

Dököll Solutions, Inc.

Application Development

Project Name:	Oracle Web Services	Start Date:	2016.10.31.10.40.AM
Purpose:	Web Services Handshake	End Date:	2016.10.31.10.40.AM
Software:	ADF, JSF, HTML5, Xpages, XML, Java, JavaAgent	Environment:	Oracle JDeveloper IBM Domino Designer
Employee Name:	Dököll Solutions	Employee ID:	Dököll Solutions
Task(s):	Exchange values between environments	Document:	Journal Entries

System Requirements

Software	Language	Protocol/Framework
Microsoft Windows 7, 8, 10	VBScript, Batch	Active Directory
Microsoft Internet Explorer	N/A	TCPIP, HTTP
Google Chrome	N/A	TCPIP, HTTP
Mozilla FireFox	N/A	TCPIP, HTTP
Oracle JDeveloper Version 12.xx IDE	ADF, JSF, CSS, HTML, JavaScript, Java	HTTP, TCPIP
Oracle Integrated WebLogic Server 12.x	WSDL	HTTP, TCPIP
IBM DB2 Express-C v9.7.4	SQL	SQL Database Server
IBM Notes Domino Designer 8.5.3, 9, 10	CSS, HTML, JavaScript, Java, JavaAgent, XML	HTTP, TCPIP, Xpages

Disclaimer:

Information contained in the following is presented as is. This tutorial assumes you have basic programming and software configuration knowledge. All tutorials are based on IBM Notes Domino or Oracle Fusion Middleware, including and not limited to items stated in System Requirements. Should you need to familiarize yourself with IBM Domino Designer or Oracle JDeveloper environments, prior to continuing, stop now and see our Journal Entries page: www.dokollolutionsinc.com for support...

Foreword:

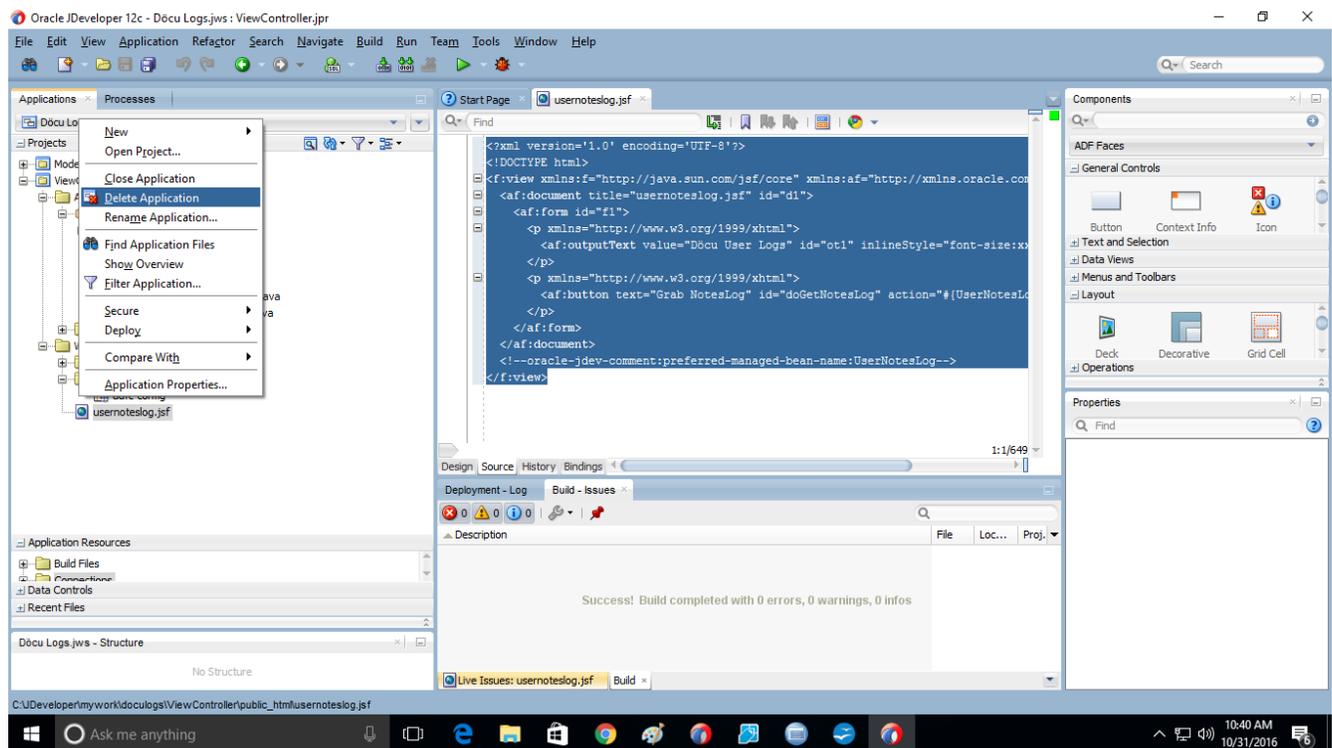
Samples included in this Journal Entries document are part of a series, be sure to consult other versions of the documentation to benefit in full.

Introduction:

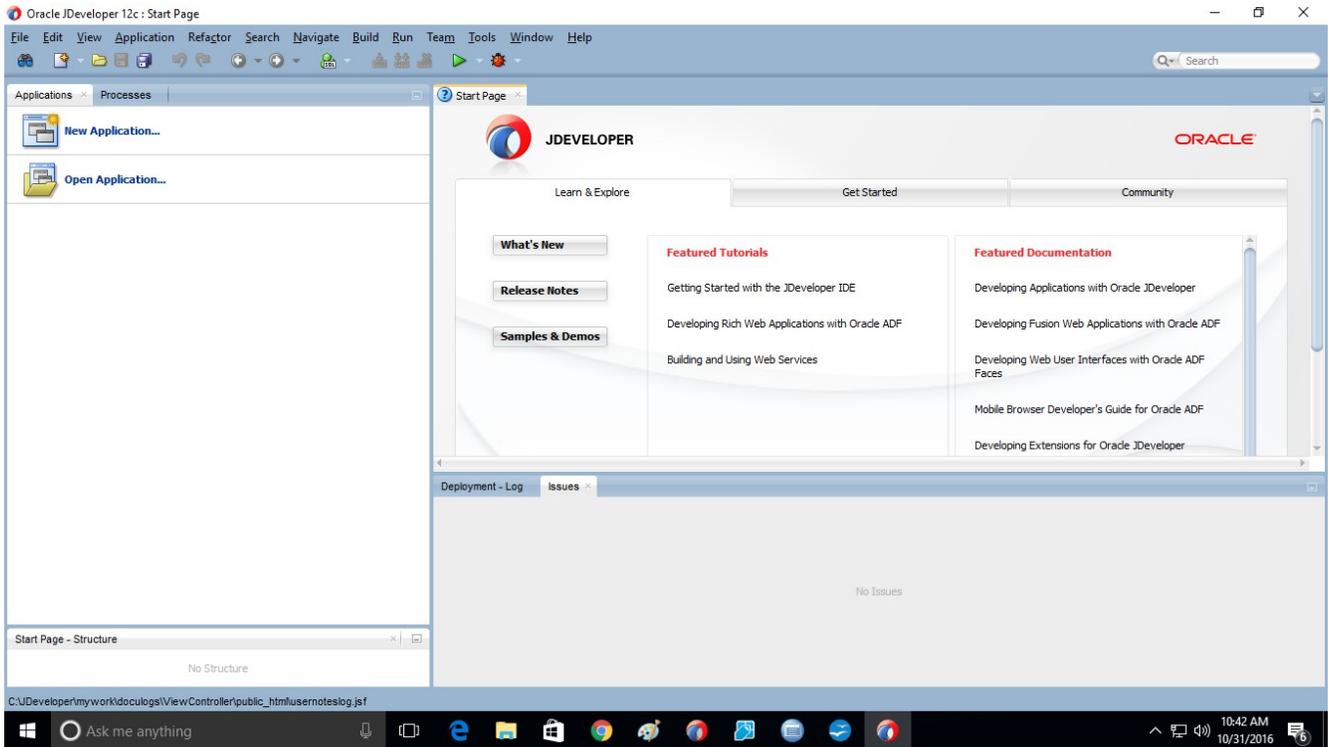
Delete existing Oracle JDeveloper Web Services project, start over from scratch, run and test programs using WebLogic Web Server, perform a handshake between external environments, mainly IBM Domino Designer. As usual, areas of interest will be highlighted where necessary.

+++++>>>
2016.10.31.10.40.AM

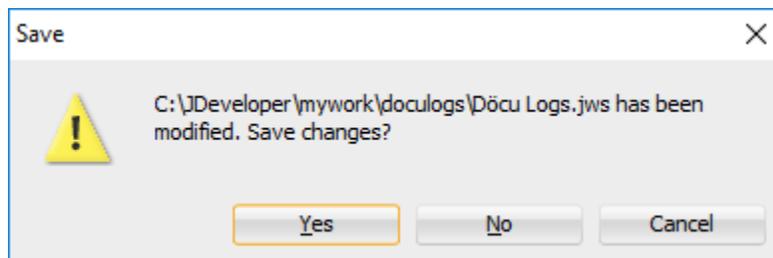
Okay we're ready to delete the App... starting over-



Closing the IDE and actually rebooting the machine...



that's weird, would have thought at this point in the shutdown process such a message would not pop up... choice is no in this case-



Hint: You should make sure to have copies of the work being deleted, some code may be salvageable...

Going ahead and also deleting code with affected areas in the external IBM Domino Designer development environment...

```
import lotus.domino.*;
```

```
/**
```

```
 * Copyright 2016 Dököll Solutions, Inc.
```

```
   Licensed under the Apache License, Version 2.0 (the "License");
```

you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

```
*
* @AppName: DöcuContent.nsf
* Program: HTTPOracleWebServiceRecordJavaAgent || OracleWebServiceJavaAgent
* jQuery Report data for Xpage, to be read by JavaBean,
* based on WebNews Web Service to be read from JDeveloper Apps
* Created: 2016.07.24.1.10.AM
*/

import java.io.PrintWriter;

import com.dokoll.solutions.inc.WebNews.WebSrv.model.UserDeptWebSrvBean;
import com.dokoll.solutions.inc.WebNews.WebSrv.model.UserDeptWebSrvBeanServiceLocator;

import lotus.domino.AgentBase;
import lotus.domino.Database;
import lotus.domino.Document;
import lotus.domino.Session;
import lotus.domino.View;
//Web Services imports
;
/**
 * @author Dököll Solutions, Inc.
 * @version 2016.07.24.1.10.AM
 *
 */
public class OracleWebServiceJavaAgent extends AgentBase {
    final String GotoPage =
        "http://localhost/doculog.nsf/";
    //button code...
    public void NotesMain() {
        try {
            // Session
            Session session = this.getSession();
            // Output using PrintWriter
            PrintWriter printWriter = this.getAgentOutput();
            // ...
            Database database = session.getDatabase("", "doculogs.nsf");
            System.out.println("database obtained..." + database);
            // ensure database is not null
            if (database != null) {
                // ...
                View uLogs = database.getView("UserWebServicesParamViewListings");
                System.out.println("View obtained..." + uLogs);
                if (database.isFTIndexed()) {
                    System.out.println("database is indeed indexed for searching...");
                    // ensure view is not null
                    if (uLogs != null) {
```

```

// grab first doc
Document doc = uLogs.getFirstDocument();
//String/Int object(s) declaration to collect records
String strUserDept = null;
String strUserName = null;
String strUserDate = null;
int c = 0;
//prepare HTML page to build jQuery report
//TODO: Ensure code is up to snuff for actual Servers
printWriter.println("<html>");
printWriter.println("<head>");
printWriter.println("<title>User Department - jQuery Report</title>");
printWriter.println("<meta charset='utf-8'>");
printWriter.println("<meta name='viewport' content='width=device-width,
initial-scale=1'> ");
printWriter.println("<link rel='stylesheet'
href='/doculogs.nsf/mobile/app/tree/css/themes/default/jquery.mobile-1.2.0.css' />
");
printWriter.println("<link rel='stylesheet'
href='/doculogs.nsf/mobile/app/tree/docs/_assets/css/jqm-docs.css'/>");
printWriter.println("<script
src='/doculogs.nsf/mobile/app/tree/js/jquery.js'></script>");
printWriter.println("<script
src='/doculogs.nsf/mobile/app/tree/docs/_assets/js/jqm-docs.js'></script>");
printWriter.println("<script
src='/doculogs.nsf/mobile/app/tree/js/jquery.mobile-1.2.0.js'></script>");
printWriter.println("</head>");
printWriter.println("<body>");

printWriter.println("<div align='left'>");
printWriter.println("<div data-role='content' align='left'>");
printWriter.println("<div class='content-secondary'>");
printWriter.println("<ul data-role='listview' data-theme='d' data-
divider-theme='d'>");
printWriter.println("<li data-theme='a'>User Department - jQuery
Report</li>");
//Scan view for all documents
while (doc != null) {
c += 1;
// Retrieve contents spit out HTML
// TODO: further Mix and Play with Xpages jQuery, add more
functionality to sites
strUserName = doc.getItemValueString("subjectUserName");
strUserDept = doc
.getItemValueString("subjectDeptID");
strUserDate = doc.getItemValueString("subjectDateCreated");
printWriter.println("<table border='1'><tr>");
printWriter.println("<li data-role='list-divider'>" + strUserName +
"<span class='ui-li-count'>" + " " + strUserDept+ "</span></li>"
+ "<li><a href="+GotoPage + uLogs + "/" + doc.getUniversalID() + "?
OpenDocument" + " target='_parent'><h3>" + strUserName + "</h3><p><strong>" +
strUserDept
+ "</strong></p><p><a href="+GotoPage + uLogs + "/" +
doc.getUniversalID() + "?OpenDocument" + " target='_parent'>" + doc.getUniversalID()
+ "</a></p><p class='ui-li-aside'><strong>"
+ c + "</strong></p></a></li></ul></div></div></div>");

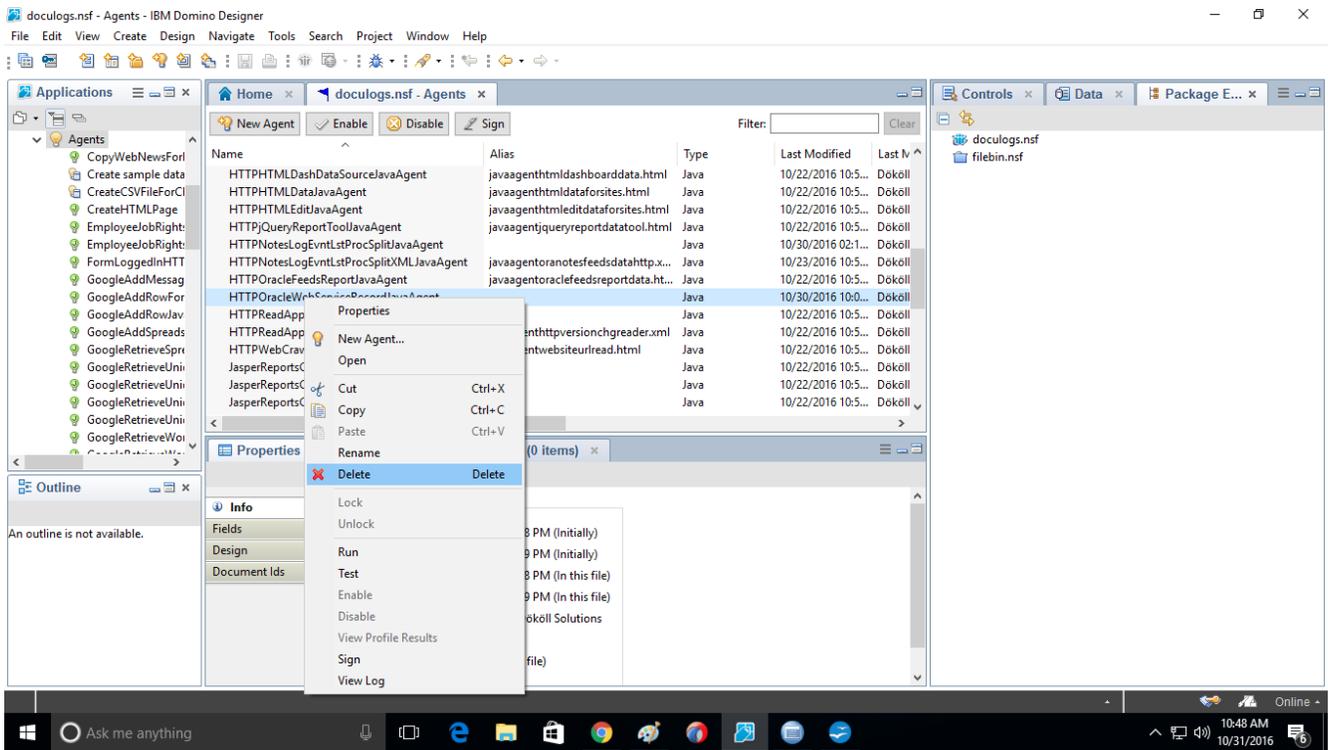
UserDeptWebSrvBean stub = new
UserDeptWebSrvBeanServiceLocator().getUserDeptWebSrvBeanPort();

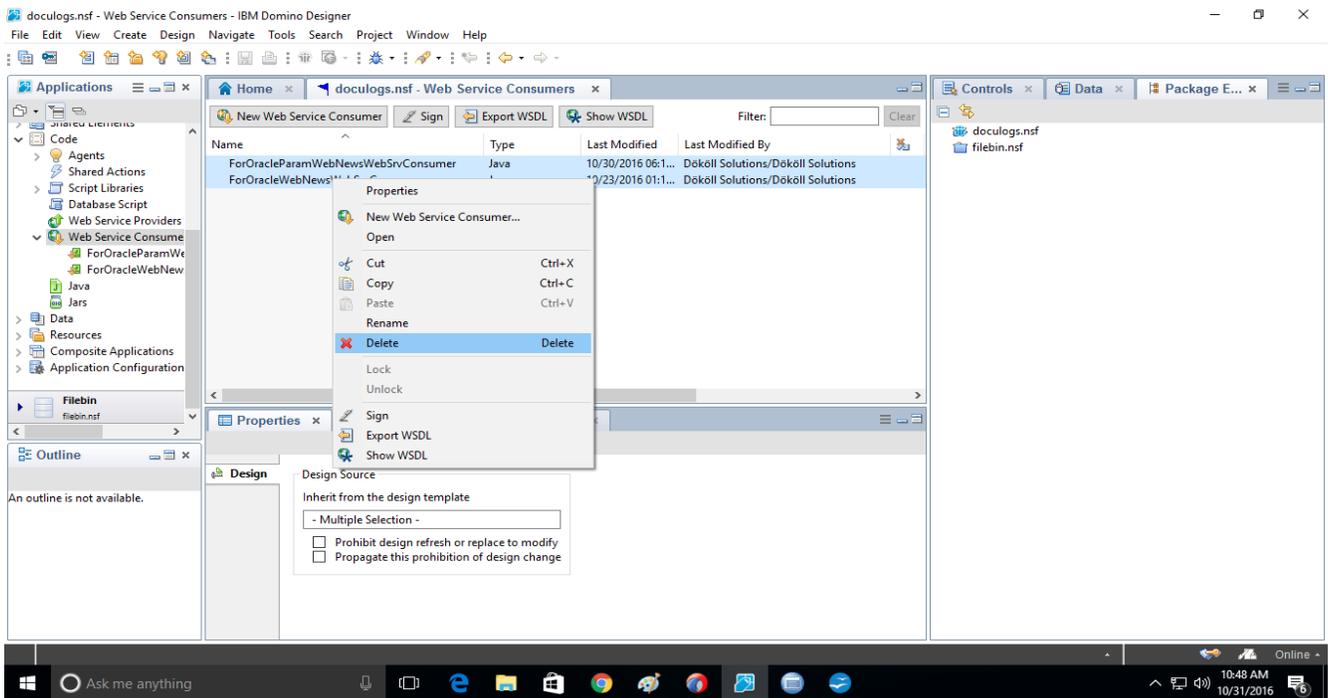
```

```

        //do something with the result of your search
        //example submit and return value via submitEntry method from Web
Service Provider file in Oracle JDeveloper
        //2016.07.24.1.10.AM
        String BrowserValue = strUserData.toString();
        //Submit view item to Web Service variable in JDeveloper
        System.out.println("User Department: " +
stub.submitEntry(BrowserValue));
        doc = uLogs.getNextDocument(doc);
    }
} else {
    System.out.println("database not indexed for searching...");
}
//recycle for memory...
uLogs.recycle();
}
// ...
database.recycle();
}
// ...
session.recycle();
} catch (Exception e) {
    e.printStackTrace();
}
}
}
}

```

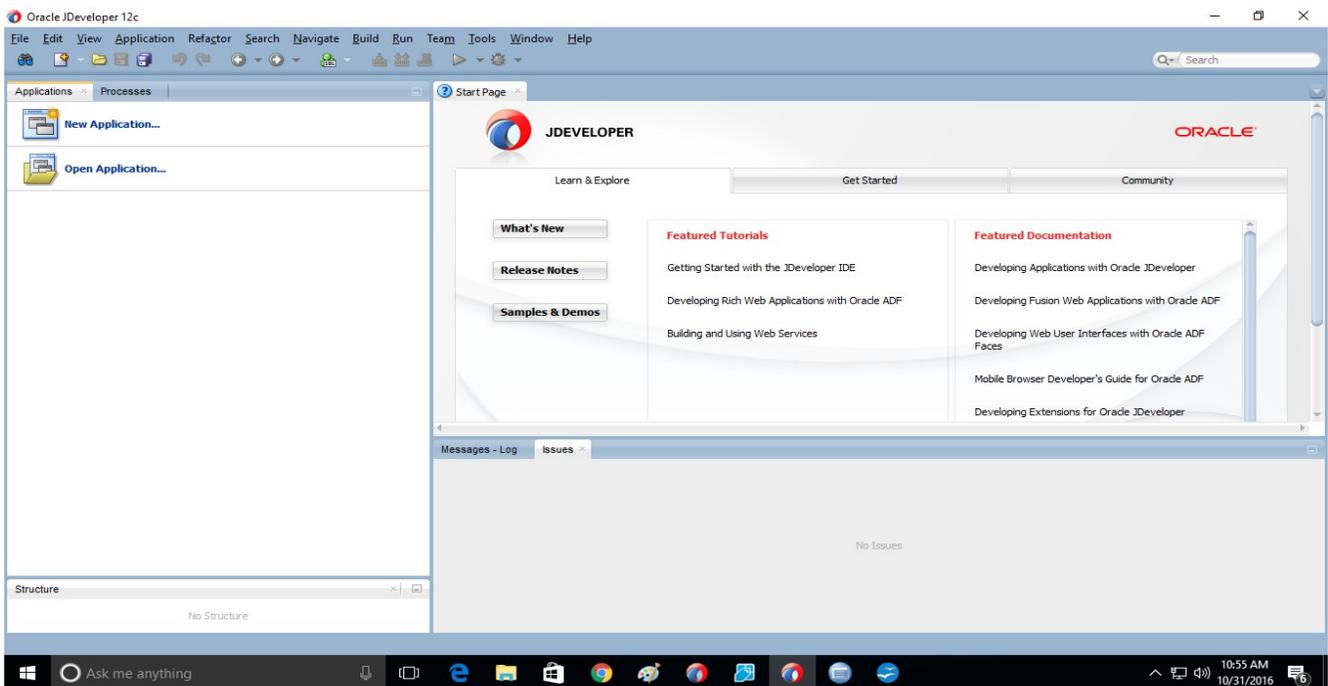




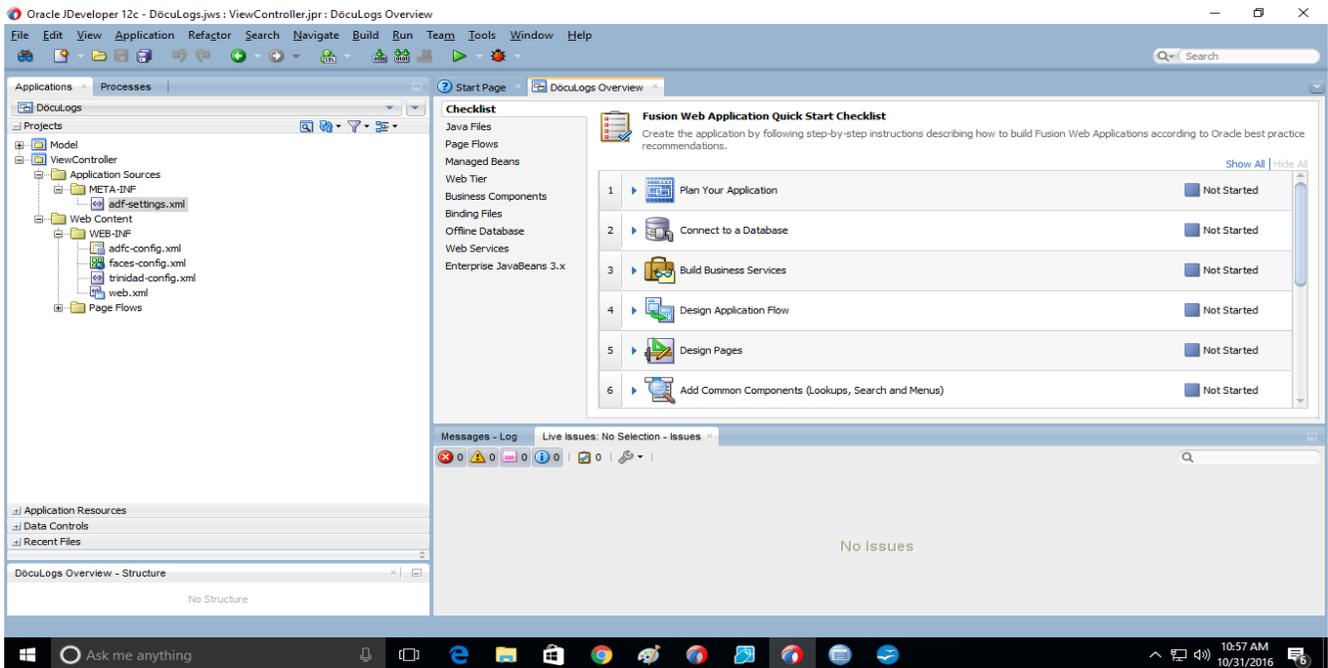
+++++>>>

2016.10.31.10.54.AM

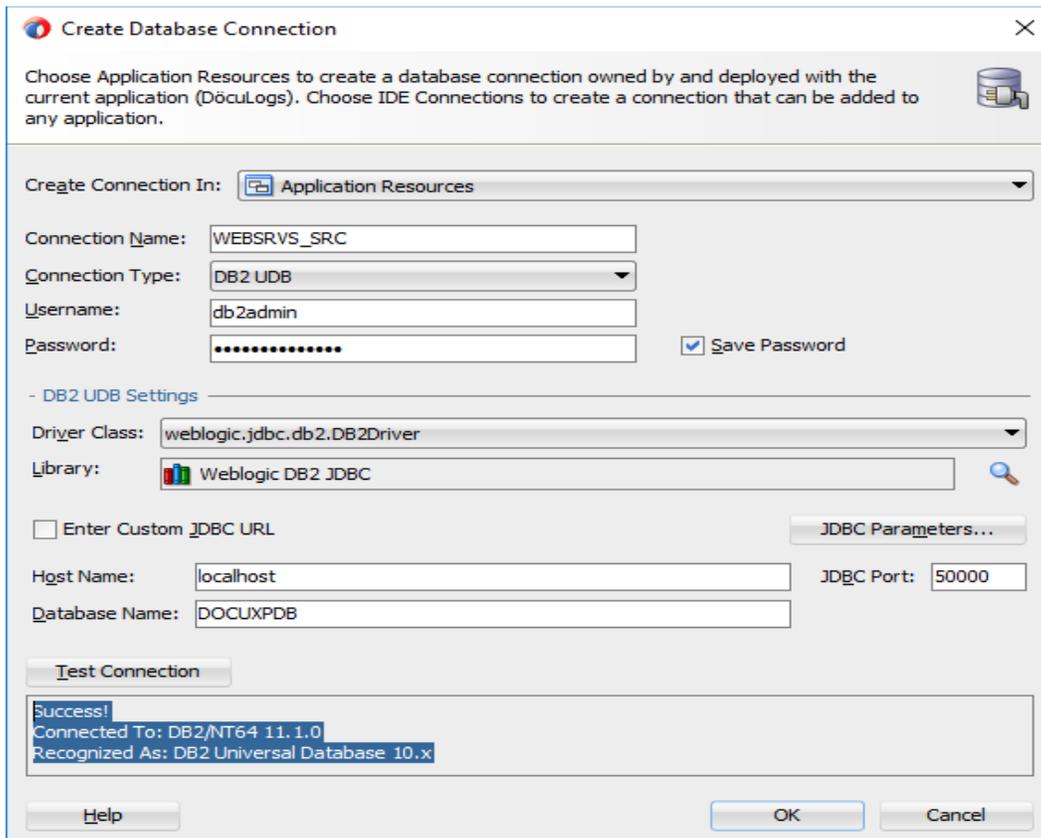
Back to Oracle JDeveloper, we are going to post very little screenshots in this particular Journal Entries document, in the interest of saving room for the complete file, and to limit redundancy in our information...



Will be creating a new App, and skipping the basic steps, will not be adding screenshots for those either- You can view prior files for support on each setup...



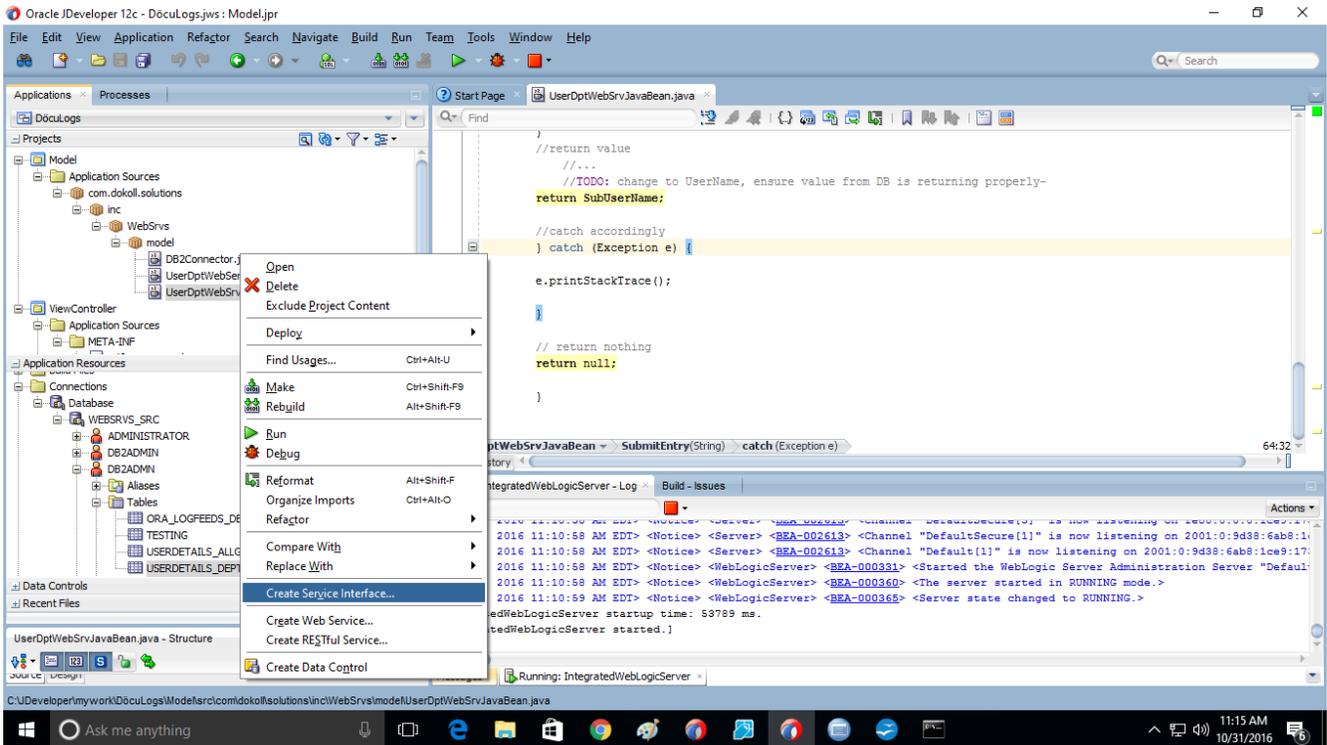
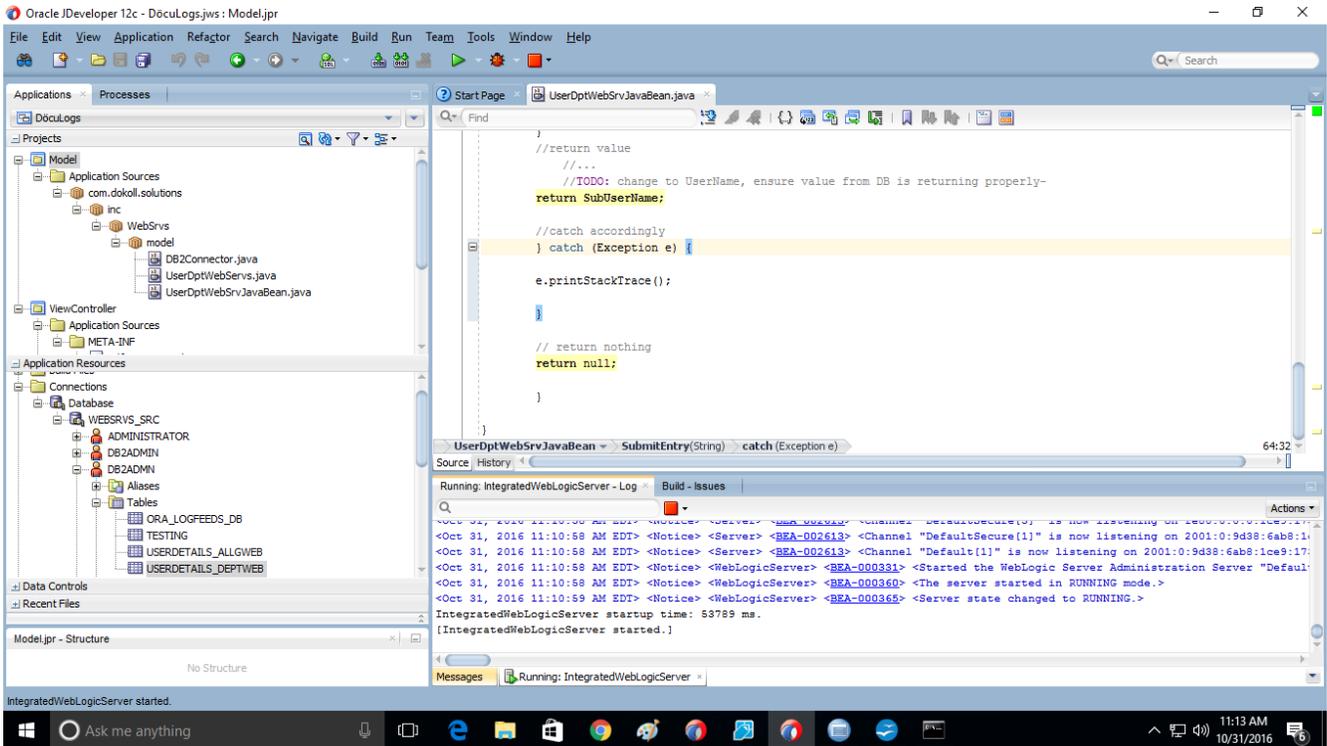
Will go ahead and create a Database connection-



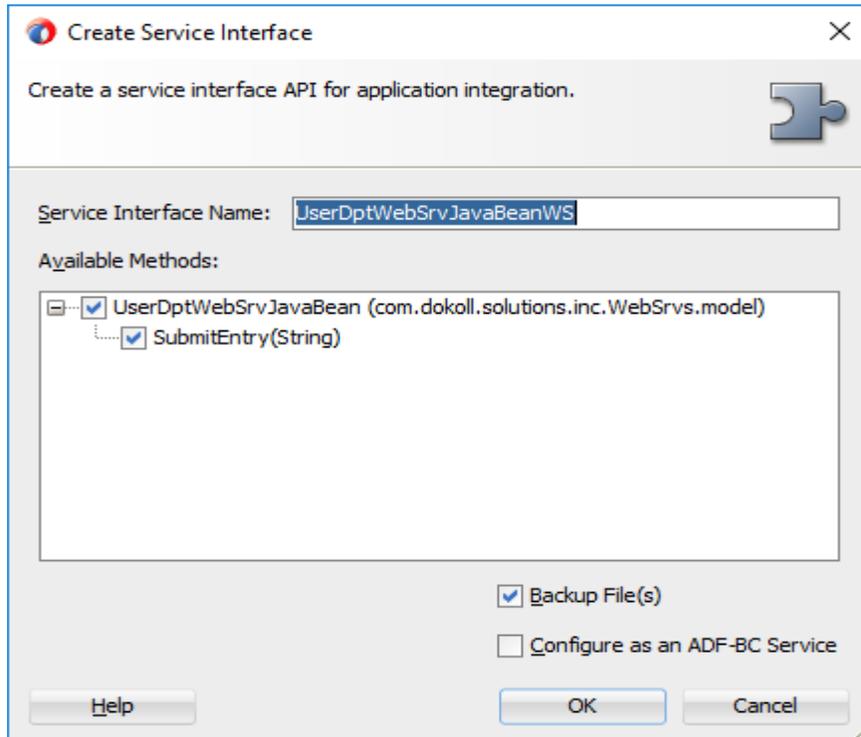
+++++>>>

2016.10.31.11.14.AM

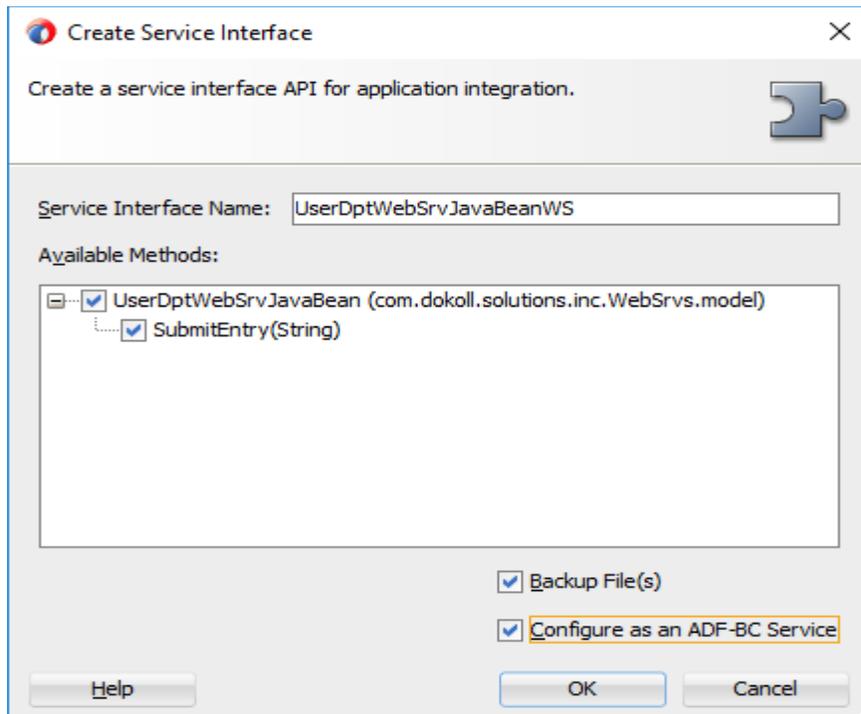
We are at a good point now to turn code into a Web Service, and we hope the test is successful-

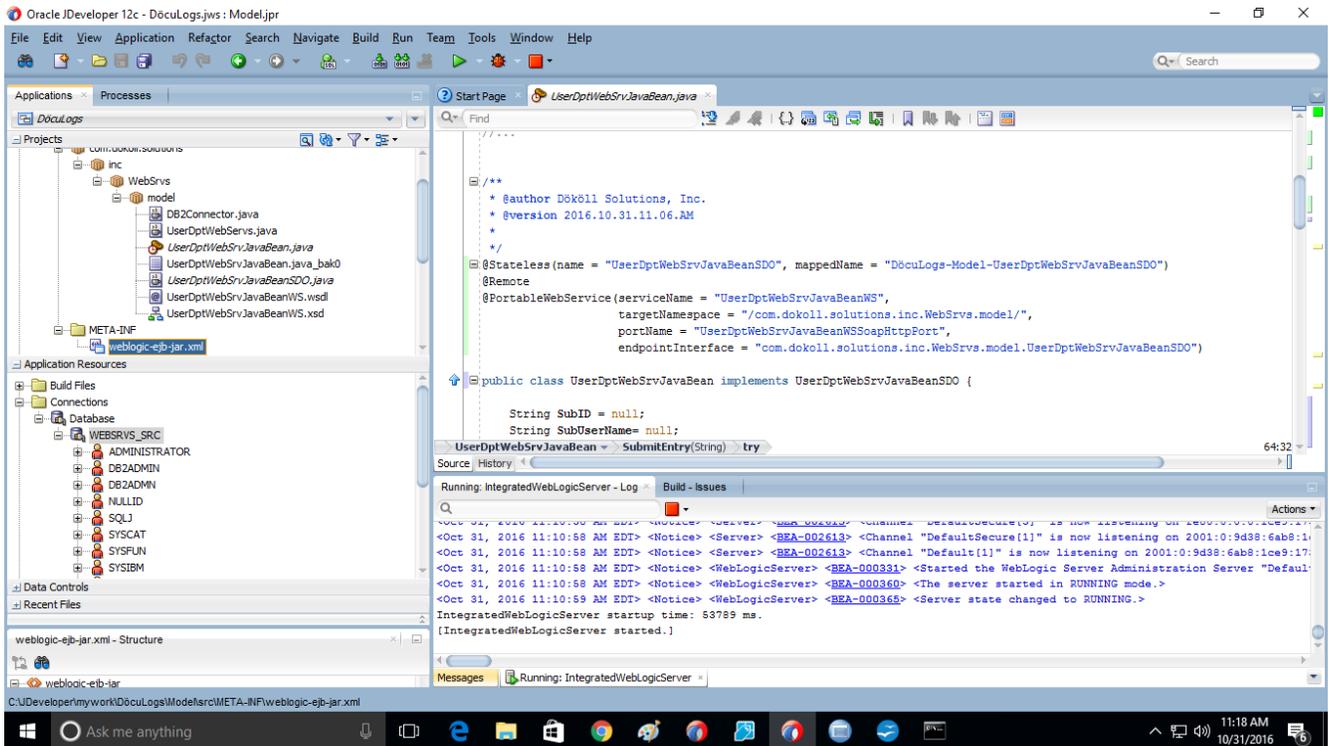


Changing course a bit, trying something for the first time here, just to show you an option to create an Interface using current code-

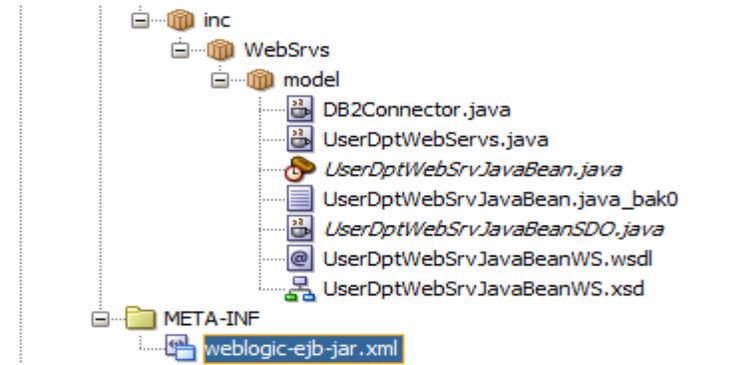


Checked as ADF BC Service below...





it puts up some technologies and attached them to the project....



```

/**
 * @author Dököll Solutions, Inc.
 * @version 2016.10.31.11.06.AM
 */
@Stateless(name = "UserDptWebSrvJavaBeanSDO", mappedName = "DöcuLogs-Model-UserDptWebSrvJavaBeanSDO")
@Remote
@PortableWebService(serviceName = "UserDptWebSrvJavaBeanWS",
  targetNamespace = "/com.dokoll.solutions.inc.WebSrvs.model/",
  portName = "UserDptWebSrvJavaBeanWSSoapHttpPort",
  endpointInterface = "com.dokoll.solutions.inc.WebSrvs.model.UserDptWebSrvJavaBeanSDO")

public class UserDptWebSrvJavaBean implements UserDptWebSrvJavaBeanSDO {

```

Back to our simpler Oracle JDeveloper method, the one that we know works:-)

```
//...
//java imports
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;
//...

/**
 * @author Dököll Solutions, Inc.
 * @version 2016.10.31.11.06.AM
 *
 */

public class UserDptWebSrvs {

    String SubID = null;
    String SubUserName= null;

    public UserDptWebSrvs() {
        super();
    }

    public String SubmitEntry (String inputDeptNo){

        //...
        try {

            //...
            Connection connection = DB2Connector.getConnection();

            Statement stat =
connection.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,
                ResultSet.CONCUR_READ_ONLY);

            connection.setAutoCommit(true);

            ResultSet rs = stat.executeQuery("select * from DB2ADMN.USERDETAILS_DEPTWEB
where DEPTID = '" + inputDeptNo + "'");

            //declare and initialize counts
            int rowcount = 0;
            if (rs.last()) {
                rowcount = rs.getRow();
                rs.beforeFirst();
            }
            //
            while (rs.next()) {
                //
                SubID = rs.getString("DEPTID");
                SubUserName = rs.getString("USERNAME");
            }

            rs.close();
            connection.close();
        }
    }
}
```

```

    }
    //return value
    //...
    //TODO: change to UserName, ensure value from DB is returning properly-
    return SubUserName;

    //catch accordingly
    } catch (Exception e) {

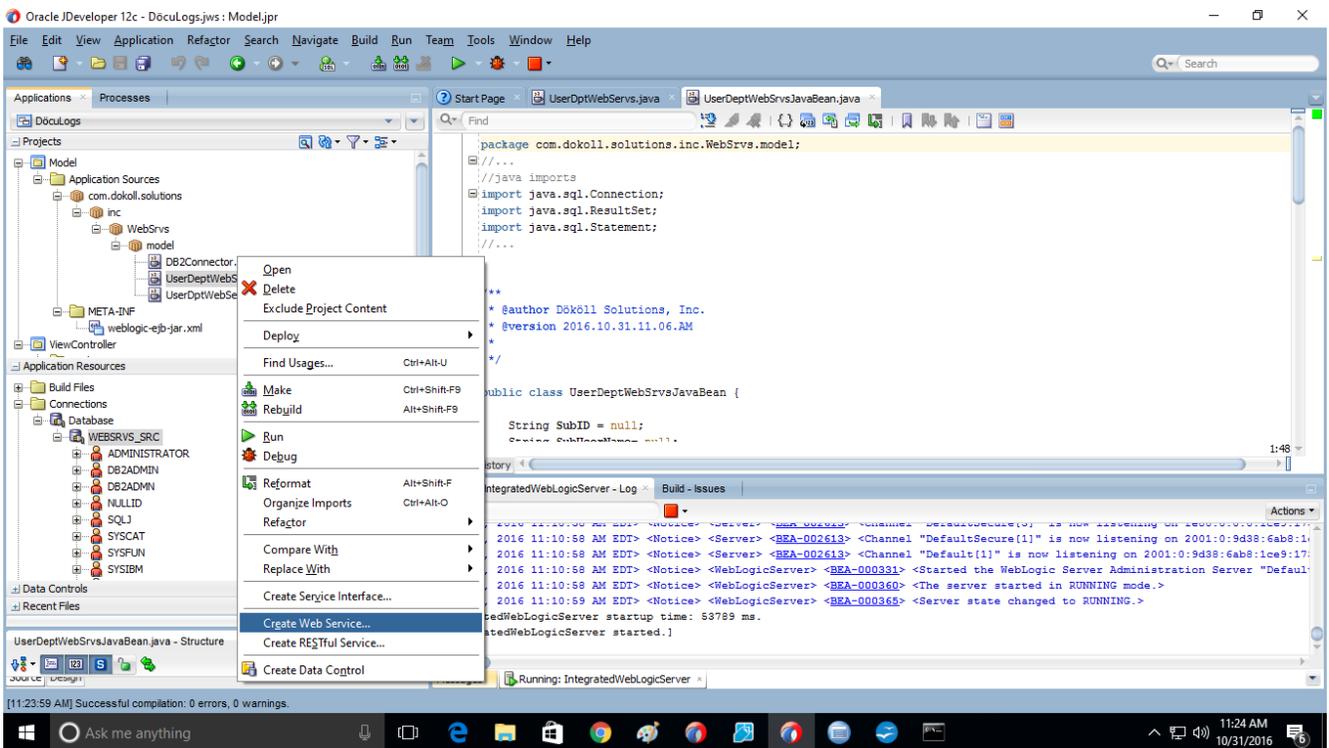
    e.printStackTrace();

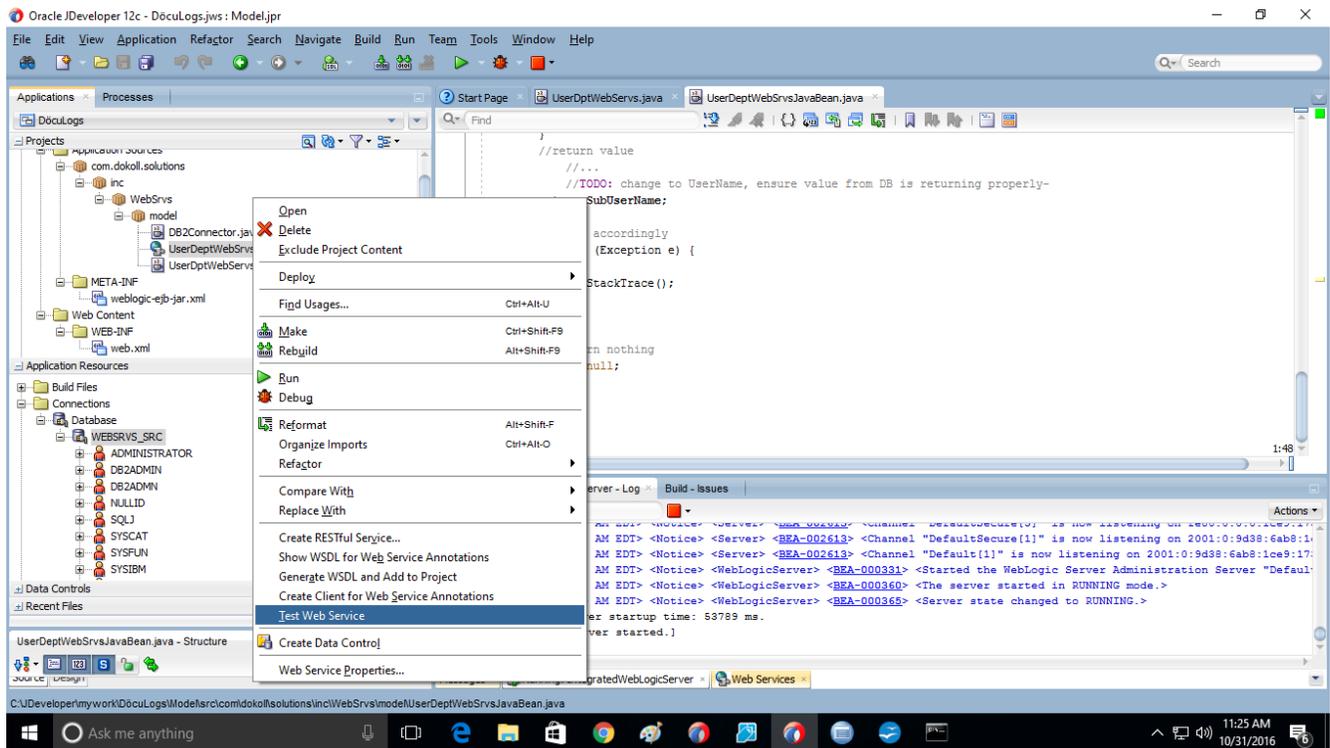
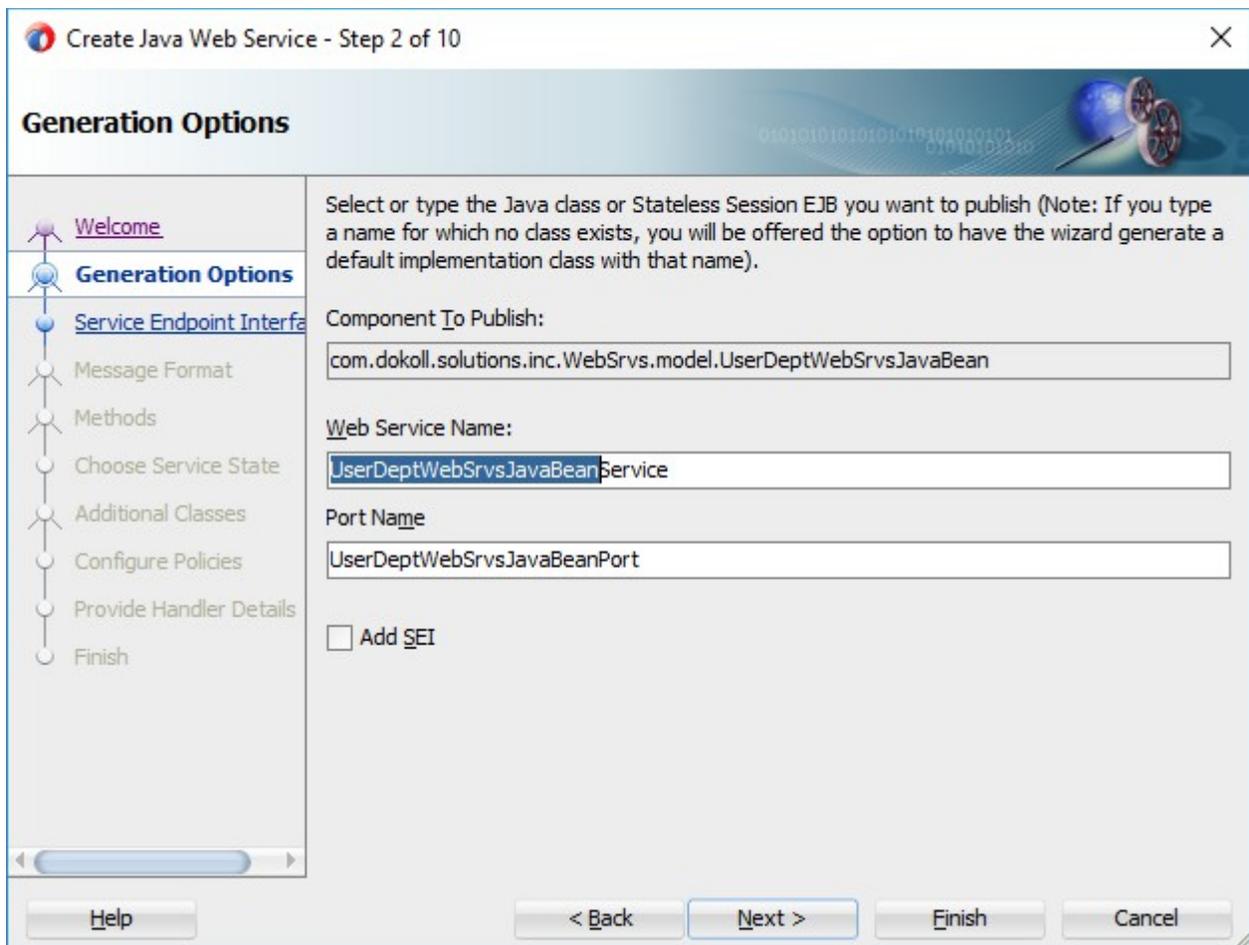
    }

    // return nothing
    return null;

    }
}

```





So far so good, we're banking on being able to connect this Web Service to IBM Domino and produce a successful handshake.

See Next Journal Entries document...

version: 2016.10.31.10.40.AM