

Processes Online User Guide

Web Application Development Tutorial

1 CREATION OF THE REGISTERDISTRIBUTOR PROJECT IN ECLIPSE IDE

The Eclipse IDE can be downloaded from the [Eclipse website](#). The Eclipse IDE requires Java to be installed on the authorized application developer's (in our current example, Sarah Connor's) machine. Java can be downloaded from the [Oracle website](#). We will need to install Tomcat and link it with Eclipse. A tutorial on how to do this can be found at [coreservlets.com](#).

Start the Eclipse IDE. Click **File > New > Dynamic Web Project**. In the form that presents itself, enter the project name as **RegisterDistributor**. Click **Finish**.

Processes Online DBaaS APIs are found in the Java class, Bizproapi, packaged in Enflow.jar. You will find the package and the Javadoc for this class, within the folder, **Processes Online_DBaaS_Kit**.

Now, expand the RegisterDistributor project in the Eclipse Project Explorer, to find the **lib** folder within the WebContent/WEB-INF folder. Now drop Enflow.jar, which is found in the folder, jar, in the folder, **Processes Online_DBaaS_Kit**, into the lib folder. Now right click Enflow.jar, and click **Build Path > Add To Build Path**.

2 ENTERING SARAH CONNOR'S CREDENTIALS IN THE REGISTER DISTRIBUTOR PROJECT

Find a folder, com, in the folder, **Processes Online_DBaaS_Kit**. Next, expand the RegisterDistributor project in the Eclipse Project Explorer, to find the src folder within the Java Resources folder. Now drop com into the src folder.

The next step is to input Sarah Connor's credentials into the class, Cred.java, which is within the com.enterpriseflow.customapp package.

Expand Cred.java to find the following definition of a String[] called cred:-

```
private static final String[] cred = {"xxxxxx.appspot.com", "OrgID", "EmployeeID", "EncryptedAPIPassword"};
```

Figure 1

Input Sarah Connor's credentials therein:-

```
String[] cred = {"processesonline.appspot.com", "ISEBDH", "ADMIN1", "1cbfe860fb2ca67efaf58645f825c2e0a663c90a282c8128022f0cdaf76a8b1b"};
```

Figure 2

3 REGISTER DISTRIBUTOR APPLICATION

Register Distributor application supports the following 4 activities:-

- Adding a Distributor.
- Rating a Distributor (*, **, ..., or *****), i.e., from 1-star upto 5-star.

- Deleting a Distributor.
- Creating a list of existing Distributors.

Recollect also that Sarah Connor has already created a Distributor class, with the following metadata:-

CLASS: Distributor

Sl. No.	Attribute Name	Datatype
1	DistrID (Key)	String
2	DistrName	String
3	DistrAddress	String
4	RepName	String
5	RepEmail	Email

Figure 3

Sarah now creates a custom attribute, named DistributorRating, of datatype String, for the Distributor class.

4 REGISTER DISTRIBUTOR APPLICATION SOFTWARE

You will find the source code for the Register Distributor application within the sourcecode/RegisterDistributor folder in the folder, **Processes Online_DBaaS_Kit**. Drop this entire source code into the WebContent folder of the RegisterDistributor application in Eclipse.

Recollect that the Register Distributor application is required to be accessible by authorized employees of Premium Products Inc. Recollect that the employee, ADMIN2::Rohit Shetty, has been given access to this application, by way of setting the value of the custom attribute, RegDistrAuth, to “Yes” for this employee.

We will now run this application in localhost. To do this, first select the RegisterDistributor application, then click the **Run As** button:-

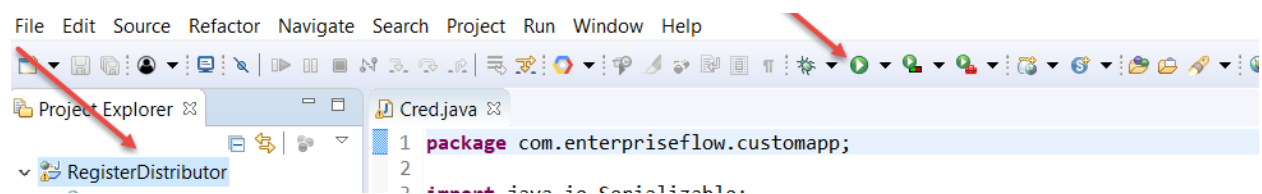


Figure 4

The following screen presents itself:-

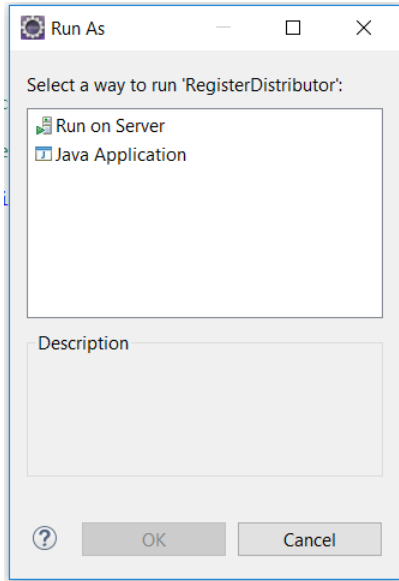


Figure 5

Select Run on Server, then click OK, and then in the screen that presents itself, click **Finish**. This will start the RegisterDistributor application in localhost.

Next, in the Chrome or other browser, visit <http://localhost:8080/RegisterDistributor/>. You will now see the following screen:-

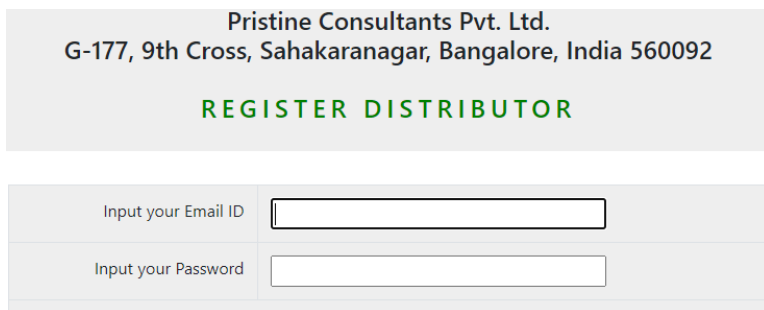


Figure 6

In order to understand how this screen got generated, expand the RegisterDistributor project in Eclipse. You will now see the following:-

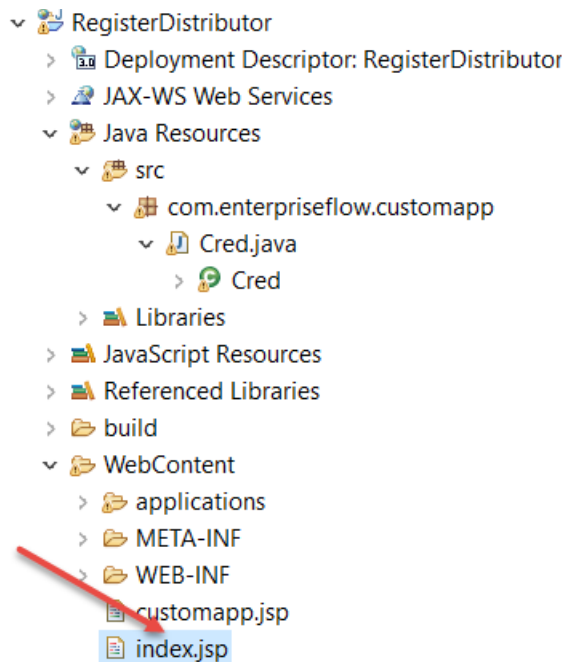


Figure 7

Click index.jsp, and you will find the page appear in the Eclipse editor:-

```

index.jsp
58= <body>
59= <div class="container">
60= <div class="panel panel-default">
61= <div style="text-align:center;background-color:#eeeeee" class="panel-body"><h4><%= Bizproapi.getOrgName() %><br><%= Bizproapi.getOrgAddress() %>
62 <br><br><span style="letter-spacing:5px;color:green;">REGISTER DISTRIBUTOR</span></h4></div>
63 </div>
64 </div>
65 <br>
66= <div class="container">
67= <div class="row">
68= <div class="col-sm-3">
69
70 </div>
71= <div class="col-sm-6">
72= <form action="customapp.jsp" method=post OnSubmit="return validate(this.Email.value, this.Password.value);">
73 <input type="hidden" name="EntityType" value="Employee">
74= <table style="background-color: #eeeeee;" class="table table-bordered">
75 <tr><td align="right">Input your Email ID</td><td><input type="text" size="40" name="Email" id="Email"></td></tr>
76 <tr><td align="right">Input your Password</td><td><input type="password" size="40" name="Password"></td></tr>
77 <tr><td colspan="2" align="center"><input type="submit" name="SUBMIT" value="SUBMIT"></td></tr>
78 </table>
79 </form>
80 </div>
81= <div class="col-sm-3">
82
83 </div>
84 </div>
85 </div>
--
  
```

Figure 8

Indeed, the company name, “Pristine Consultants Pvt. Ltd.”, and company address, “G-177, 9th Cross, Sahakaranagar, Bangalore, India 560092” in Figure 6 were generated respectively by the APIs, Bizproapi.getOrgName() and Bizproapi.getOrgAddress(), shown in Figure 8. The employee login form shown in Figure 6 was generated by the html form code shown in Figure 8.

Notice in Figure 8 that when an employee logs in, he / she will be taken to a screen generated by the file customapp.jsp (which you can see in Figure 7).

Next, let us have Rohit Shetty login. Rohit will see the following screen generated by the file customapp.jsp:-

Hello, Rohit Shetty !

What do you want to do now ?

Add Distributor

Rate Distributor

Delete Distributor

List Distributors

Figure 9

In order to understand how Rohit got there, let us have a look at the file customapp.jsp:-

```
<%
String entityType = request.getParameter("EntityType");
String email = request.getParameter("Email");
String password = request.getParameter("Password");
boolean auth = false;
if(Bizproapi.verifyPassword(entityType, email, password).equals("OK")){auth = true;}
if(auth)
{
    String myname = Bizproapi.getMyNameFromEmail(entityType, email);
    boolean caninit = Bizproapi.roleAuthorized(entityType, email, "RegDistrAuth");
}%>
<br>
<p style="text-align:center;"><i>Hello, <%= myname %> !</i><br><br>What do you want to do now ?<br><br></p>
<%
if((caninit))
{
}%>
<div class="container">
<div class="row">
<div class="col-sm-4">
</div>
<div class="col-sm-4">
<p style="background-color: #eeeeee;text-align:center;padding: 10px;border-radius: 10px;letter-spacing:3px;">
<a href="applications/AddDistributor/first.jsp" target="_blank">Add Distributor</a>
</p>
</div>
<div class="col-sm-4">
</div>
</div>
</div>
<div class="row">
<div class="col-sm-4">
</div>
<div class="col-sm-4">
<p style="background-color: #eeeeee;text-align:center;padding: 10px;border-radius: 10px;letter-spacing:3px;">
<a href="applications/RateDistributor/first.jsp" target="_blank">Rate Distributor</a>
</p>
</div>
<div class="col-sm-4">
</div>
</div>
```

```

</div>
</div>
<div class="row">
<div class="col-sm-4">
</div>
<div class="col-sm-4">
<p style="background-color: #eeeeee;text-align:center;padding: 10px;border-radius: 10px;letter-spacing:3px;">
<a href="applications/DeleteDistributor/first.jsp" target="_blank">Delete Distributor</a>
</p>
</div>
<div class="col-sm-4">
</div>
</div>
<div class="row">
<div class="col-sm-4">
</div>
<div class="col-sm-4">
<p style="background-color: #eeeeee;text-align:center;padding: 10px;border-radius: 10px;letter-spacing:3px;">
<a href="applications/ListDistributors/first.jsp" target="_blank">List Distributors</a>
</p>
</div>
<div class="col-sm-4">
</div>
</div>
</div>
</div>
</div>
<%
}
else
{
%>
<p style="text-align:center;"><br><br>You are not authorized to access this application</p>
<%
}
%>
<br>
<%
}
else
{
%>
<p style="text-align:center;"><br><br>Authentication failure</p>
<%
}
%>

```

Figure 10

Notice that Rohit’s password was verified by the API, Bizproapi.verifyPassword(). Thereafter, the API, Bizproapi.roleAuthorized() ensured that Rohit had been assigned the value “Yes” for the custom attribute, RegDistrAuth; this caused the boolean variable, caninit, to have the value, true; accordingly the screen shown in Figure 9 presented itself to Rohit.

You are now encouraged to see what happens when Rohit clicks the buttons, **Add Distributor**, **Rate Distributor**, **Delete Distributor**, and **List Distributors**, and understand which APIs in the files within the folder, applications, of the RegisterDistributor project (see Figure 7), caused these to happen.

5 CONCLUSION

We hope you have now mastered web application development in *Processes Online*. Now go ahead and develop great applications for your own organization !