java.io.PrintWriter" %>
<%
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 web services admin account
rsr.setPassword("test");      // set to the password of the above account
rsr.setOrgId(1);
rsr.setFunction("EXPORTCONTENT");

// specify which dashboard to export:
ContentResource[] cr = new ContentResource[1];
cr[0] = new ContentResource();
cr[0].setResourceId(70058);
cr[0].setResourceType("GROUP");
cr[0].setResourceOrgId(1);

rsr.setContentResources(cr);

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
    byte[] data = rs.getBinaryAttachments()[0].getData();
    String xml = new String(data, "UTF-8");
    PrintWriter writer = new PrintWriter("/Applications/Yellowfin 7.4/YFexport.xml", "UTF-8");
    writer.println(xml);
    writer.close();

    ReportBinaryObject[] bo = rs.getBinaryAttachments();
    for (ReportBinaryObject o : bo){
        out.write("<br><br>Key: " + o.getKey());
        out.write("<br>Content Type: " + o.getContentType());
    }

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

ContentResourceIDGETCONTENT

LoginId	String	Yellowfin WebIDIDID YellowfinYellowfin Web
Password	String	
OrgId	Integer	YellowfinID1
Function	String	WebGETEXPORTDEPENDENCIES
OrgRef	String	ID
ContentResources	ContentResource	

ContentResource

ContentResource		
ResourceID	Integer	ID
ResourceType	String	<ul style="list-style-type: none">• RPTCATEGORY• RPTSUBCATEGORY• DATASOURCE• VIEW• GROUP• REPORT• ETLPROCESS
ResourceUUID	String	UUID

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>GETEXPORTDEPENDENCIES</function>
        <contentResources>
          <resourceId>56169</resourceId>
          <resourceType>VIEW</resourceType>
        </contentResources>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

--	--	--

StatusCode	String	Web <ul style="list-style-type: none"> • SUCCESS • FAILURE
ContentResources	ContentResource[]	

SOAP

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <contentResources>
          <resourceDescription/>
          <resourceId>54701</resourceId>
          <resourceName>Yellowfin Configuration Database</resourceName>
          <resourceOrgId>1</resourceOrgId>
          <resourceType>DATASOURCE</resourceType>
        </contentResources>
        <errorCode>0</errorCode>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>97d7f893d787daf2806a13cdfa6f09d3</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Java

- web

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

-

```
rsr.setOrgRef("org1");
```

- ContentResource

```
ContentResource[] cr = new ContentResource[1];

cr[0] = new ContentResource();
cr[0].setResourceId(70307);
cr[0].setResourceType("GROUP");
cr[0].setResourceOrgId(1);
```

-

```
rsr.setContentResources(cr);
```

-

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

web

- StatusCodeContentResource

web

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

```

<%
/*      ws_getexportdependencies.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.*" %>
<%@ page import="java.nio.file.Files" %>
<%@ page import="java.io.PrintWriter" %>
<%
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("GETEXPORTDEPENDENCIES");

ContentResource[] cr = new ContentResource[1];

cr[0] = new ContentResource();
cr[0].setResourceId(70307);
cr[0].setResourceType("GROUP");
cr[0].setResourceOrgId(1);

rsr.setContentResources(cr);

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");
    ContentResource[] crs = rs.getContentResources();
    out.write("<table>");
    out.write("<tr><td> id </td><td> type </td><td> UUID </td></tr>");
    for (ContentResource c: crs) {
        out.write("<tr>");
        out.write("<td>" + c.getResourceId() + "</td><td>" + c.getResourceType() + "</td><td>"
+ c.getResourceUUID() + "</td>");
        out.write("</tr>");
    }
} else {
    out.write("Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>

```

web

Yellowfinweb

1. webYFXXMLEXPORTCONTENT
 - YellowfinwebAPI
 - web
2. ContentResourceGETIMPORTCONTENT
3. ImportOptionIMPORTCONTENTIMPORTIMPORTCONTENTNOVALIDATIONYellowfinImportOptionoptionKey
4. ImportOption
5. TESTIMPORTCONTENTTESTIMPORTCONTENTNOVALIDATIONweb
6. IMPORTCONTENTYellowfindevelopment/examples/webservices/ws_admin_import.jsplImportOptionYellowfin/appserver/webapps/ROOT

http://<host>:<port>/ws_admin_import.jsp

ImportContentoptionValueoptionKey

optionKey	optionValue
SKIP	truefalse
OPTION	REPLACEADD
EXISTING	YellowfinUUID optionKey=OPTIONoptionValue=REPLACE
PASSWORD1	

YellowfinContentResourceGETIMPORTCONTENT

ResourceId	ResourceType	ResourceName	ResourceUUID
70031	DATASOURCE	CSV	null
70209	VIEW	months	b974f55a-269b-4a4b-b1c6-bf9b968be723
null	RPTCATEGORY	Tutorial	a23c2ec6-a2fa-45c7-b5da-dcf3f02e6633
null	RPTSUBCATEGORY	Reports	58834ae1-2f65-44c0-b6c3-7c9cd2f91bd5
70279	REPORT	My Report	fd3794b3-62c0-4cf8-bac0-755e68d9c41e

5ImportOptionitemIndexContentResourceindexContentResource

ResourceId	ResourceType	importOption itemIndex
70031	DATASOURCE	0
70209	VIEW	1
null	RPTCATEGORY	2
null	RPTSUBCATEGORY	3
70279	REPORT	4

ImportOption

itemIndex	optionKey	optionValue	
0	SKIP	true	
1	SKIP	true	
2	SKIP	true	
3	SKIP	true	
4	SKIP	false	
4	OPTION	REPLACE	
4	EXISTING	70287	YellowfinID

ImportOption

--	--

YFXMLContentResource

LoginId	String	Yellowfin WebIDIDID Yellowfin Yellowfin Web
Password	String	
OrgId	Integer	YellowfinID1
Function	String	Web GETIMPORTCONTENT
OrgRef	String	ID
Parameter	String[]	StringsStringUTF-8 String YFXML

StatusCode	String	Web <ul style="list-style-type: none">• SUCCESS• FAILURE
ContenResources	ContentResource []	Yellowfin

Java

- web

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

-

```
Path path = Paths.get("/Applications/Yellowfin 7.4/qwerty.yfx");

byte[] data = Files.readAllBytes(path);
byte[] encodeBase64 = java.util.Base64.getEncoder().encode(data);
String f = new String(encodeBase64, "UTF-8");
```

- YFXXML

```
rsr.setParameters(new String[] {f, "YFX"});
```

-

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

web

- StatusCodeContentResource

web

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


```

<%
/*      ws_getimportcontent.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.*" %>
<%@ page import="java.nio.file.Files" %>
<%@ page import="java.nio.file.Paths" %>
<%@ page import="java.nio.file.Path" %>

<%
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 above account
rsr.setOrgId(1);
rsr.setFunction("GETIMPORTCONTENT");

Path path = Paths.get("/Applications/Yellowfin 7.4/qwerty.yfx");      // existing file

byte[] data = Files.readAllBytes(path);
byte[] encodeBase64 = java.util.Base64.getEncoder().encode(data);
String f = new String(encodeBase64, "UTF-8");

rsr.setParameters(new String[]{f,"YFX"});

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

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

```

XMLYFXYellowfin

LoginId	String	Yellowfin WebIDIDID Yellowfin Yellowfin Web
Password	String	
OrgId	Integer	YellowfinID1

Function	String	Web IMPORTCONTENT
OrgRef	String	ID
Parameters	String[]	StringsStringUTF-8 String YFXXML
ImportOptions	ImportOption []	Yellowfin

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>
      <!--Optional:-->
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>IMPORTCONTENT</function>
        <parameters>PD94bWwgdmVyc2lvcj0iMS4wIiBlbmNvZGluZz0i ... </parameters>
        <parameters>XML</parameters>
        <importOption>
          <optionIndex>0</optionIndex>
            <optionKey>OPTION</optionKey>
            <optionValue>REPLACE</optionValue>
          <optionIndex>1</optionIndex>
            <optionKey>EXISTING</optionKey>
            <optionValue>70279</optionValue>
        </importOption>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

StatusCode	String	Web <ul style="list-style-type: none">SUCCESSFAILURE

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>3c25c8a81c971e26bd23d4ed60194fba</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>

```

Java

- web

```

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

```

-

```

Path path = Paths.get("/Applications/Yellowfin 7.4/YFexport.xml");

byte[] data = Files.readAllBytes(path);
byte[] encodeBase64 = java.util.Base64.getEncoder().encode(data);
String f = new String(encodeBase64, "UTF-8");

```

- YFXXML

```

rsr.setParameters(new String[]{f,"XML"});

```

-

```

ImportOption[] options = new ImportOption[2];
options[0] = new ImportOption();
options[0].setItemIndex(0);
options[0].setOptionKey("OPTION");
options[0].setOptionValue("REPLACE");

options[1] = new ImportOption();
options[1].setItemIndex(0);
options[1].setOptionKey("EXISTING");
options[1].setOptionValue("70279");

```

-

```
rsr.setImportOptions(options);
```

-

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

web

- StatusCode

web

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

```

<%
/*      ws_importcontent.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.*" %>
<%@ page import="java.nio.file.Files" %>
<%@ page import="java.nio.file.Paths" %>
<%@ page import="java.nio.file.Path" %>
<%
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 above account
rsr.setOrgId(1);
rsr.setFunction("IMPORTCONTENT");

/*      Yfexport.xml contains just one report with no dependencies
      which can be retrieved using an EXPORTCONTENT web service call and passing a single report id.
      FYI. Latest Yellowfin builds do not allow export Yellowfin content WITHOUT dependencies,
      so ImportOption in this example will not suit any YFX file.
      You need to define proper ImportOption anyway
*/
Path path = Paths.get("/Applications/Yellowfin 7.4/YFexport.xml");

byte[] data = Files.readAllBytes(path);
byte[] encodeBase64 = java.util.Base64.getEncoder().encode(data);
String f = new String(encodeBase64, "UTF-8");

rsr.setParameters(new String[]{f,"XML"});

      ImportOption[] options = new ImportOption[2];
      options[0] = new ImportOption();
      options[0].setItemIndex(0);
      options[0].setOptionKey("OPTION");
      options[0].setOptionValue("REPLACE");

      options[1] = new ImportOption();
      options[1].setItemIndex(0);
      options[1].setOptionKey("EXISTING");

      /*      existing report Id. Can be retrieved from ReportHeader table of Yellowfin database,
ReportId column.

      keep in mind that the reportId changes each time when a user modifies the report.
      You can use the GETIDFORUUID call to get the valid reportId value for the report.
      */
      options[1].setOptionValue("70279");

rsr.setImportOptions(options);

AdministrationServiceResponse rs1 = adminService.remoteAdministrationCall(rsr);

      if ("SUCCESS".equals(rs1.getStatusCode()) ) {
          out.write("<br>Test Import Success");
      }
      else {
          out.write("Failure");
          out.write(" Code: " + rs1.getErrorCode());
      }

%>

```

YFXXML

LoginId	String	Yellowfin WebIDIDID YellowfinYellowfin Web
Password	String	
OrgId	Integer	YellowfinID1
Function	String	WebTESTIMPORTCONTENT
OrgRef	String	ID
Parameter	String[]	StringsStringUTF-8 StringYFXXML
ImportOptions	ImportOption[]	

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>
      <!--Optional:-->
      <arg0>
        <loginId>admin@yellowfin.com.au</loginId>
        <password>test</password>
        <orgId>1</orgId>
        <function>TESTIMPORTCONTENT</function>
        <parameters>PD94bWwgdmVyc2lvcj0iMS4wIiBlbmNvZGluZz0i ... </parameters>
        <parameters>XML</parameters>
        <importOption>
          <optionIndex>0</optionIndex>
          <optionKey>OPTION</optionKey>
          <optionValue>REPLACE</optionValue>
          <optionIndex>1</optionIndex>
          <optionKey>EXISTING</optionKey>
          <optionValue>70279</optionValue>
        </importOption>
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

--	--	--

StatusCode	String	Web <ul style="list-style-type: none"> • SUCCESS • FAILURE
ImportIssues	ImportIssue[]	
ContentResources	ContentResource[]	Yellowfin

SAOP

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <contentResources>
          <resourceDescription>months, 6/3/2018 8:53 AM</resourceDescription>
          <resourceId>70279</resourceId>
          <resourceName>My Report 1000</resourceName>
          <resourceType>REPORT</resourceType>
          <resourceUUID>fd3794b3-62c0-4cf8-bac0-755e68d9c41e</resourceUUID>
        </contentResources>
        <errorCode>0</errorCode>
        <importIssues>
          <issueElements>
            <messageKey>error.reports.import.view</messageKey>
            <renderedMessage>View has not been selected.</renderedMessage>
          </issueElements>
          <issueElements>
            <messageKey>error.reports.import.category</messageKey>
            <renderedMessage>Folder has not been selected.</renderedMessage>
          </issueElements>
        </importIssues>
        <resource>
          <resourceDescription>months, 6/3/2018 8:53 AM</resourceDescription>
          <resourceId>70279</resourceId>
          <resourceName>My Report 1000</resourceName>
          <resourceType>REPORT</resourceType>
          <resourceUUID>fd3794b3-62c0-4cf8-bac0-755e68d9c41e</resourceUUID>
        </resource>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Java

- web

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

-

```
Path path = Paths.get("/Applications/Yellowfin 7.4/qwerty.yfx");

byte[] data = Files.readAllBytes(path);
byte[] encodeBase64 = java.util.Base64.getEncoder().encode(data);
String f = new String(encodeBase64, "UTF-8");
```

- YFXXML

```
rsr.setParameters(new String[]{f,"YFX"});
```

-

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

web

-

```
if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");

    ContentResource[] crs = rs.getContentResources();

    ImportIssue[] ImportIssues = rs.getImportIssues();
    out.write("<br>Import Issues: " + (ImportIssues!=null?ImportIssues.length:"no issues"));

    out.write("<table>");
    out.write("<tr><td> ResourceId </td><td> ResourceType </td><td> ResourceName </td><td>
ResourceUUID </td></tr>");
    for (ContentResource c: crs) {
        out.write("<tr>");
        out.write("<td>" + c.getResourceId() + "</td><td>" + c.getResourceType() + "<
/td><td>" + c.getResourceName() + "</td><td>" + c.getResourceUUID() + "</td>");
        out.write("</tr>");
    }
} else {
    out.write("Failure");
    out.write(" Code: " + rs.getErrorCode());
}
```

- StatusCodeImportIssuesContentResources

web

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


```

<%
/*      ws_testimportcontent.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.*" %>
<%@ page import="java.nio.file.Files" %>
<%@ page import="java.nio.file.Paths" %>
<%@ page import="java.nio.file.Path" %>
<%
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 above account
rsr.setOrgId(1);
rsr.setFunction("TESTIMPORTCONTENT");

Path path = Paths.get("/Applications/Yellowfin 7.4/www.yfx");          // existing file

byte[] data = Files.readAllBytes(path);
byte[] encodeBase64 = java.util.Base64.getEncoder().encode(data);
String f = new String(encodeBase64, "UTF-8");

rsr.setParameters(new String[]{f,"YFX"});

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

if ("SUCCESS".equals(rs.getStatusCode()) ) {
    out.write("<br>Success");

    ContentResource[] crs = rs.getContentResources();

    ImportIssue[] ImportIssues = rs.getImportIssues();
    out.write("<br>Import Issues: " + (ImportIssues!=null?ImportIssues.length:"no issues"));

    out.write("<table>");
    out.write("<tr><td> ResourceId </td><td> ResourceType </td><td> ResourceName </td><td> ResourceUUID
</td></tr>");
    for (ContentResource c: crs) {
        out.write("<tr>");
        out.write("<td>" + c.getResourceId() + "</td><td>" + c.getResourceType() + "</td><td>"
+ c.getResourceName() + "</td><td>" + c.getResourceUUID() + "</td>");
        out.write("</tr>");
    }
} else {
    out.write("Failure");
    out.write(" Code: " + rs.getErrorCode());
}
%>

```

web[TESTIMPORTCONTENT](#)

Yellowfinweb

Yellowfin

webCSV

LoginId	String	Yellowfin WebIDIDID Yellowfin Yellowfin Web
Password	String	
OrgId	Integer	YellowfinID1
Function	String	Web EXPORTTRANSLATIONALL

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

StatusCode	String	Web <ul style="list-style-type: none"> • SUCCESS • FAILURE
ContentType	String	
FileName	String	
BinaryData	String	Base64String

SAOP

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:remoteAdministrationCallResponse xmlns:ns2="http://webservices.web.mi.hof.com/">
      <return>
        <binaryData>77u/VVVJRCxUZZh0IFR5cGUSS2V5LE9yaWdpbmFsIFRleHQs ... </binaryData>
        <contentType>text/comma-separated-values</contentType>
        <errorCode>0</errorCode>
        <fileName>Translations - 10 Mar 2018.csv</fileName>
        <messages>Successfully Authenticated User: admin@yellowfin.com.au</messages>
        <messages>Web Service Request Complete</messages>
        <sessionId>4a19aa468b23ab18d3aee5c7121bcacd</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Java

- web

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

-

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

web

- StatusCodeBinaryDataFileNameContentType

web

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

```

<%
/*      ws_exporttranslationall.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.*" %>
<%@ page import="java.nio.file.Files" %>
<%@ page import="java.io.PrintWriter" %>
<%
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 above account
rsr.setOrgId(1);
rsr.setFunction("EXPORTTRANSLATIONALL");

AdministrationServiceResponse rs = adminService.remoteAdministrationCall(rsr);

        if ("SUCCESS".equals(rs.getStatusCode()) ) {
            out.write("<br>Success");

            //response.setBinaryData(Base64.encodeBytes(pdf.getData()));

            String Base64encoded = rs.getBinaryData();
            Base64encoded = Base64encoded.replace("\n", "").replace("\r", "");

            byte[] bytes = Base64encoded.getBytes();
            byte[] decoded = java.util.Base64.getDecoder().decode(bytes);

            String text = new String(decoded, "UTF-8");
            PrintWriter writer = new PrintWriter("/Applications/Yellowfin 7.4/" + rs.getFileName(),
"UTF-8");

            writer.println(text);
            writer.close();

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

```

YellowfinCSV

LoginId	String	Yellowfin WebIDIDID Yellowfin Yellowfin Web
Password	String	
OrgId	Integer	YellowfinID1
Function	String	Web IMPORTTRANSLATION
BinaryData	String[]	CSV

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>IMPORTTRANSLATION</function>
        <binaryData>-17</binaryData>
        <binaryData>-69</binaryData>
        <binaryData>-65</binaryData>
        <binaryData>85</binaryData>
        <binaryData>85</binaryData>
        ...
      </arg0>
    </web:remoteAdministrationCall>
  </soapenv:Body>
</soapenv:Envelope>
```

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

SAOP

```
<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>11c466874bfbcdb80f5d250c97ffbd03</sessionId>
        <statusCode>SUCCESS</statusCode>
      </return>
    </ns2:remoteAdministrationCallResponse>
  </S:Body>
</S:Envelope>
```

Java

- web

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

- CSV

```
Path path = Paths.get("/Applications/Yellowfin 7.4/Translations - 8 Mar 2018.csv"); // existing file  
byte[] data = Files.readAllBytes(path);  
  
rsr.setBinaryData(data);
```

-

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

web

- StatusCode

web

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

```

<%
/*      ws_importtranslation.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.*" %>
<%@ page import="java.nio.file.Files" %>
<%@ page import="java.nio.file.Paths" %>
<%@ page import="java.nio.file.Path" %>
<%

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 the password of the account above
rsr.setOrgId(1);
rsr.setFunction("IMPORTTRANSLATION");

Path path = Paths.get("/Applications/Yellowfin 7.4/Translations - 8 Mar 2018.csv");      // existing file

byte[] data = Files.readAllBytes(path);

rsr.setBinaryData(data);

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

```