

LOGO

PLATFORM TAILOR

1. STARTING TO USE	3
1.1. Purpose of Using Logo Platform Tailor	3
2. CREATING A PROJECT.....	3
2.1. Steps to creating a new project	3
2.2. Project Settings.....	5
2.3. Connection Settings.....	6
3. FOLDERS IN THE PROJECT AND HOW TO USE THEM	7
3.1. Tables	7
3.2. Objects.....	10
3.2.1. Creating a new object.....	10
3.2.2. Expanding an existing product object (Extension).....	12
3.3. Queries	15
3.4. Forms	18
3.4.1. Creating a new form.....	18
3.4.2. Customizing an existing product form	26
3.5. Reports	30
3.6. Trees	35
3.7. Profile	37
4. DEPLOYMENT.....	40
4.1. Deploying on Eclipse	40
4.2. Deploying in the product.....	41
4.3. Operations after Deploying	45

1. STARTING TO USE

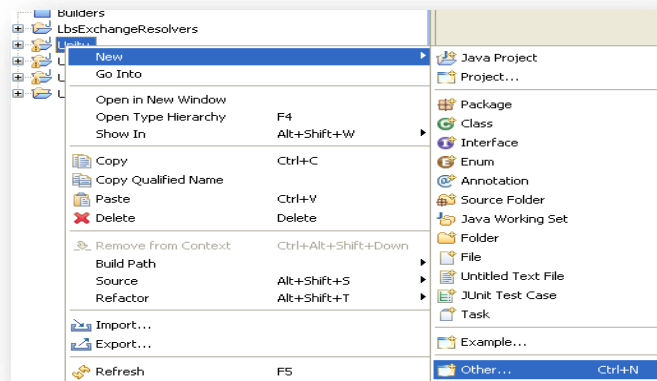
1.1. Purpose of Using Logo Platform Tailor

- You can customize tree structure that you use in j-guar, and you can add new modules.
- You can add new forms to j-guar interface and add new fields to the existing forms.
- You can add new queries.
- You can add a new report or expand existing reports.
- You can integrate other systems with j-guar. To do so, you can either use standard Web services or you can add new Web Services.
- You can add new methods that are specific to the project in use of Excellent.
- You can create new batch operations.

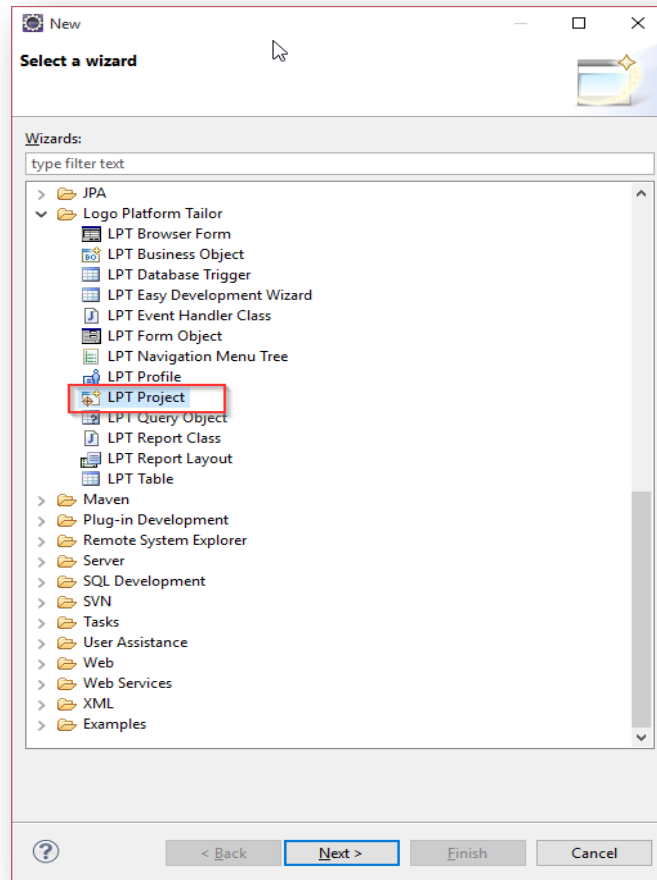
2. CREATING A PROJECT

2.1. Steps to creating a new project

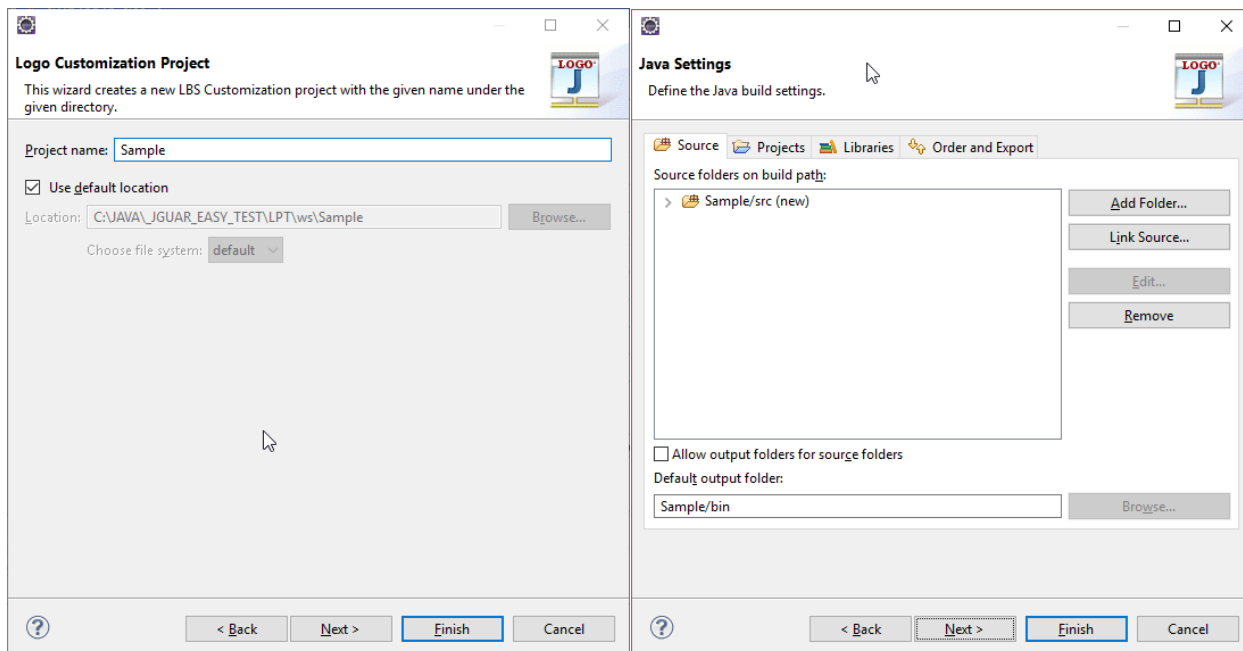
The first step to customization is to create a new customization project. You use “Logo Customization Project” wizard to create a project in LPT. Right click on Package Explorer, and then click **New > Other** on the list.

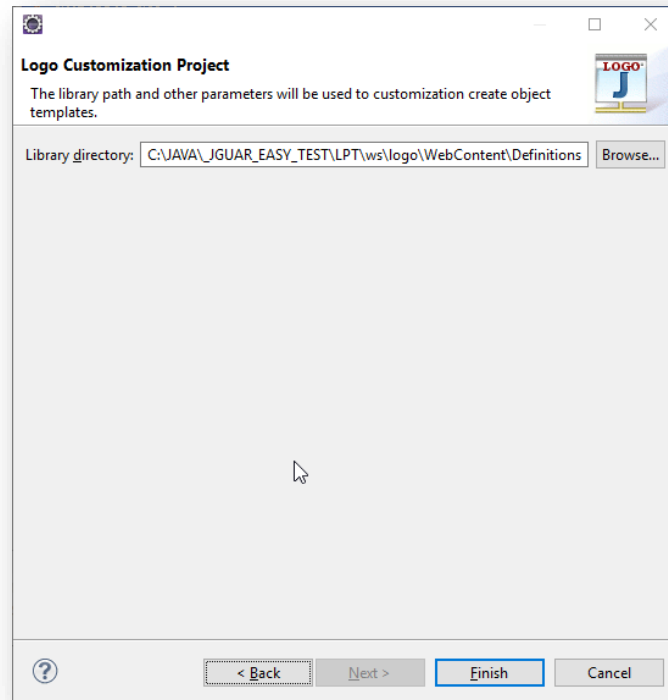


Wizards that you can use for customization are listed under Logo Platform Tailor folder in the window that opens. You must select a wizard from the list to create a project.



Continue by writing customization project name that you want to create in wizard.

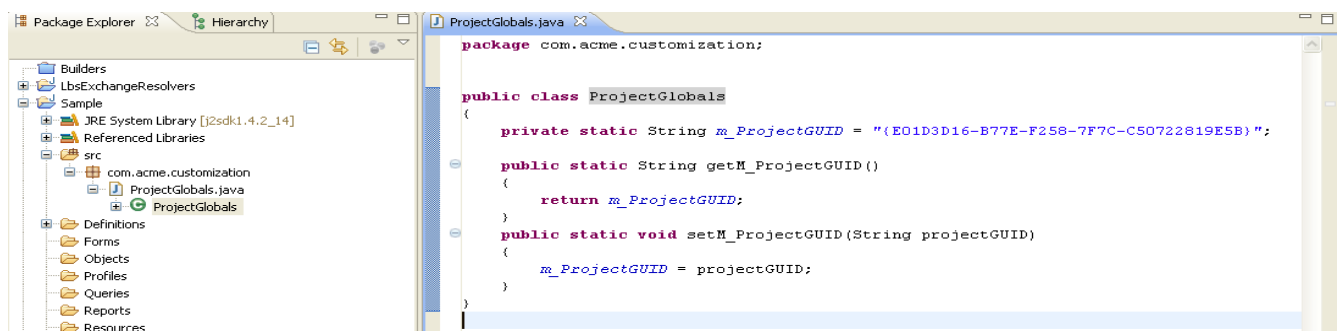




* LbsDefinitions.jar is a version dependent jar file that keeps all object definitions of j-guar product. Up-to-date LbsDefinitions.jar file can only be created via <http://localhost:8080/logo/Status>

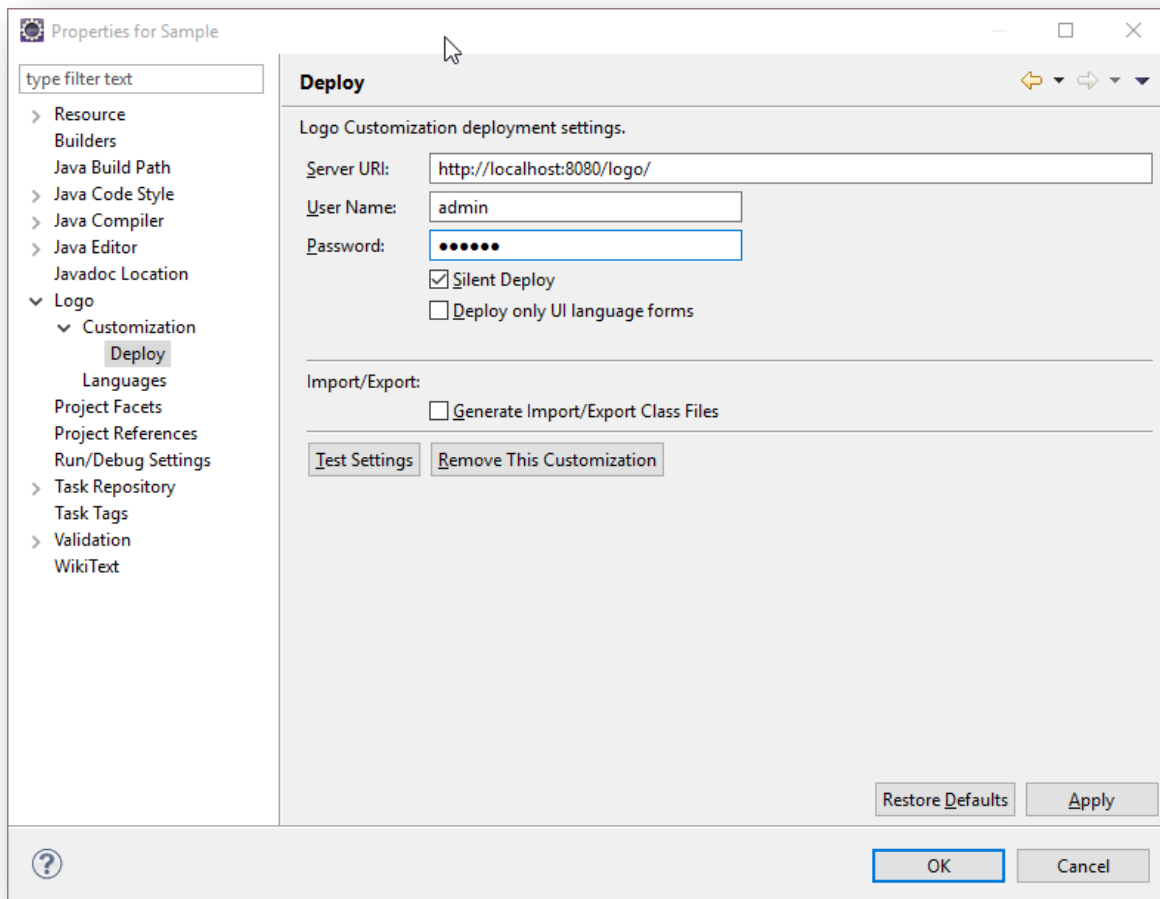
2.2. Project Settings

The value that you see in Project GUID field indicates the unique ID of the project. Since this value will be needed when coding, it is kept in **ProjectGlobals** file under SRC folder of the project.



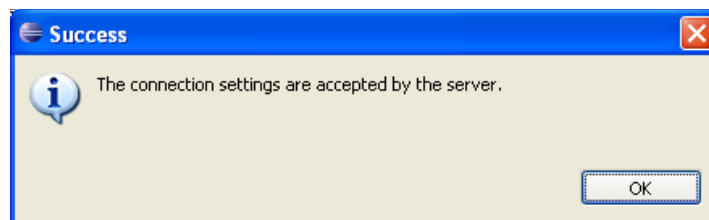
2.3. Connection Settings

You must adjust connection settings to have Customization Project connect with j-guar product. Right click on the project and select **Properties**. On the left of project properties window, expand **Logo** and then **Customization** option. Click **Deploy**. When opened initially, connection information will be shown empty. Click **Restore Defaults** option to bring default connection settings. You can edit these settings and then click **Apply** button.



You can also check if connection is successfully established with the specified settings by clicking **Test Settings** button. You receive a warning messaging that informs you about the status of connection.

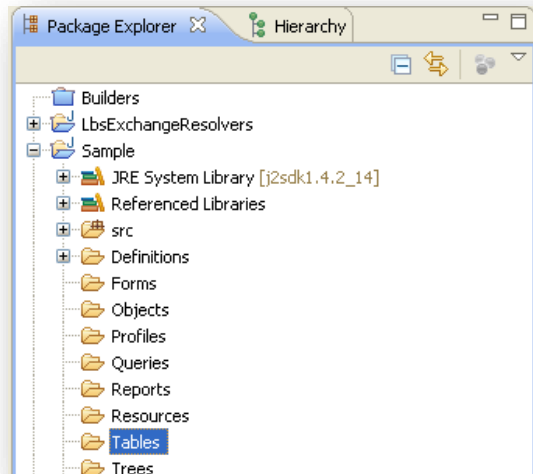
Note: Note that server must be open to test connection.



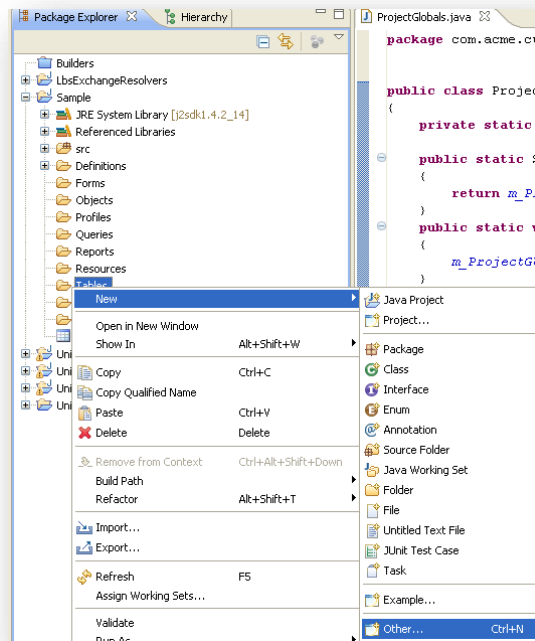
3. FOLDERS IN THE PROJECT AND HOW TO USE THEM

3.1. Tables

Tables are kept in **Tables** folder in the customization project.

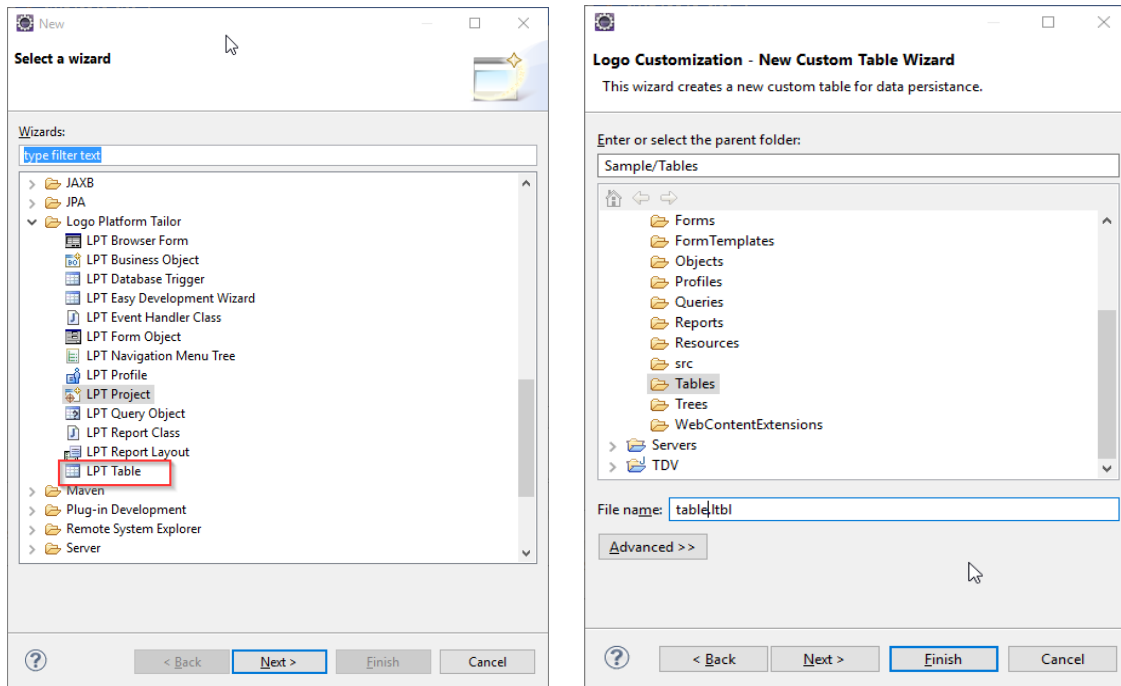


Let's see an example of how to create tables. In this example, we will make a table definition to use in defining machines. To do so, let's right click on Tables folder and click **New > Other** in the menu to open wizards list.

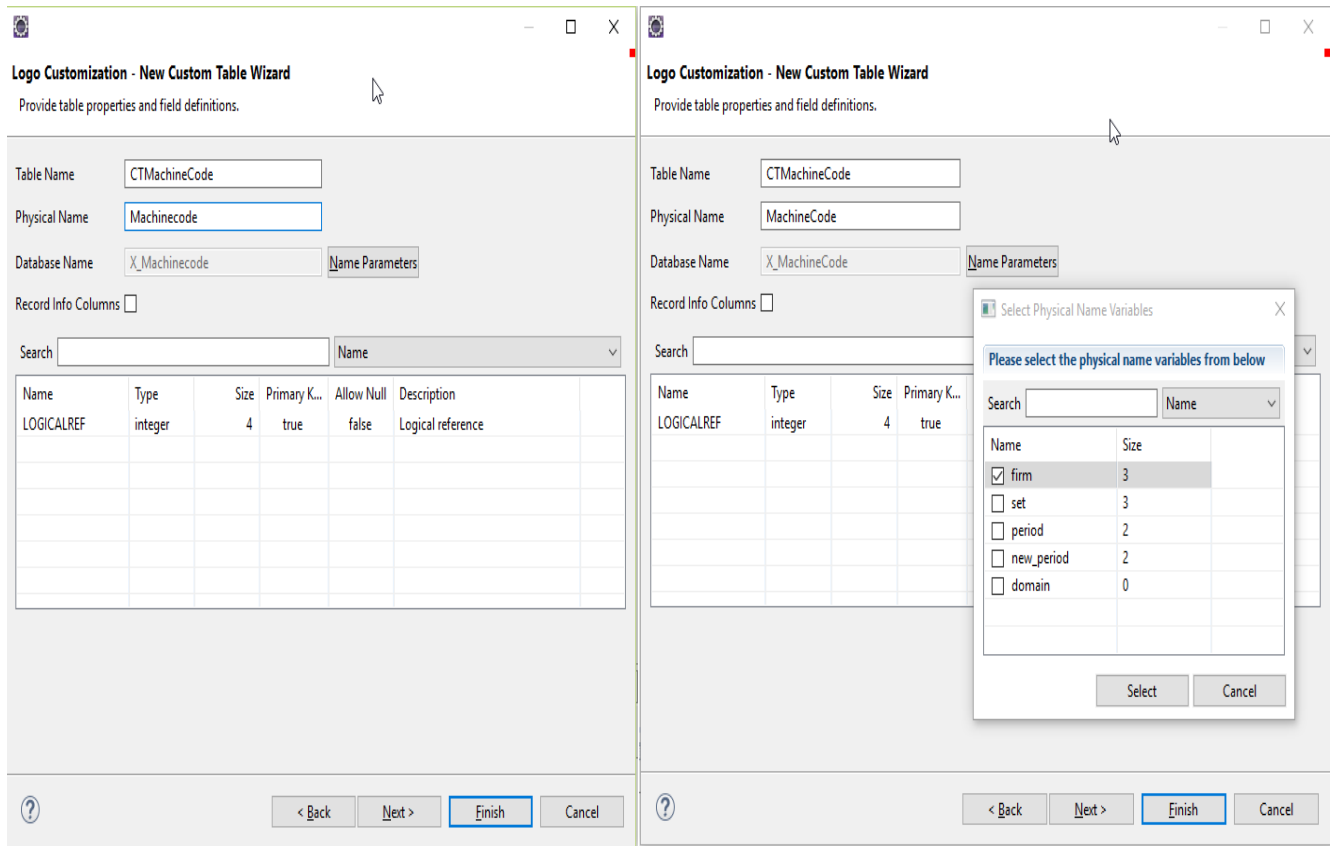


Select Custom Table in this window and click **Next**. In the next window, you'll be prompted to enter a name for the table.

Note: The name you enter for table at this step is NOT the name of the table that will be created in database. It is only the name of the table definition.



The name of the table that should be created in database is entered in **Physical Name Prefix** field in the window that opens after clicking **Next** button. You can designate whether this table is a firm and/or period table in the small window that will open after clicking **Name Parameters** button.



“**Logicalref**” column is the **primary key** column that is in all tables of j-guar application; for this reason, it is generated automatically for each table that is created.

We add two columns named **Code** and **Description**, and click **Finish** button to complete table generation operation.

Logo Customization - New Custom Table Wizard
Provide table properties and field definitions.

Table Name:

Physical Name:

Database Name: [Name Parameters](#)

Record Info Columns

Search: Name:

Name	Type	Size	Primary K...	Allow Null	Description
LOGICALREF	integer	4	true	false	Logical reference
CODE	string	30	false	false	
DESCRIPTION	string	30	false	false	

< Back Next > **Finish** Cancel

Note: You must update customization tables in j-guar Administrator for tables to be created in database physically.

3.2. Objects

Two-way data transaction between forms and tables is performed by objects. It is safe to say that Objects form a layer between tables and forms. There are two types of object in j-guar:

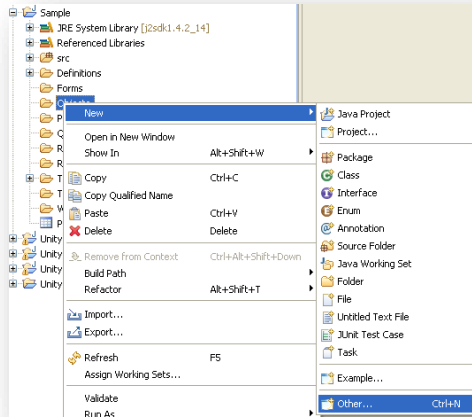
- Business Objects
- Custom Business Objects

The difference between these object groups is that Business Objects belong to j-guar and Custom Business Objects indicate the new objects that are generated by the person who developed the customization.

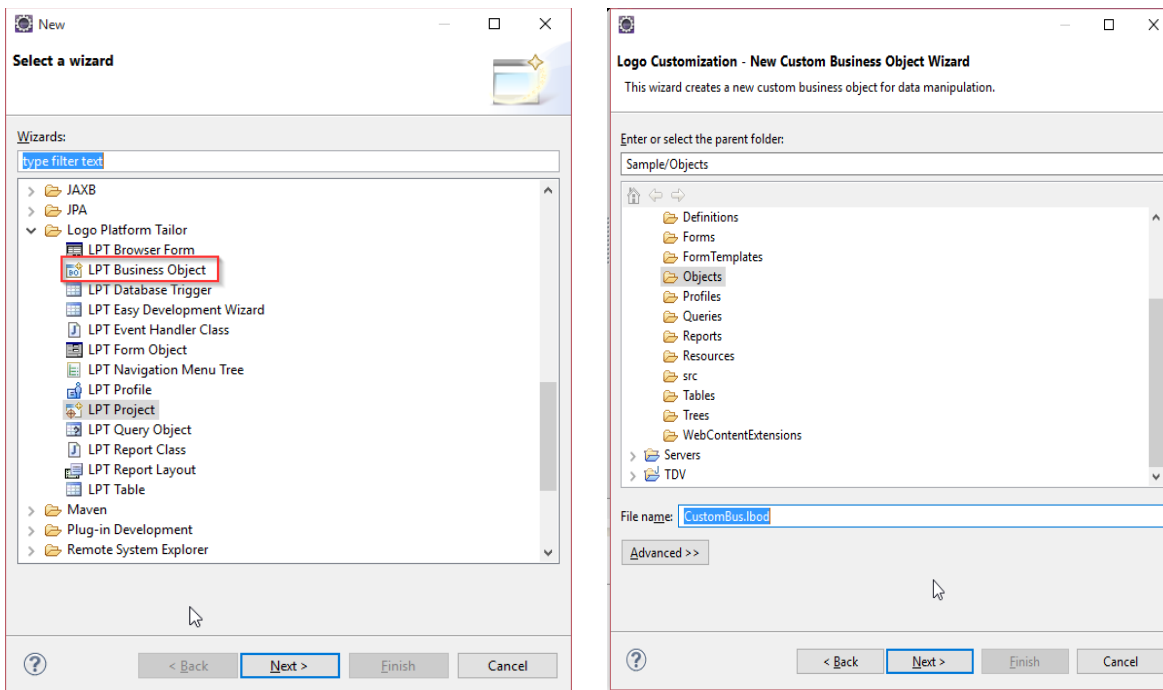
3.2.1. Creating a new object

We created a table for Machine Codes in section [3.1](#). Now, let's give an example by making an object definition for this table.

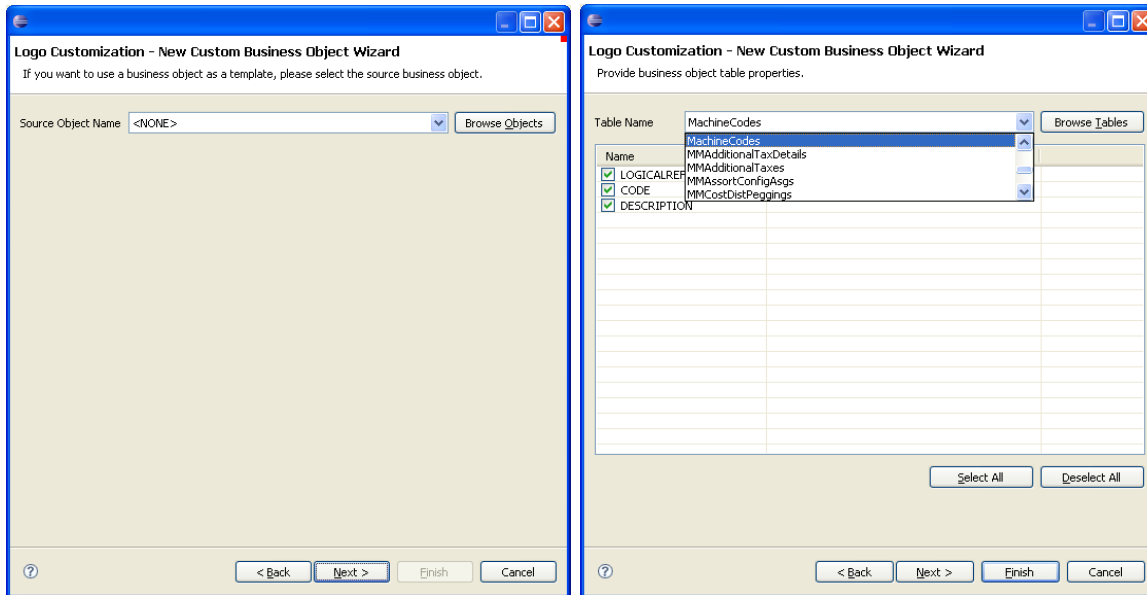
Right click on Objects folder. Then click **New > Other** in the menu to open Wizards window.



We mentioned that Custom Business Objects are the new objects that are created by the person who performs the customization. Therefore, we select **LPT Business Object** and click **Next**. In the next window, we'll be prompted to enter a name for the object we want to create. We enter a name for our object and click **Next**.



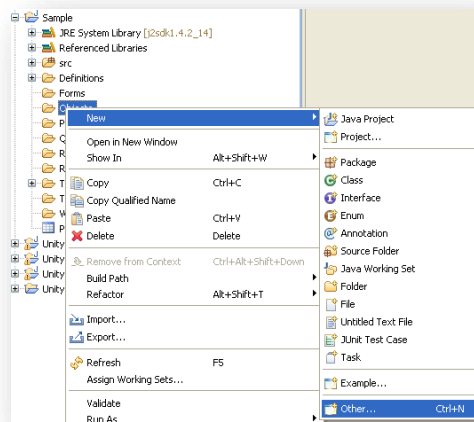
Let us proceed without making any selection in the next window by clicking **Next** button. Now, we need to enter the table to which the object we are creating will be linked. At this step, we select CTMachineCodes table that we created in section [3.1](#). After selecting the table, system will display the columns of this table in the grid on screen. If there is any column that you do not want to include in the object you are creating, you must uncheck this column and then click **Next** button.



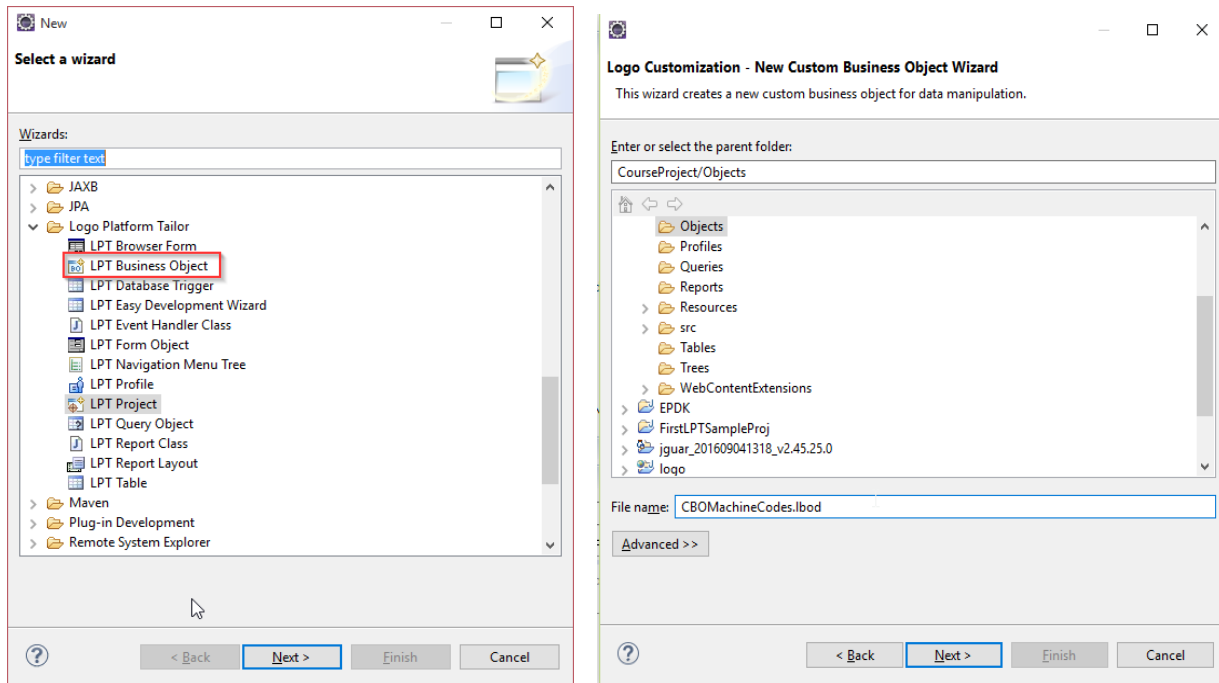
We proceed without making any selection in the next window by clicking **Next** button. We reach the final page of the wizard. Click **Finish** button without making any operation and end the wizard.

3.2.2. Expanding an existing product object (Extension)

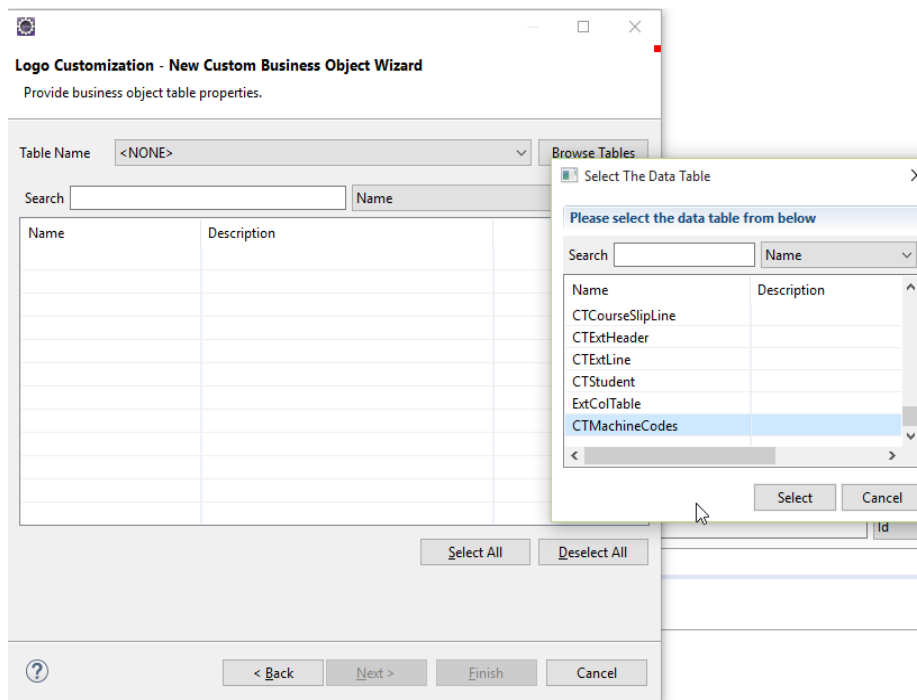
In this example, let's see how to expand an AR/AP card object by adding a field. Right click on Objects folder. Click **New > Other** in the menu to open Wizards window.



As in the previous example, we select **LPT Business Object** and click **Next**. In the next window, we give a name for our object, and click **Next** button to open the next page.



We proceed without making any selection in the next window by clicking **Next** button. Now, we select a table to which our object will be linked. After selecting the table, system will display the columns of this table in the grid on screen. Click **Finish** button to end the wizard.



Logo Customization - New Custom Business Object Wizard
Provide business object table properties.

Table Name: CTMachineCodes Browse Tables

Search: Name: Name

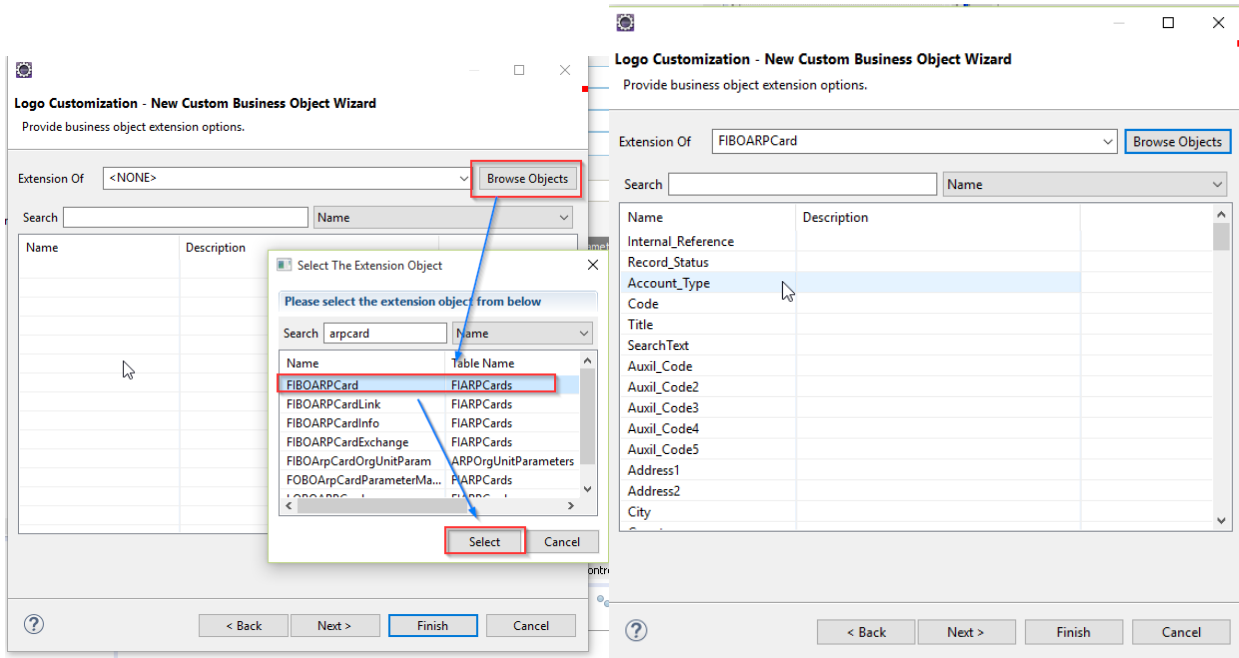
Name	Description
<input checked="" type="checkbox"/> LOGICALREF	Logical reference
<input checked="" type="checkbox"/> CODE	
<input checked="" type="checkbox"/> DESCRIPTION	

Select All Deselect All

? < Back Next > Finish Cancel

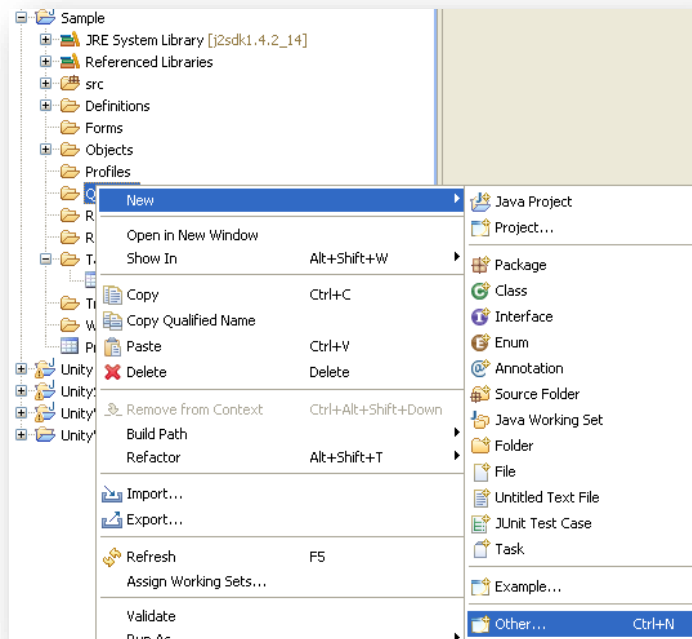
We add new fields to the product forms by using objects. To do so, we need to make **Extensions** definitions. We can add a second object over the same table and link it to the AR/AP card. See [Customization in Product Forms](#) to see how they are used on forms.

What we'll do next is to define an object with a different name and to select the product object to define **Extension**. The object name we'll be using in this example is CBOExtMachineCodes.

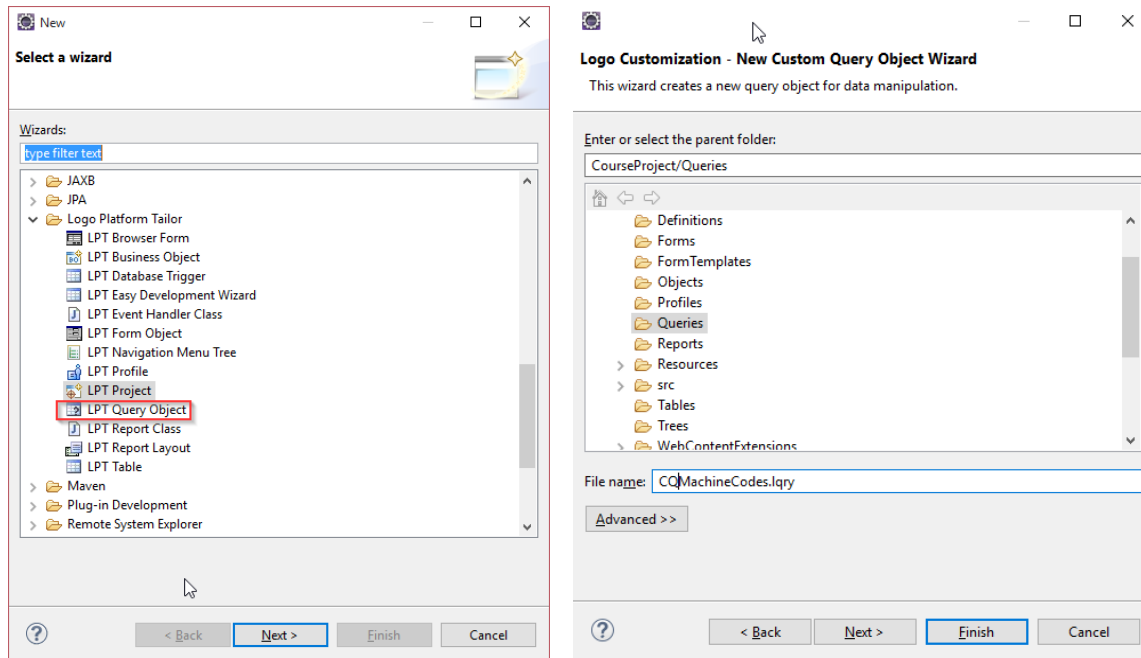


3.3. Queries

Customization query objects (Query Business Object) are kept in Queries folder. To create a new Query object, right click on Queries, and then click **New > Other** to open Wizards window.



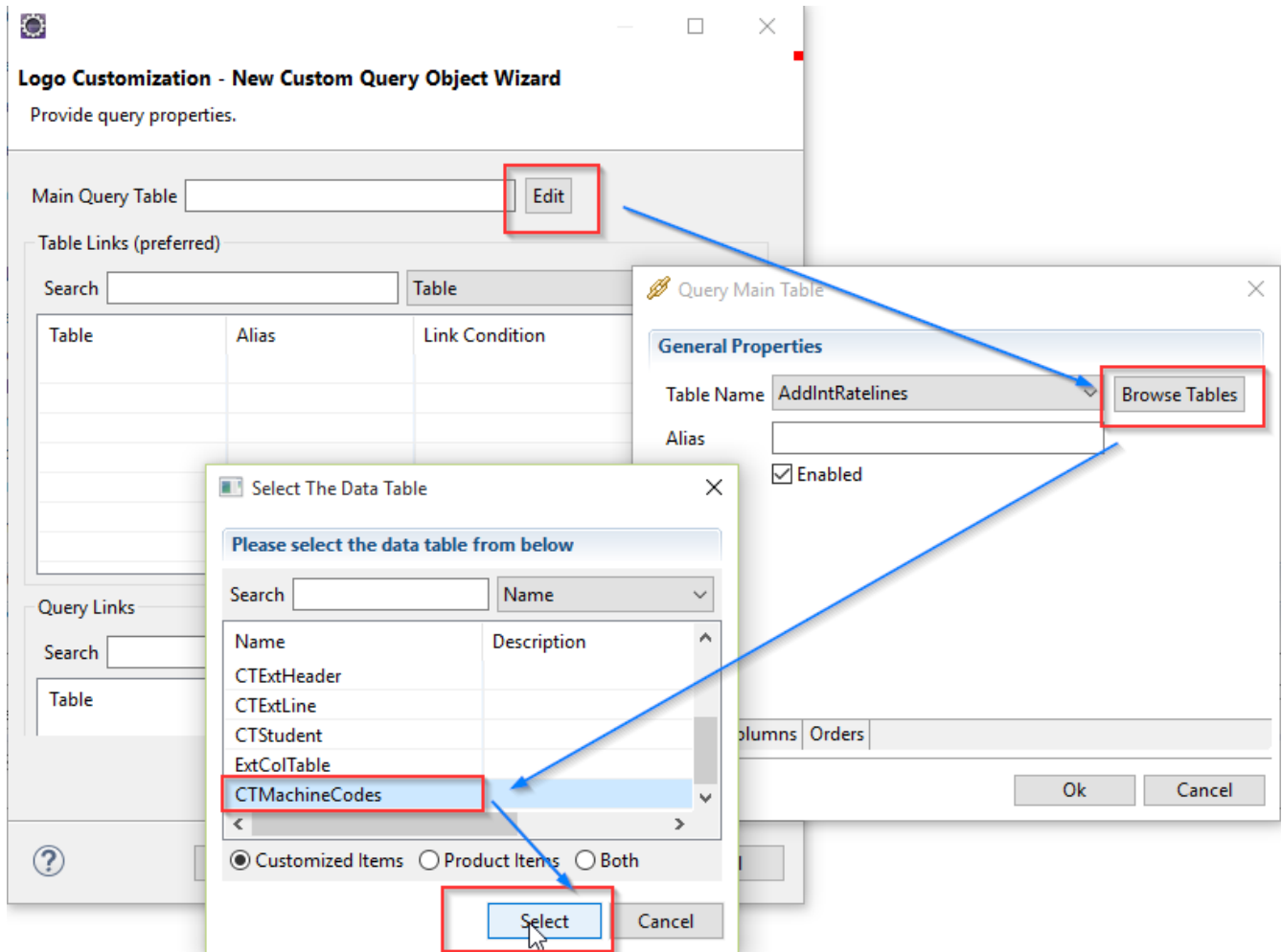
We select LPT Query Object in the window that opens, and then click **Next** button. In the next window, we'll be prompted to enter a name for the object we want to create. Let's enter a name and click **Next** button.



The next window list the existing queries. We can either use one of these queries as a template, or we can create a new query. To do the latter, we select **<Empty>** and click **Next** button.

An empty grid will appear. Right click on this grid and then click **Add** option.

A pop up window will open for us to select a table. Let's select **MachineCodes** table from the combo box and click **OK** button to close this popup window.

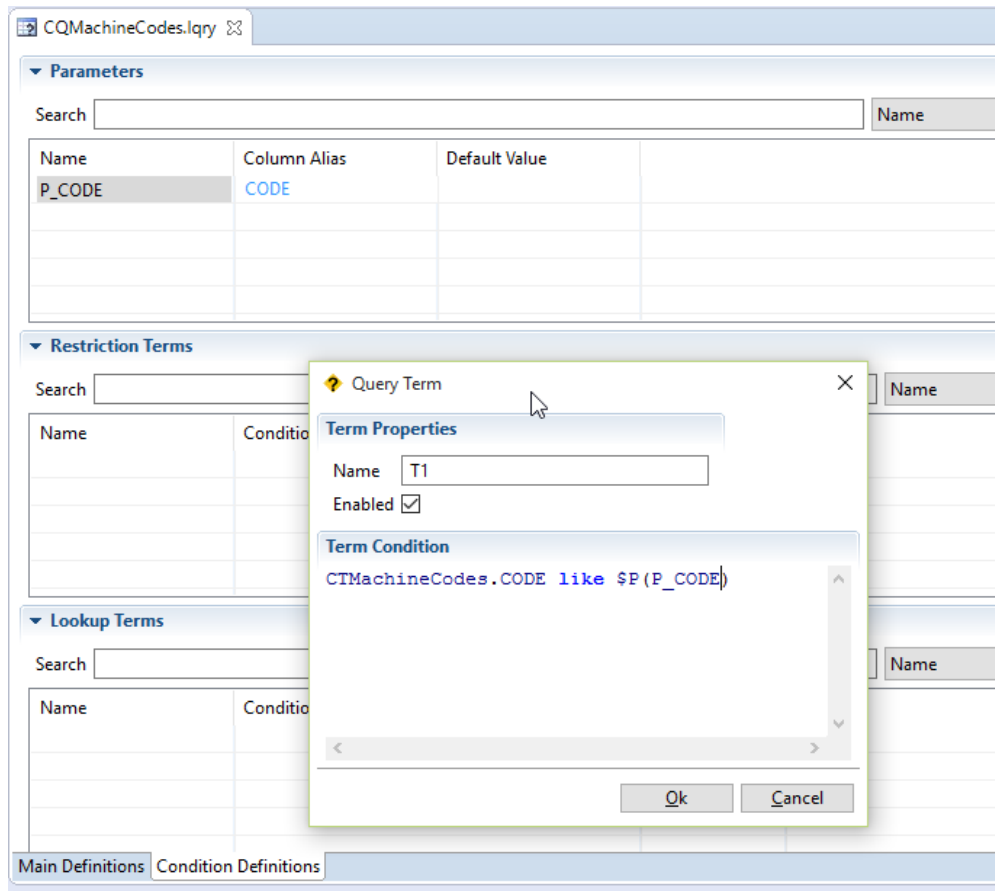


We end the wizard by clicking **Finish** button.

After closing the wizard, a query definition will appear on screen. This screen has two tabs.

- **Main Definition**
This tab shows columns that will be called by using SELECT statement and table and order definitions that will be combined by JOIN statement.
- **Condition Definition**
You can create conditions for the query under this tab. At the top of this tab, Parameters are shown. The parameter value that you define here can be used in **Restriction Terms** section in the middle of the Condition Definition tab.

Let's prepare a condition by which we can filter machine code over **Lookup Terms**. The first step is to determine parameter value. After defining parameter value with name **P_Code**, we use this parameter value in the condition as below.

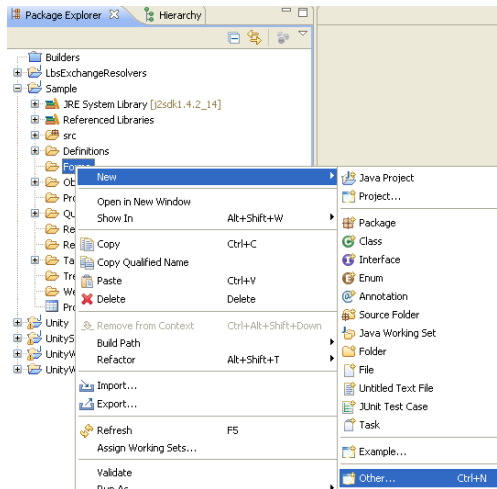


3.4. Forms

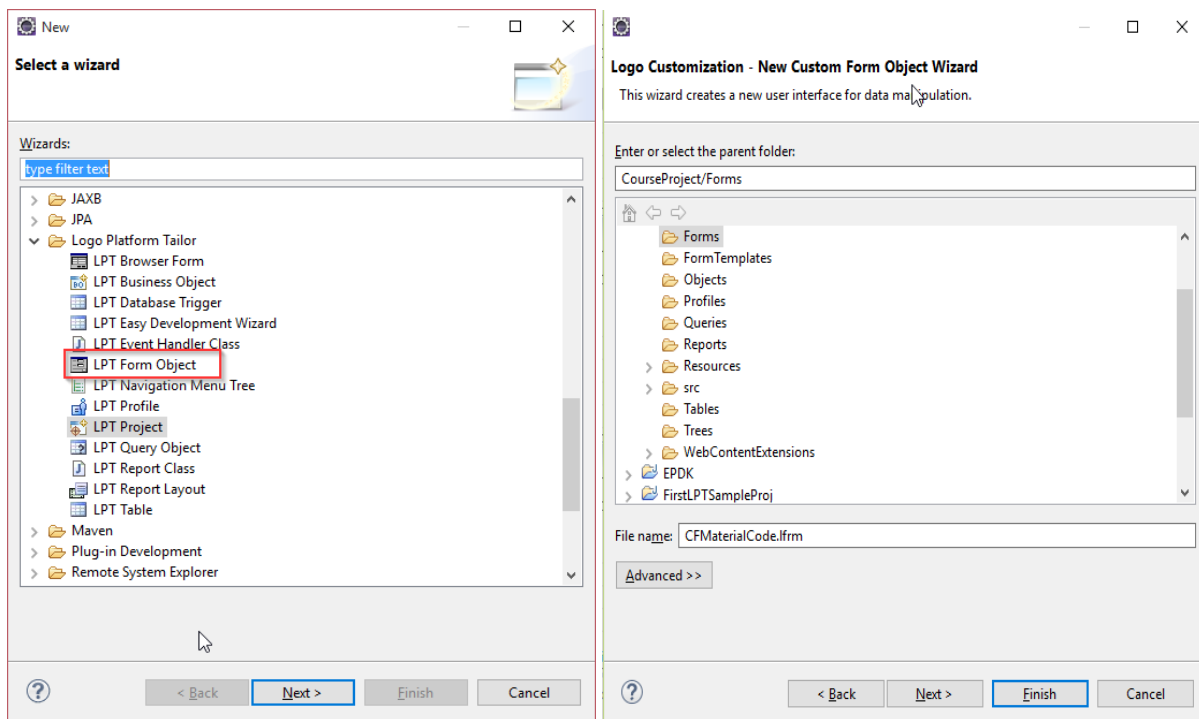
3.4.1. Creating a new form

Forms are kept in Forms folder in the customization project. It has a structure including j-guar cards and browsers. The structure of cards and browsers will be explained respectively.

Right click on Forms folder. Click **New** > **Other** to open Wizards window.



On the first page of the Wizard, select **LPT Form Object (Table Layout)** option and click **Next** button. On the next page, enter a name for the form object and click **Next**.



In Form Template window of the Wizard, you designate whether you want to use a template for the form you are creating. (Form templates are kept in “LbsDefinitions.jar” file.) If you want an empty screen to open, you shouldn’t change the default template option “<Empty>”. Let’s talk about the other options in this window.

- **Create event handler class:** This option automatically creates class file which will have you write the code for the form. It is checked by default, meaning that class file will be created.
- **Clear inherited events & properties:** If you are creating the form by using a template, meaning that if you are customizing one of the forms of the product, the product code behind the related form will

also be working in this new form that will be created automatically. Unless you want this product code to work, check this option.

Click **Next** to proceed.

Form Template

The available form templates are listed below.

Search Name

Name	Module	Description
<EMPTY>		Empty form
CMXFCommission.jfm	Satış Yönetimi	Komisyon Tanımı
CMXFCommissionBrowser.jfm	Satış Yönetimi	Komisyon Tanımları
CMXFCommissionLineBrowser.jfm	Satış Yönetimi	Satış Temsilcisi Komisyonl...
CMXFCommissionStateChange.jfm	Satış Yönetimi	Komisyon Durum Değişikl...
CMXFCommissionUpdate.jfm	Satış Yönetimi	Satış Komisyonları Güncel...
CMXFCustomer.jfm	Satış Yönetimi	Müşteri Kartı
CMXFCustomerBrowser.jfm	Satış Yönetimi	Müşteriler
CMXFCustomerContact.ifm	Satış Yönetimi	Müşteri Bağlantıları Kartı

Create event handler class

Clear inherited events & properties

Layout

Form Width Column Count

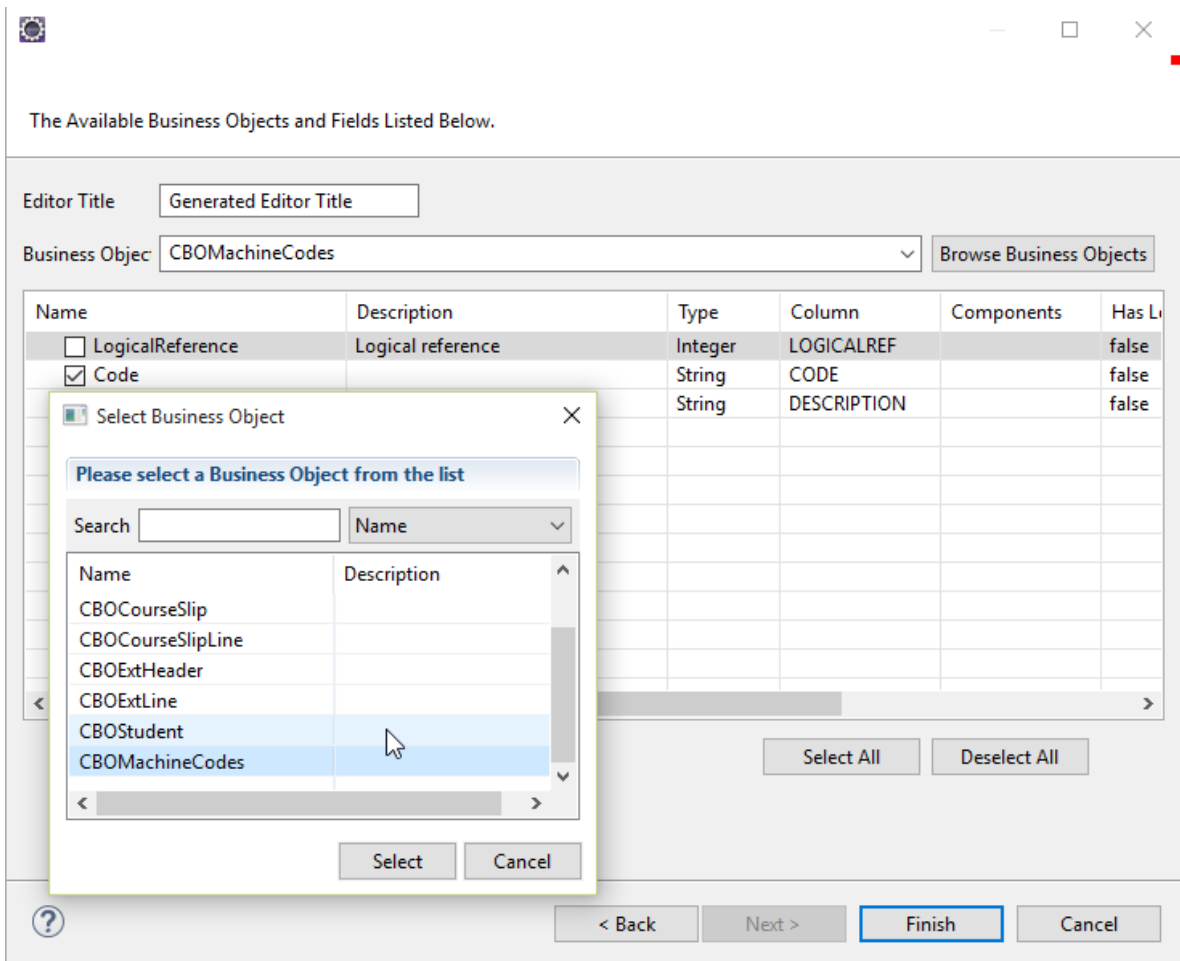
Form Height Row Count

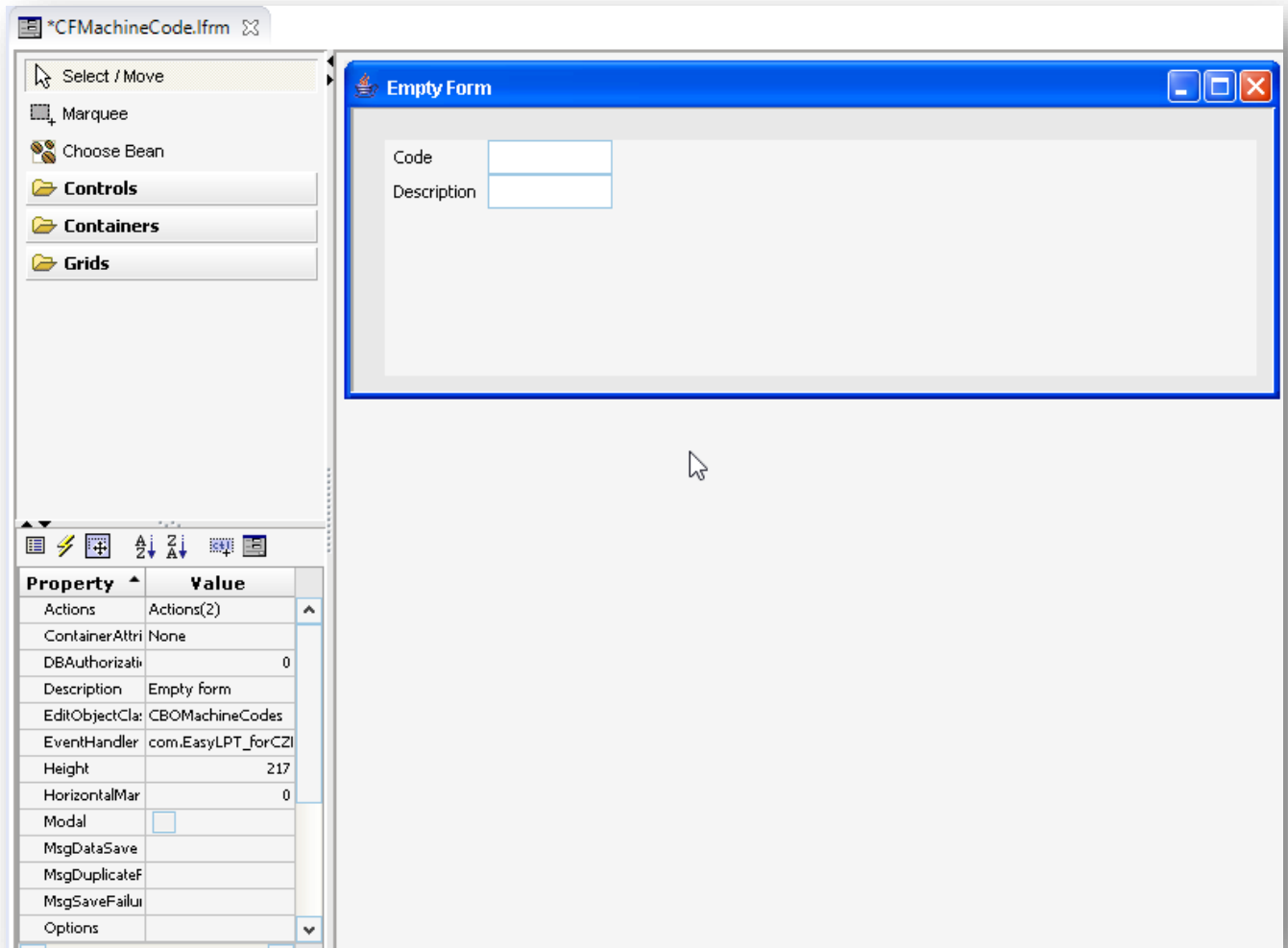
Append a "FILL" column to the end (for resize issues)

Append a "FILL" row to the end (for resize issues)

If the form that you are creating has a Custom Business Object, select this object. Otherwise, if you are defining a form independent of CBO, complete the operation by clicking **Finish** button.

To link a CBO, click **Browse Business Object** button. Select **CBOMachineCodes** that we created previously in Select Business Object window. After selecting, all fields in CBO are called in the Wizard. You select the ones you need from these fields.



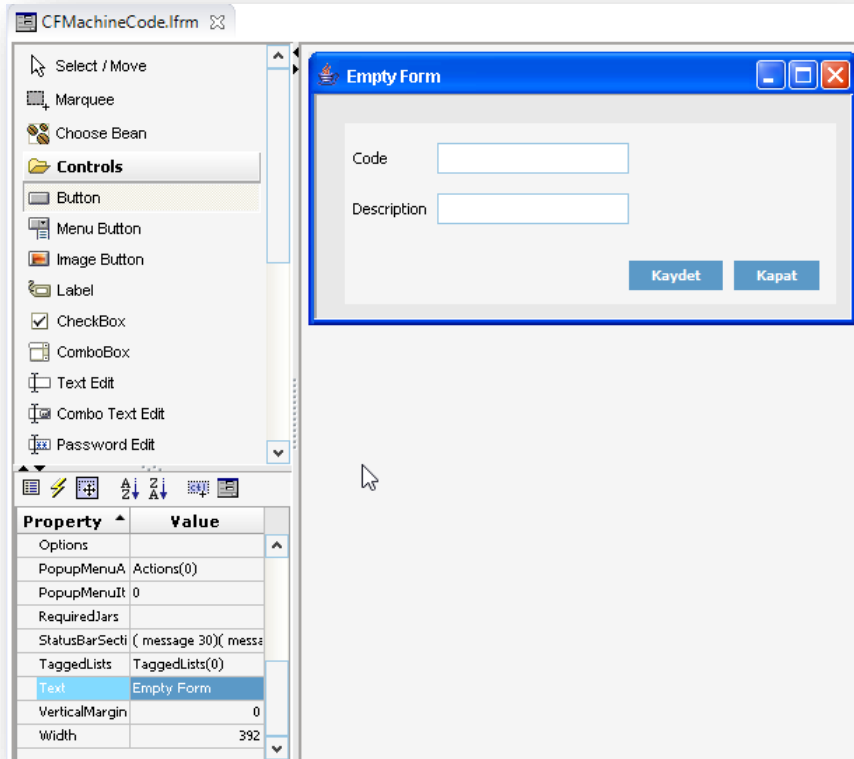


Let's continue with our Machine Codes example and create a card and a browser that are necessary for this example. We see that the object we created under section [3.2.1](#) is linked to the form.

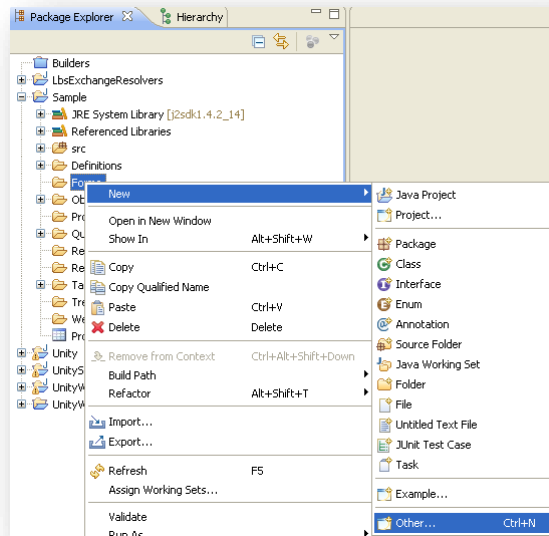
We add two buttons and name them "Save" and "Close".

In Property tab, we select **Save & Close** as Action feature for **Save** button, and we select **Close** as Action feature for **Close** button.

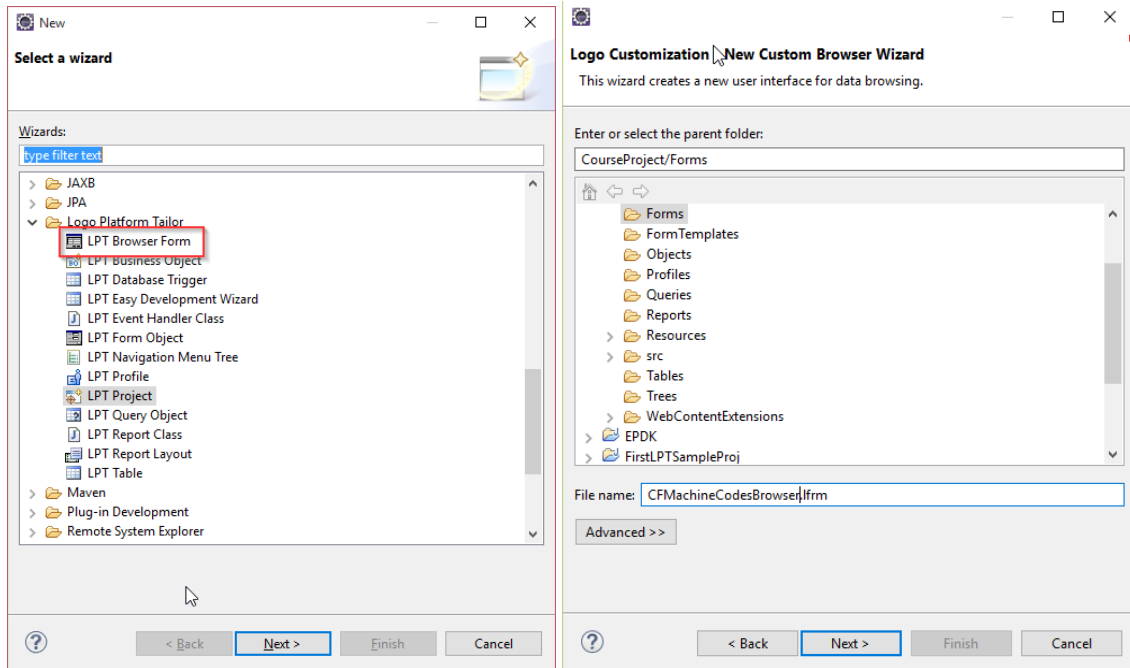
This is how we define our Card.



Let us start to create our Browser. **Right click** on Forms folder and click **New > Other** to open Wizards window.



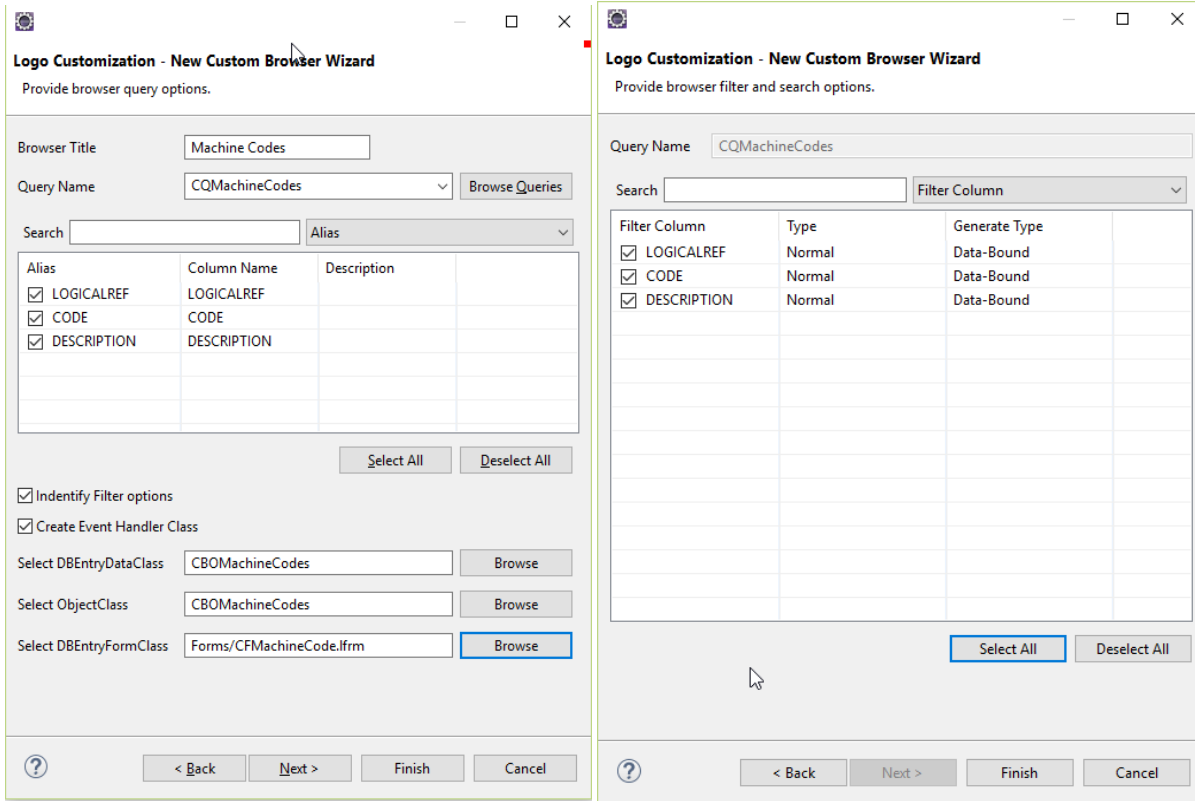
Select **LPT Browser Form** feature in the wizard window that opens and click **Next** button. On the next page, enter a name for the browser and click **Next**.



In this window,

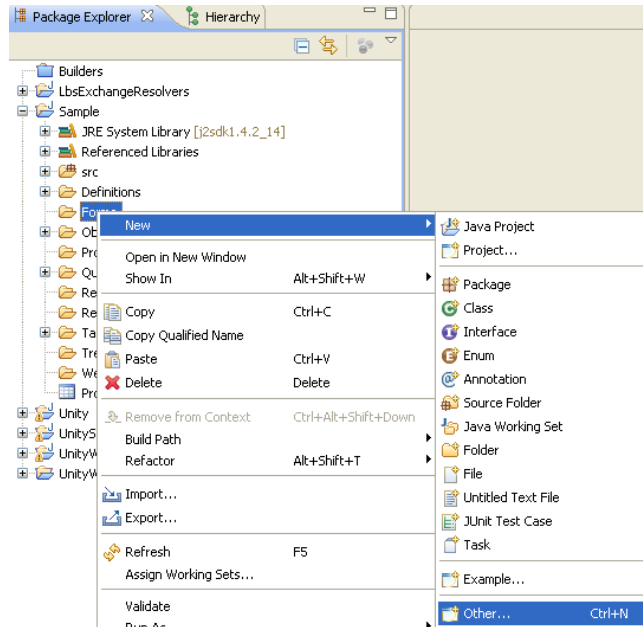
- Enter a heading in **Browser Title** field.
- Select the query we previously defined in **Query Name** field.
- After selecting Query, the fields that are defined in this table will be listed. Click **Select All**.
- Enter Business Object that we defined earlier in **Select DBEntryDataClass**.
- Enter Business Object that we defined earlier in **Select Object Class**.
- Select the form you want to open over Browser in **Select DBEntryFormClass** field and click **Next** button.
- Select the field you want to display in Browser in the window that opens. Click **Finish** to complete the operation.

Note: Since it is better to select all and then remove the fields that are not necessary over browser design-time, we select all and click **Finish** button.

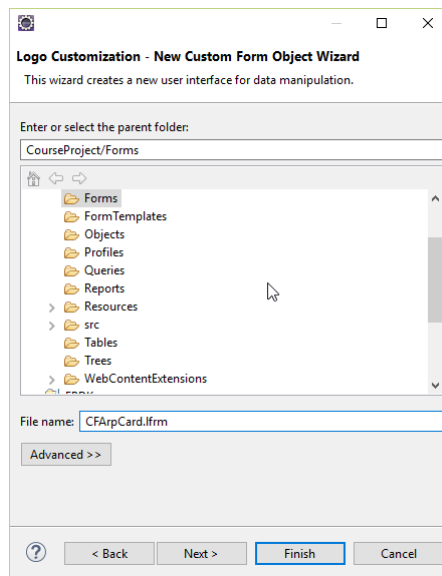
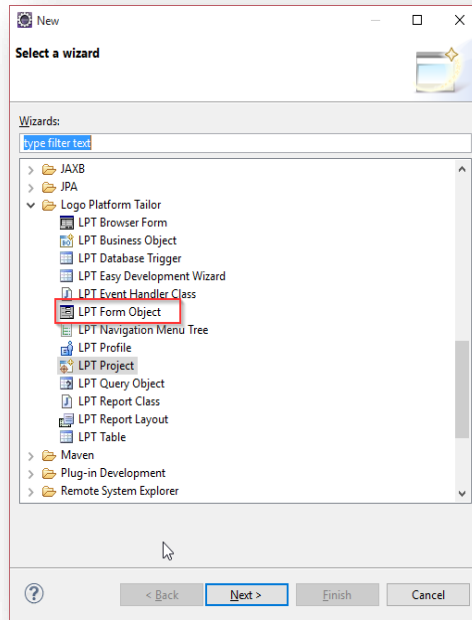


3.4.2. Customizing an existing product form

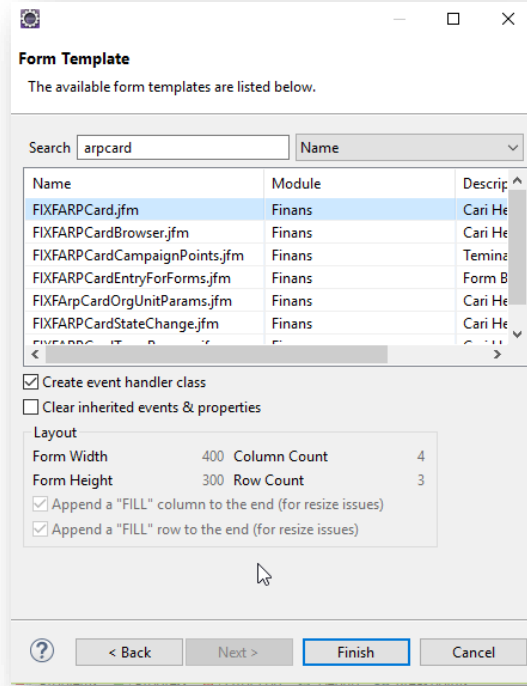
Let's continue with an example. **Right click** on Forms folder and click **New > Other** to open Wizards window.



In this wizard page, select **LPT Form Object** and then click **Next** button. Enter a name for the form in the window that opens and click **Next**.



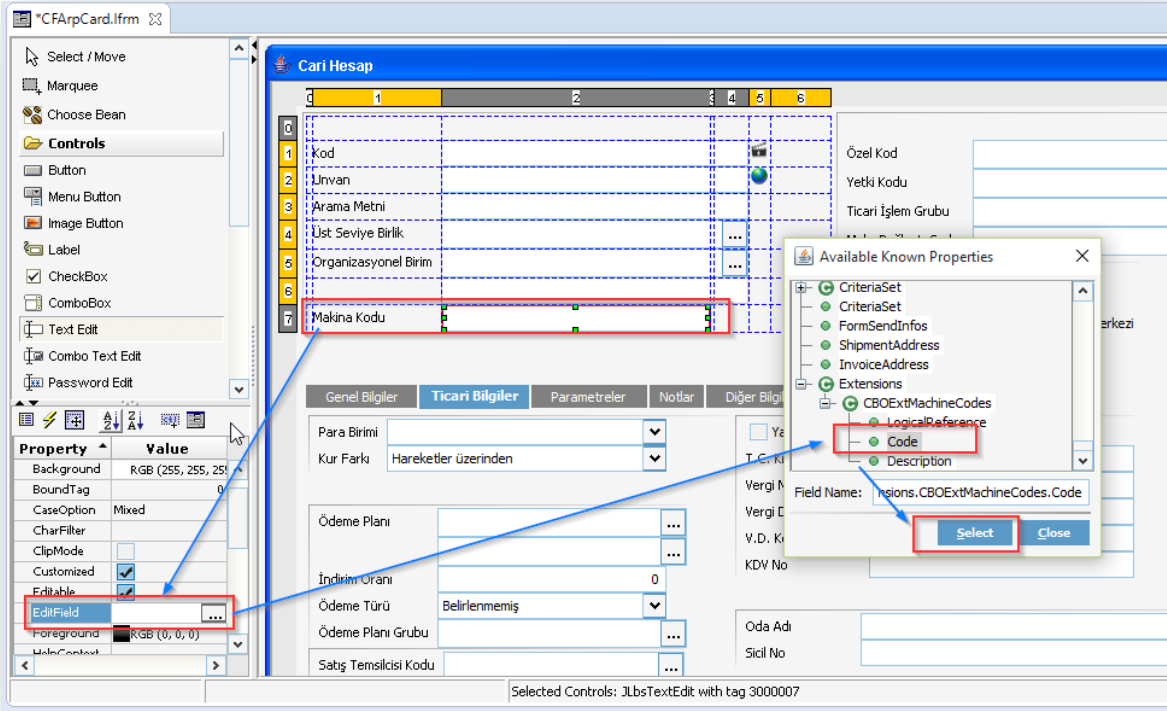
Select the product form you want to customize in the window that opens. In this example, we select **FIXFARPCard.jfm** AR/AP card and click **Finish** button to end the wizard.



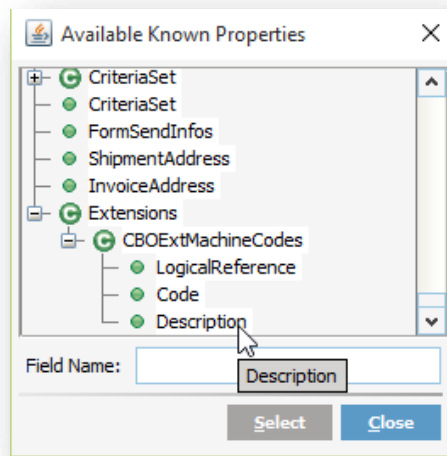
After closing the wizard, AR/AP card of the product card is displayed.

- We add label and textbox for the Machine Code in AR/AP card.
- We link the fields in our object to the textbox.

- After selecting textbox, we click **EditField** in Property. We see that object elements are listed in the popup window that opens.

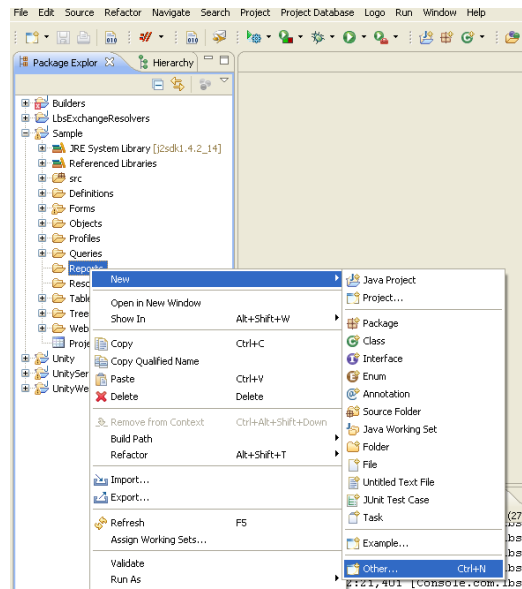


- Extensions, namely the elements that are included through customization, are listed at the bottom of the Object. We link these elements to the fields that we added in form with customization, and we end the process.



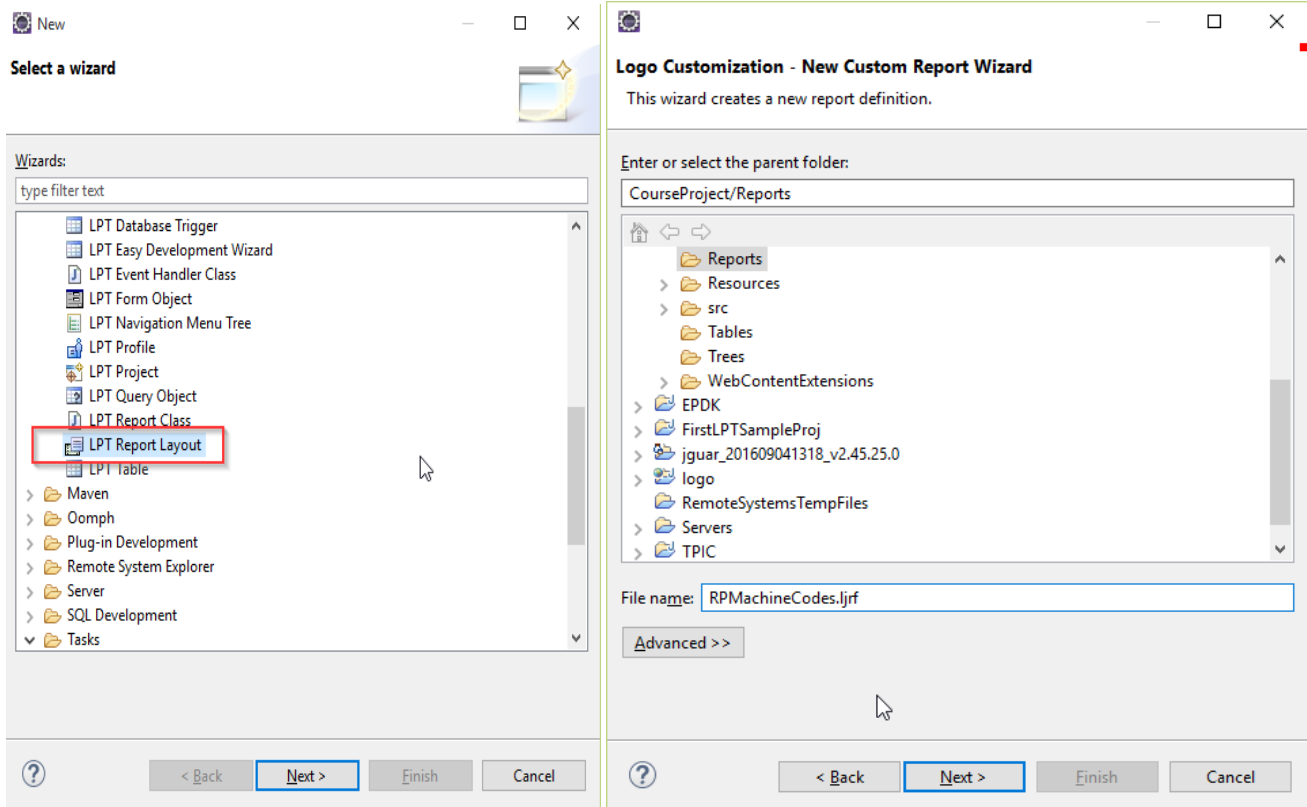
3.5. Reports

Right click on Reports folder and click **New > Other** to open Wizards window.

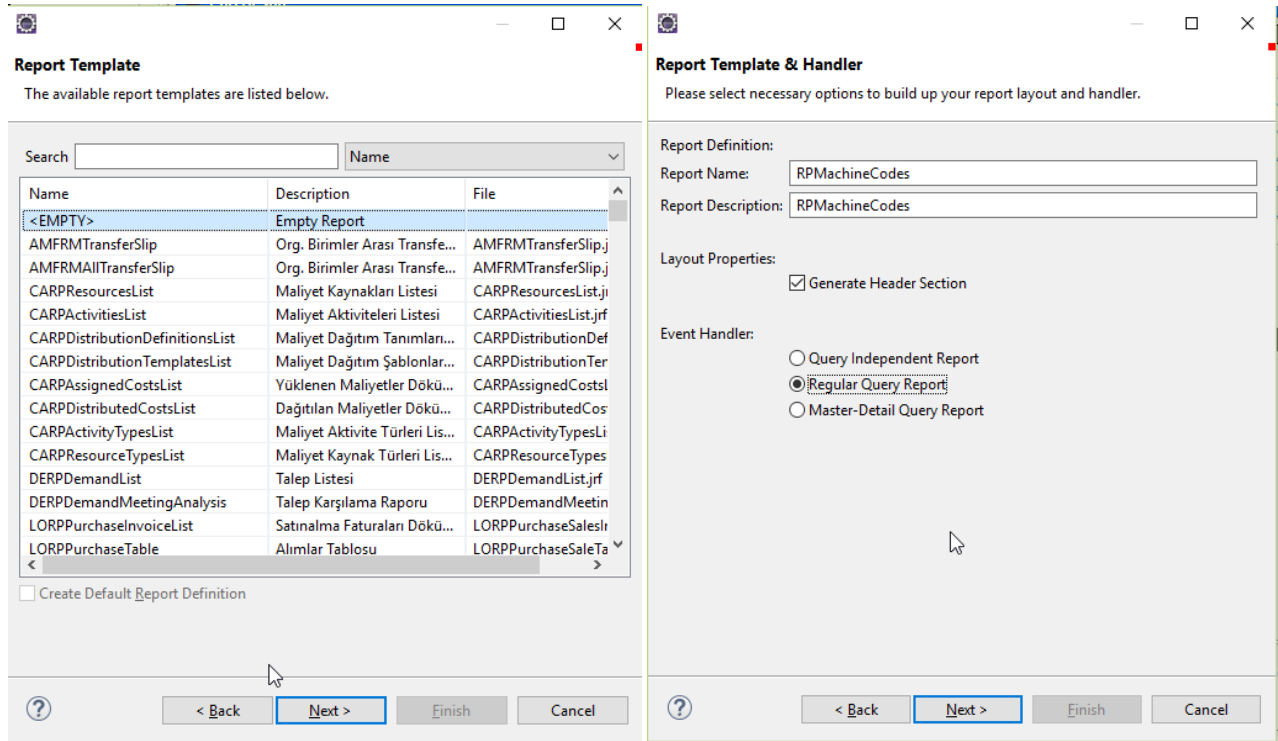


Let's continue with the example we prepared about Machine Codes and see how to prepare a report for the table records.

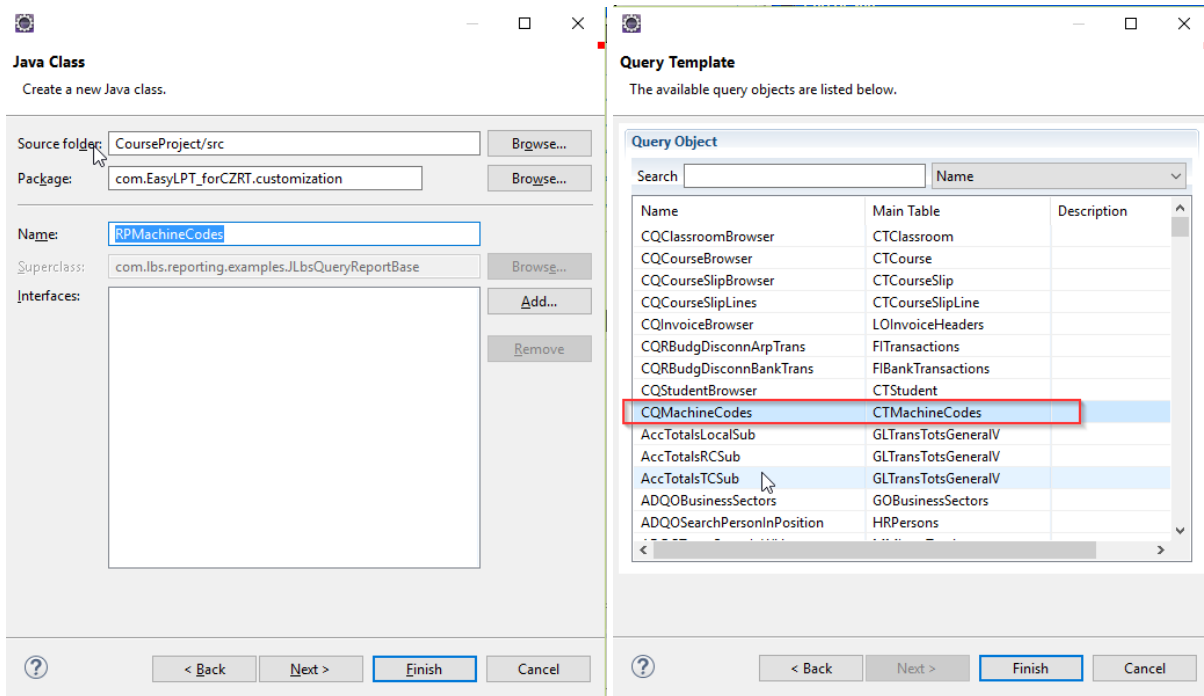
- Select **LPT Report Layout** in the wizard and click **Next** button.
- Enter a name for the report and click **Next** button.



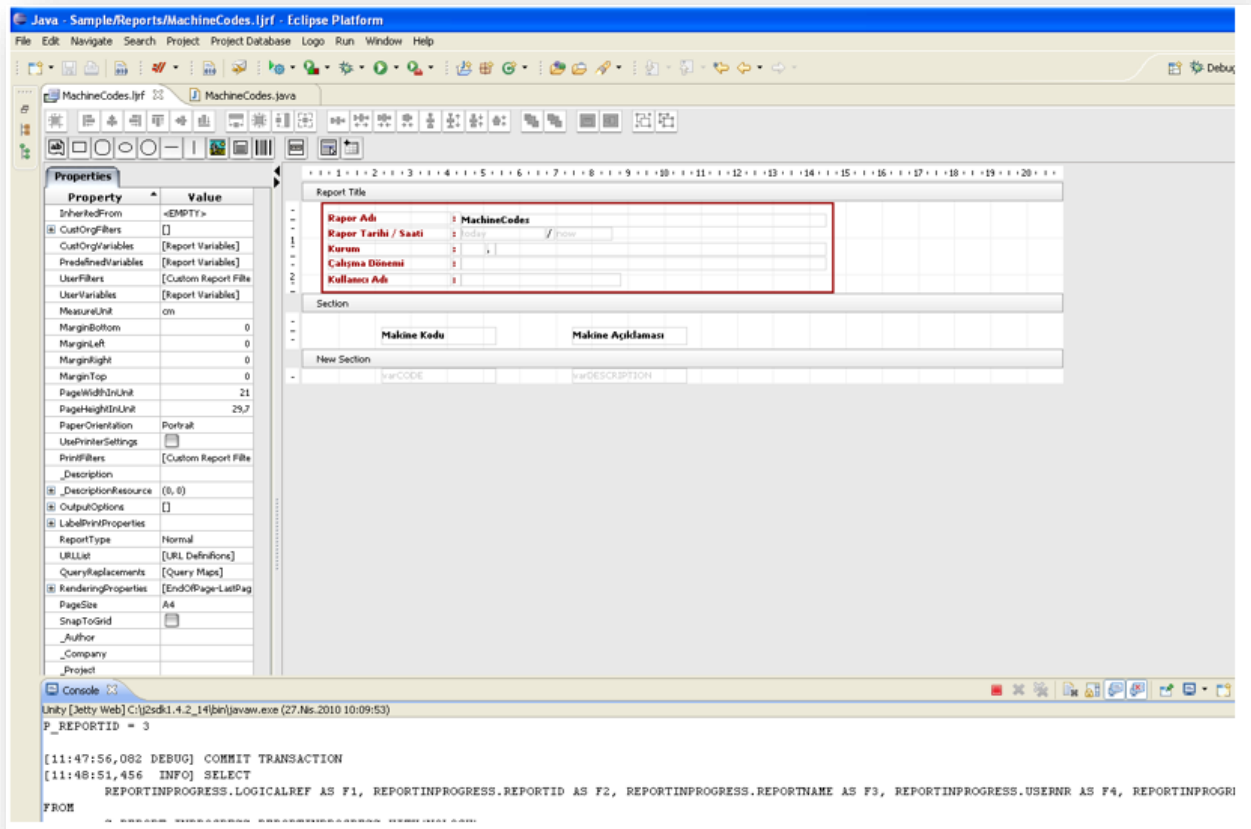
- Select a template for the report. If no template should be used, select **<EMPTY>** option and click **Next** button.
- You'll be prompted to enter a report name and a report description that will be used in report definition.
- Select **Regular Query Report** to use predefined query objects. The selection you make in this section will be coded in class file. Click **Next** button to continue.



- Enter a name for the class file of the report in the window that opens.
- Click **Next** to reach the final page of the report wizard. Select report query in this page.
- We select the report query that we prepared under section 3.3, and click **Finish** button to end the wizard.



- After ending the wizard, report design window opens and it opens as an empty window. Let's have the first section as the header. We add two rectangles for Machine Code and Machine Description. We set the section type to Header.
- We add a second section and we add two fields to show data in this section.
- We right click on Report Design and select **Import Query Fields**. It will import the fields of the query that we have designated in the Wizard into **User Variables** field. We select these variables for **VariableName** that is in the field properties.



- When we check out the Class file of the report, we see that a code block has been generated automatically. At this stage, if we edit ID value of the section ("2" in our example) where data will be displayed, our report will be ready.

```
RPMachineCodes.java
1 package com.EasyLPT_forCZRT.customization;
2
3 import com.lbs.reporting.JLbsReportLayout;
4
5
6
7 public class RPMachineCodes extends JLbsQueryReportBase {
8
9     public RPMachineCodes(JLbsReportLayout layout, IReportRenderer renderer) {
10         super(layout, renderer);
11         // TODO Auto-generated constructor stub
12         setMainQueryName("CQMACHINECODES");
13
14         m_BodySection = 1;
15     }
16 }
17 }
18 }
```

Report's layout and class information is kept in Reports.Irdf under Definitions folder. Information that is kept here is created automatically by the Wizard.

The screenshot shows the Logo Platform Tailor IDE interface. On the left, the Project Explorer displays a tree view with folders like 'src', 'Web App Libraries [logo]', and 'Definitions'. Under 'Definitions', there is a folder named 'reports.Irdf'. The main window shows the 'reports.Irdf' file with a 'Report Definitions' table. A 'Report Definition Properties' dialog is open, showing the following details:

Name	Class Name	Layout File	Description
StudentQueryIndepen...	com.LPT.customizati...	Reports/CRStudentQu...	Student QueryIndependent Exam...
StudentMasterQuery E...	com.LPT.customizati...	Reports/CRStudentM...	StudentMasterQuery Example
StudentMasterDetailQ...	com.LPT.customizati...	Reports/CRStudentM...	Student MasterDetailQuery Exam...
RPMachineCodes	com.EasyLPT_forCZR...	Reports/RPMachineC...	RPMachineCodes

The 'Report Definition Properties' dialog shows:

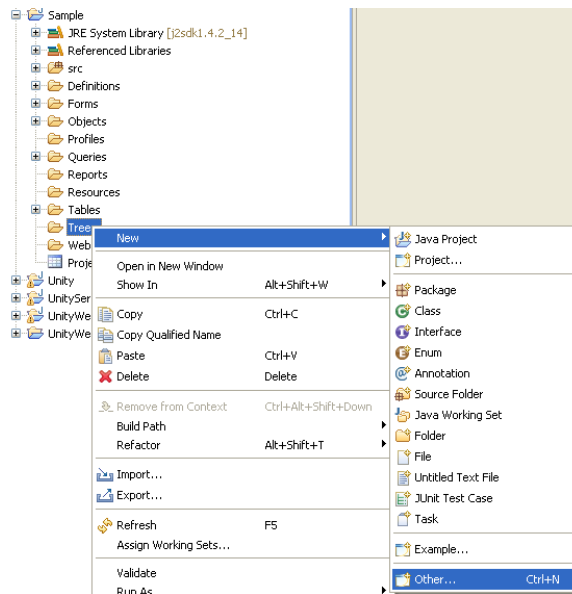
- Report Name: RPMachineCodes
- Layout File: Reports/RPMachineCodes.ljrf
- Report Class: com.EasyLPT_forCZRT.customization.RPMachineCodes
- Description: RPMachineCodes

3.6. Trees

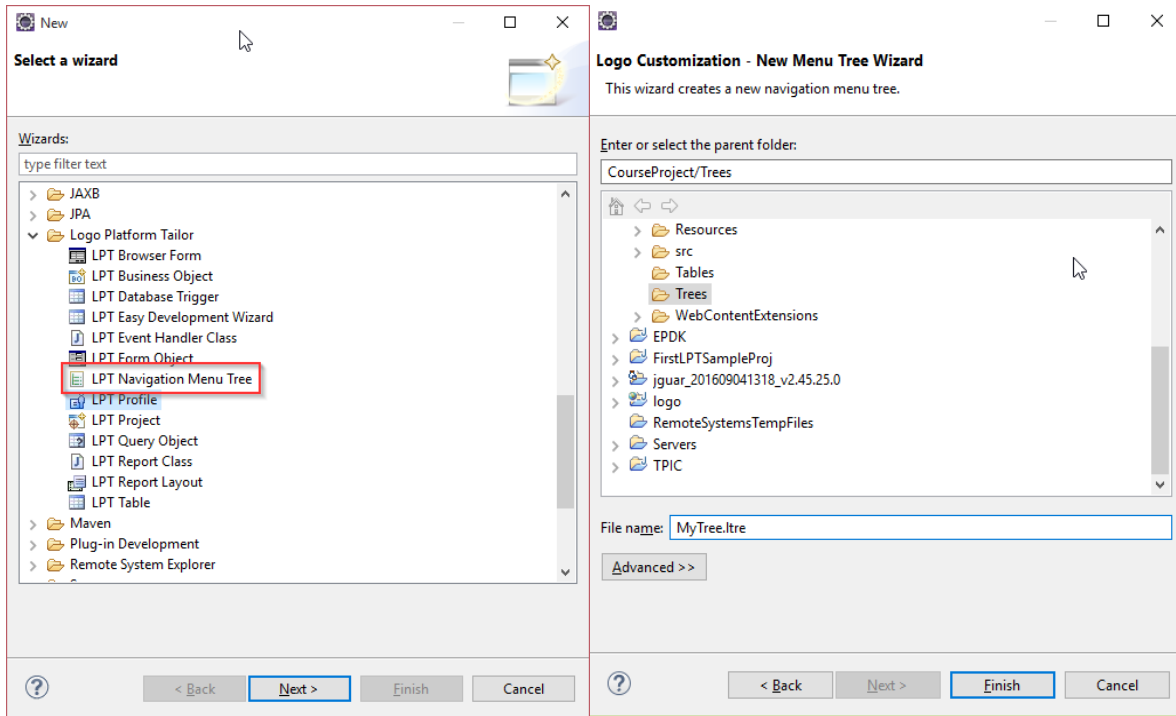
j-guar menu is in a tree structure and is divided into modules. Generally, the following child modules are listed under parent modules.

- Definitions
- Transactions
- Operations
- Reports

New modules or child modules can be added to this tree through customization. Right click on **Trees** folder and click **New > Other** to open Wizards window.

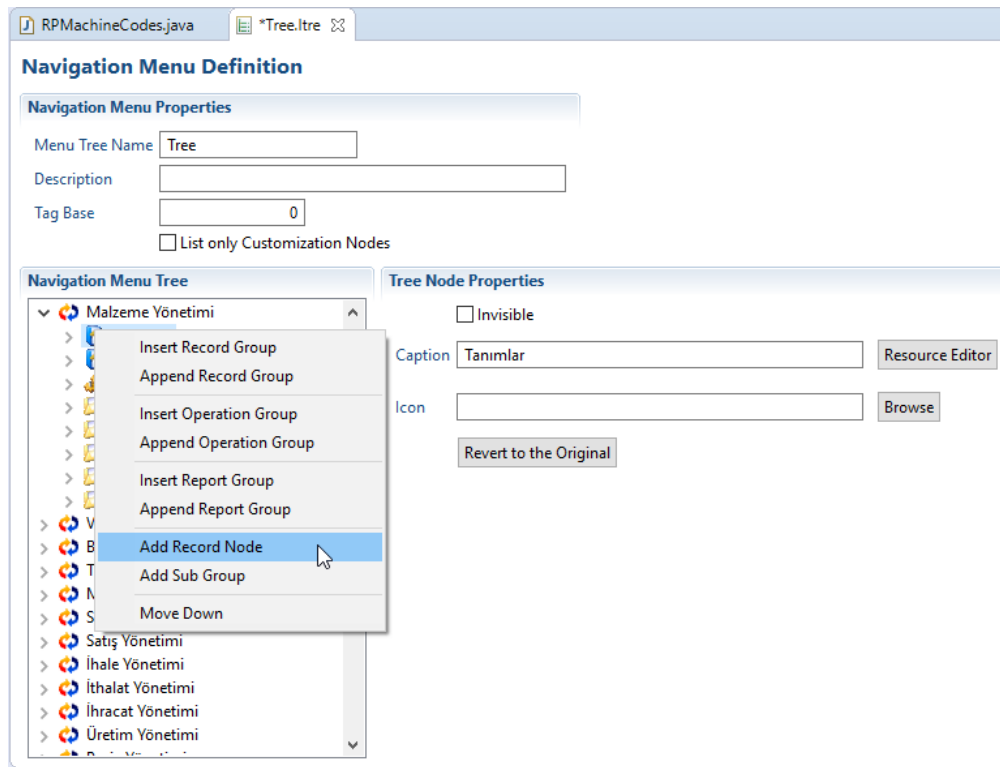


Select **LPT Navigation Menu Tree** option on the first page of the wizard and click **Next** to proceed. Enter a name for the tree on the window that opens and end wizard by clicking **Finish** button.



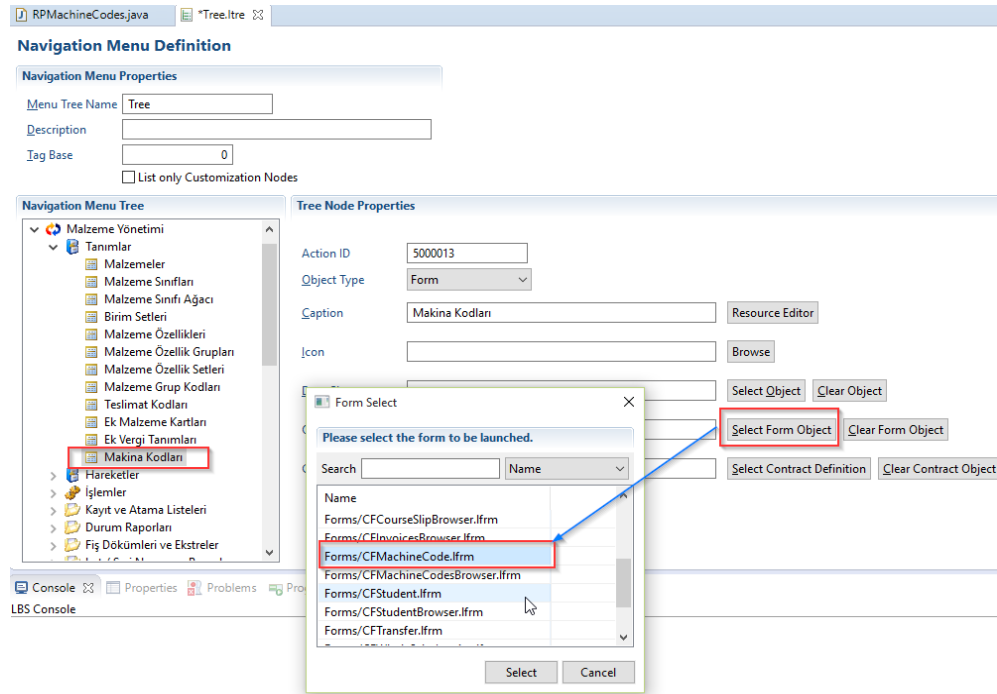
Tree structure will appear on screen.

Let's continue with our example. We add Machine Codes browser as an element to the Definitions child group under Material Management. To do so, we click Material Management on the tree and make child groups visible. Then, we right click Definitions and select **Add Record Node** option.



We set **Object Type** to form in the definition window that opens.

We write Machine Codes in **Caption** field for the heading.
 Lastly, we select the related form that we want to open, and complete our operation.

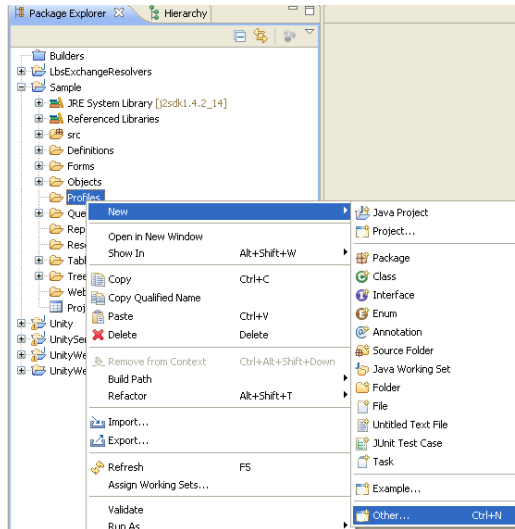


3.7. Profile

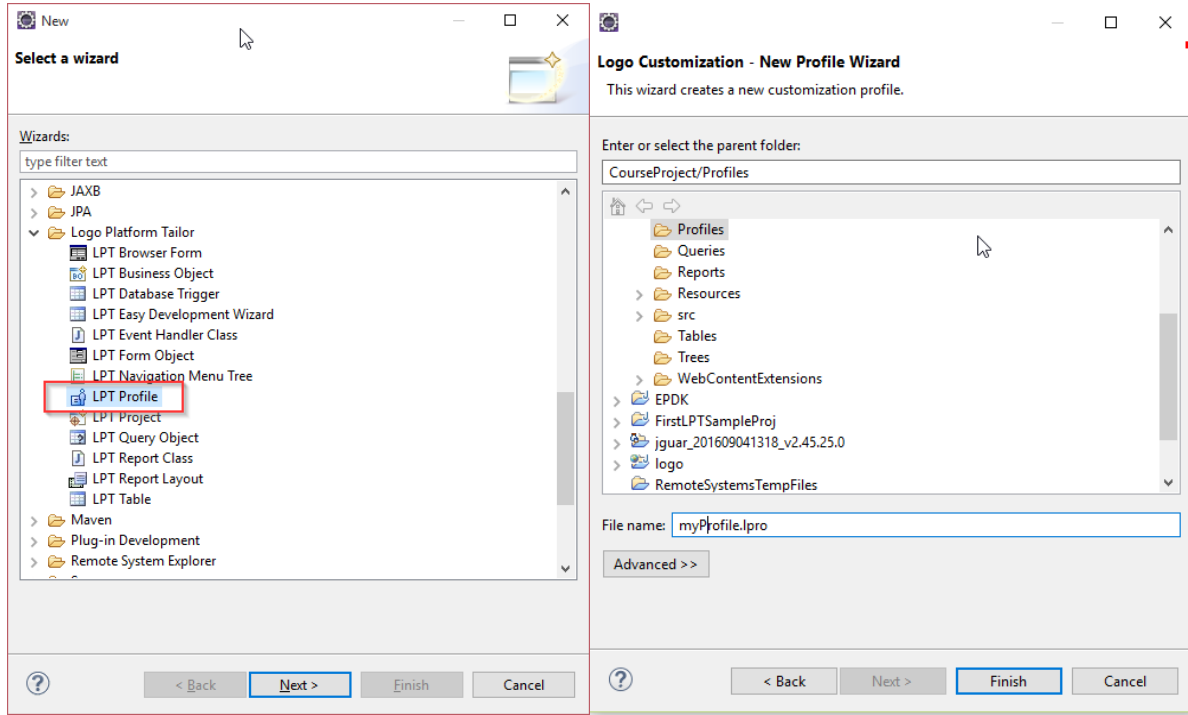
Profiles are the structures that keep customization content of j-guar users. Each profile needs a tree structure. Customizations that are defined in a tree apply only to the users who belong to the related profile.

ATTENTION: You can define multiple profiles in a single customization project.

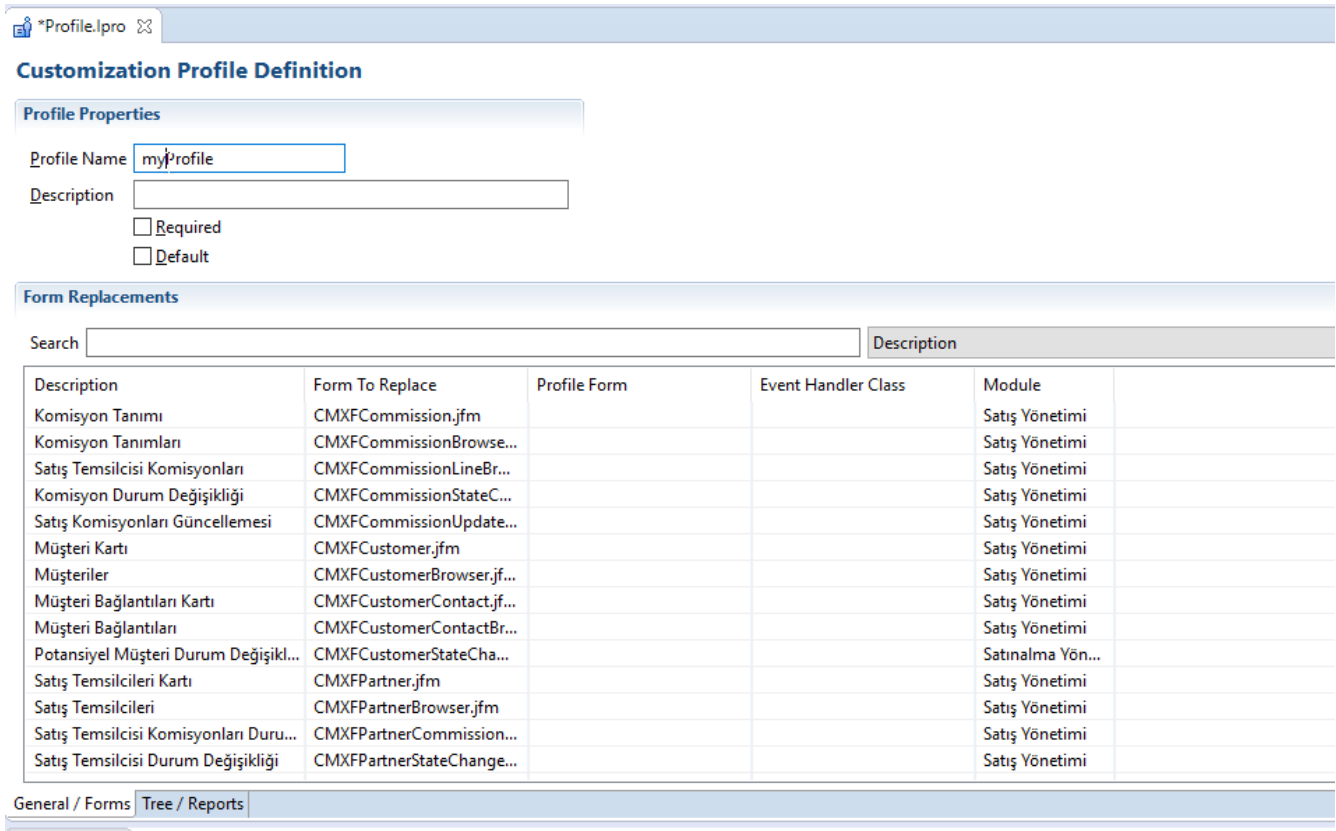
Now, let's see how to define a profile. Right click on **Profiles** folder, the click **New > Other** to open Wizards window.



Select **LPT Profile** option in the wizard window that opens. Then click **Next** button. In the next window, enter a name for the profile and click **Finish** to end the wizard.



After closing the wizard, profile page will appear on screen.



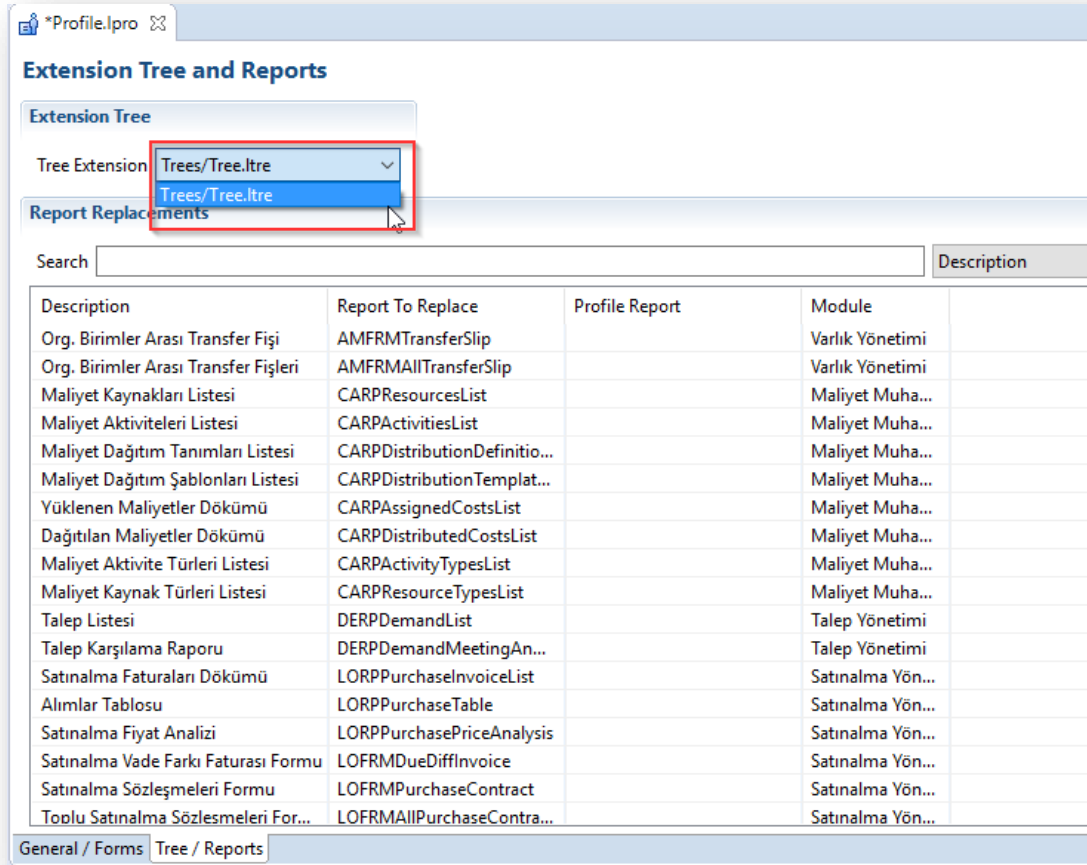
Profile page has all form objects in j-guar product. These forms can be changed with the customized forms.

Let's give an example. We made a customization for the AR/AP card under section [3.4.2](#). Now, let have this customized card appear on screen instead of the original card when a user opens AR/AP record. First, we find AR/AP card in the list and then double click the card. A pop up window that lists the existing customization forms opens. We select our AR/AP card from this list, click **Select** and close the pop up window.

The screenshot shows the 'Customization Profile Definition' window in the Logo Platform Tailor application. The 'Form Replacements' section is active, displaying a table with columns: Description, Form To Replace, Profile Form, Event Handler Class, and Module. The first row, 'Cari Hesap', is highlighted with a red box. A 'Replacing New Form' dialog box is open over the table, showing a list of forms. The form 'Forms/CFArpCard.lfrm' is selected and highlighted with a red box. The 'Select' button in the dialog is also highlighted with a red box.

Description	Form To Replace	Profile Form	Event Handler Class	Module
Cari Hesap	FIXFARPCard.jfm			Finans
Cari Hesaplar	FIXFARPCardBrowser.jfm			Finans
Teminatlar	FIXFARPCardCampaignPo...			
Form Basımı İçin Cari Hesap Girişi	FIXFARPCardEntryForFor...			
Cari Hesap Org.Bir. Parametreleri	FIXFARPCardOrgUnitPara...			
Cari Hesap Durum Değişikliği	FIXFARPCardStateChange...			
Cari Hesaplar - İlişkili Kayıtlar - Ha...	FIXFARPCardTransBrowse...			

You need to make tree definition for the profiles. To do so, click **Tree / Reports** tab at the bottom of the grid and open **Extension Tree and Reports** page. Here we select the tree we defined earlier in **Tree Extensions** field. Note that all reports in j-guar product are shown in the grid. These reports can be changed with custom reports.

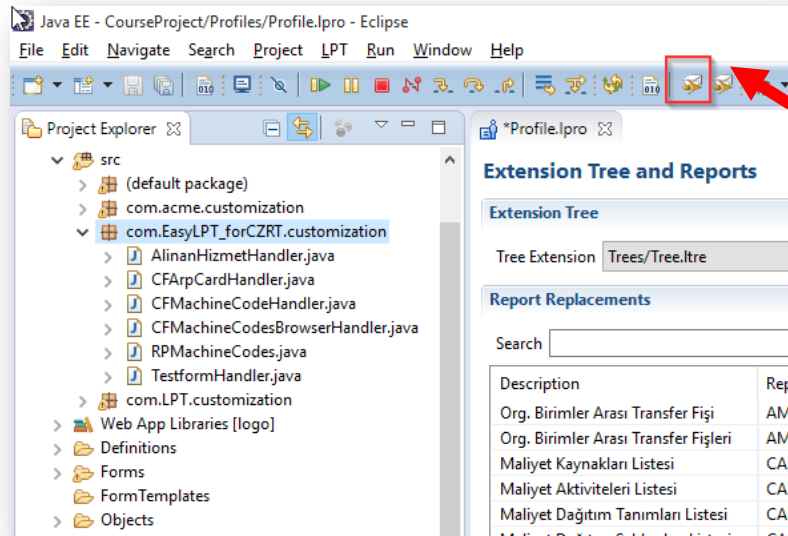


4. DEPLOYMENT

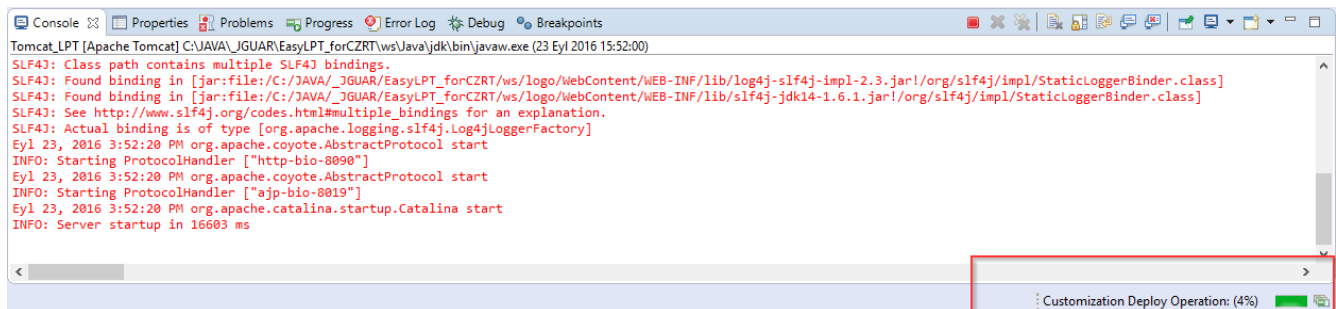
4.1. Deploying on Eclipse

Let us see how to deploy customizations to server.

To deploy over Eclipse, you must start by checking connection settings. See the details about establishing connection in section [2.3](#). After testing the connection, click the project over Package Explorer and activate it. Then click **Deploy** button.

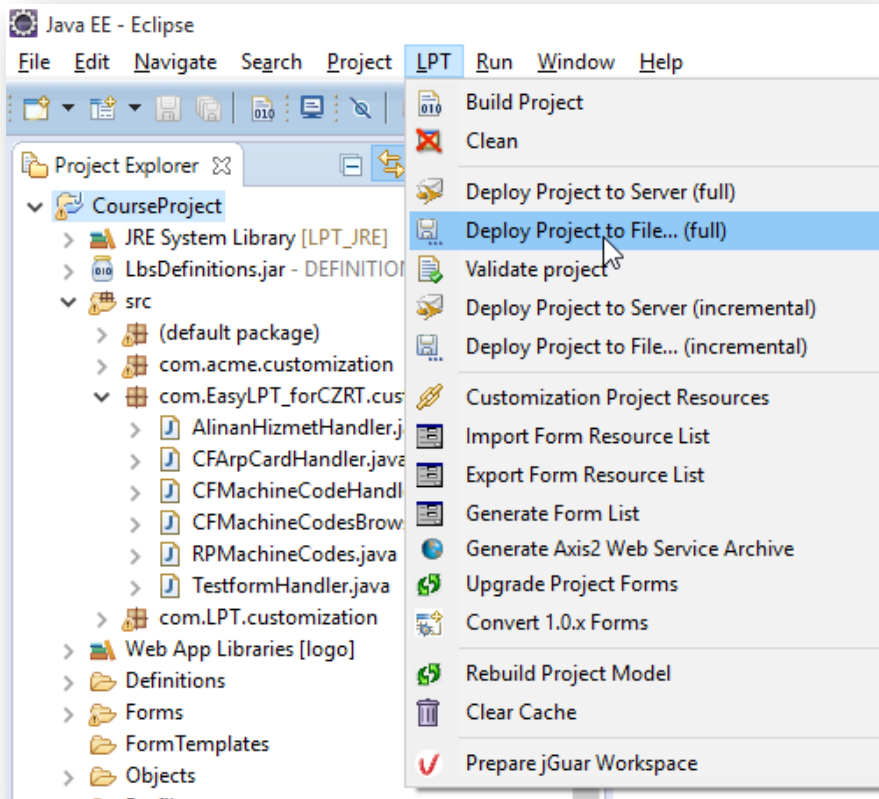


After starting Deploy operation, you can monitor the status of customization deploy to server by checking out the right bottom section of Eclipse.




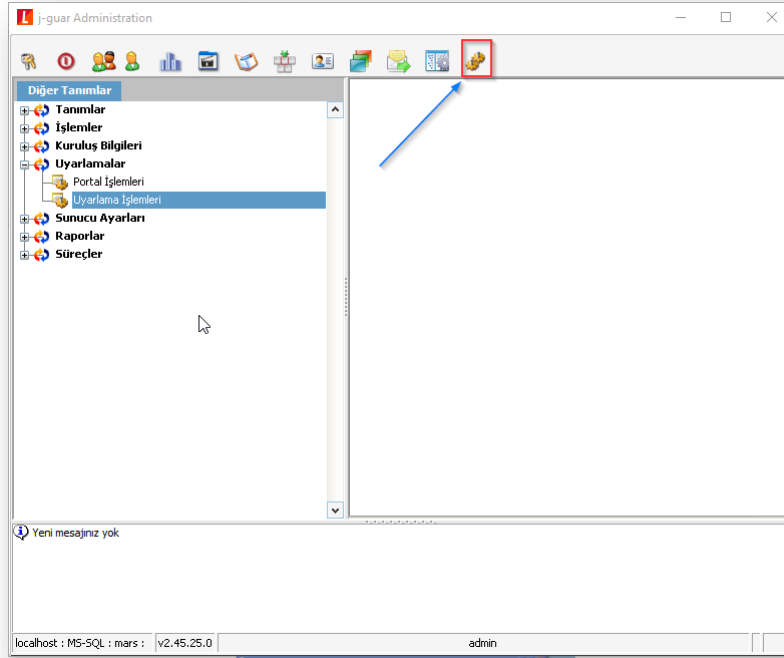
4.2. Deploying in the product

First, you need to create a customization file with extension **ladd** to deploy in the product. To do so, select **Deploy project to file... (full)** option under Logo menu of Eclipse.

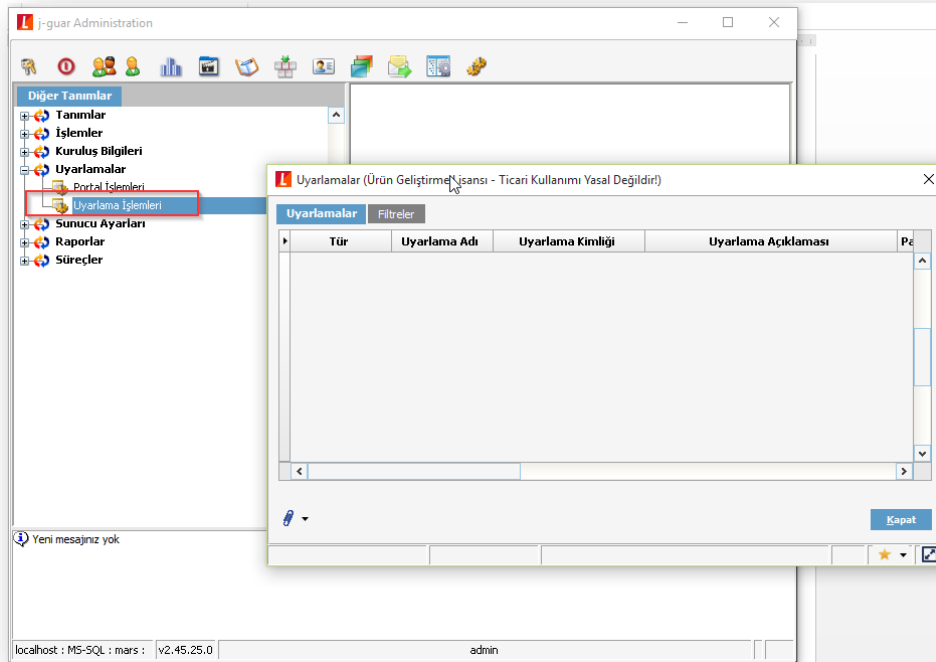


Designate the location to save Lpdd file. Note that **lpdd** file is named automatically.

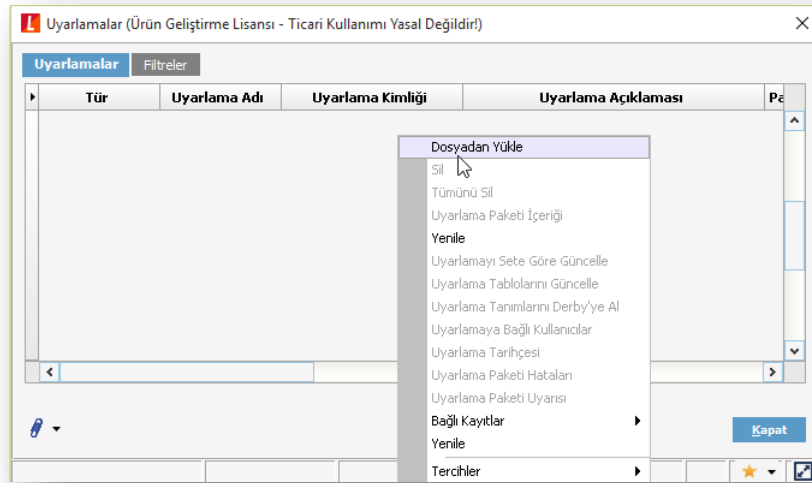
After the file with lpdd extension is generated, j-guar Administrator is opened. Click Other Definitions  button on the horizontal menu.



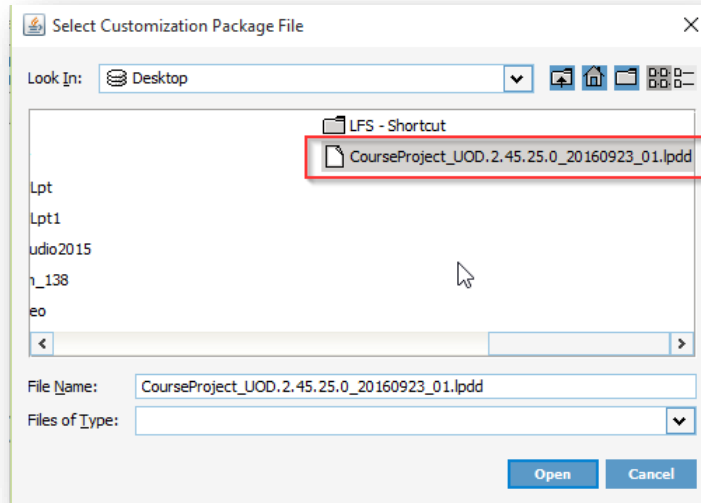
Click **Customizations > Customization Operations** option under Other Definitions tree. Customizations browser that lists the customization packages opens.



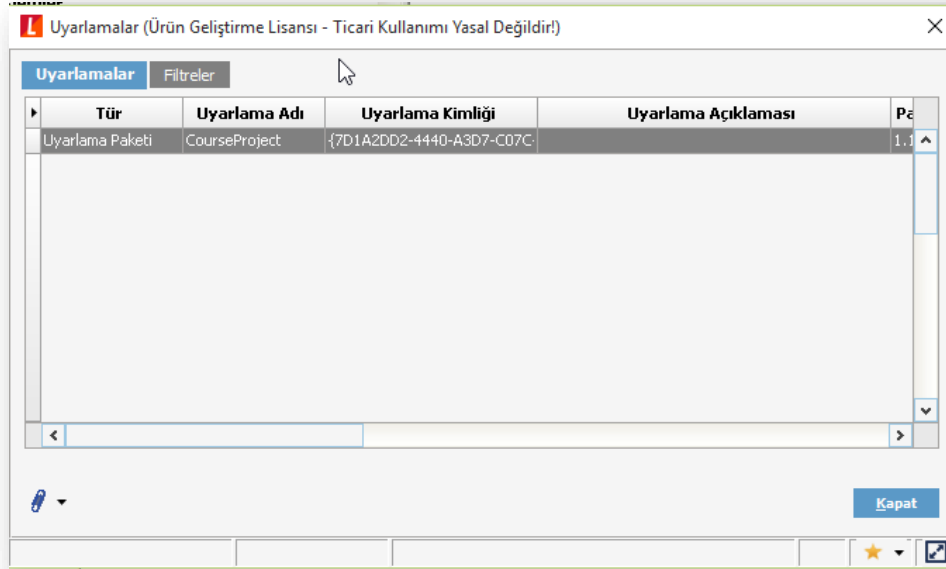
Right click on Customizations browser and select **Load from File** option.



Select the file with Ipdd extension and click **Open**.

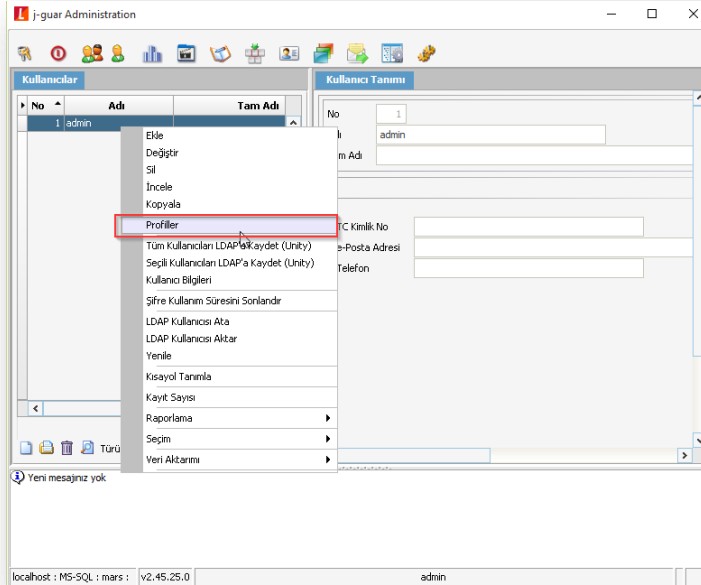


After a short while, customization file will appear in the browser. Thereby customization will be **deployed**.

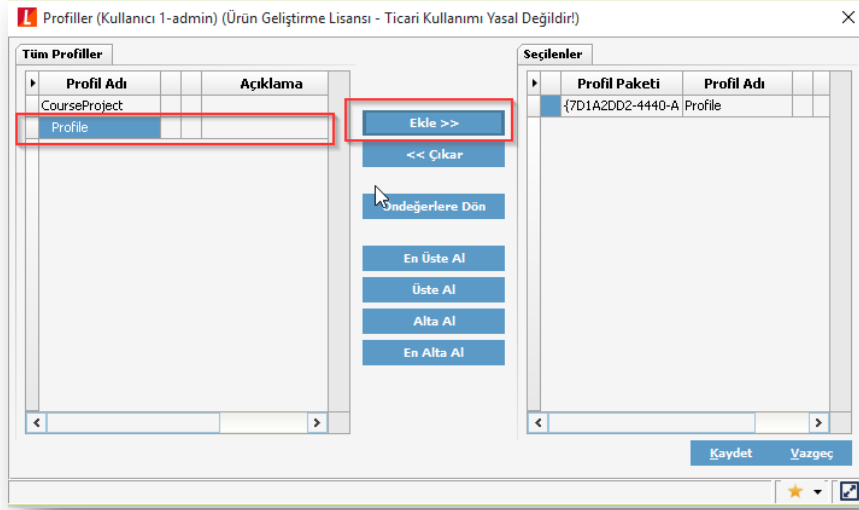



4.3. Operations after Deploying

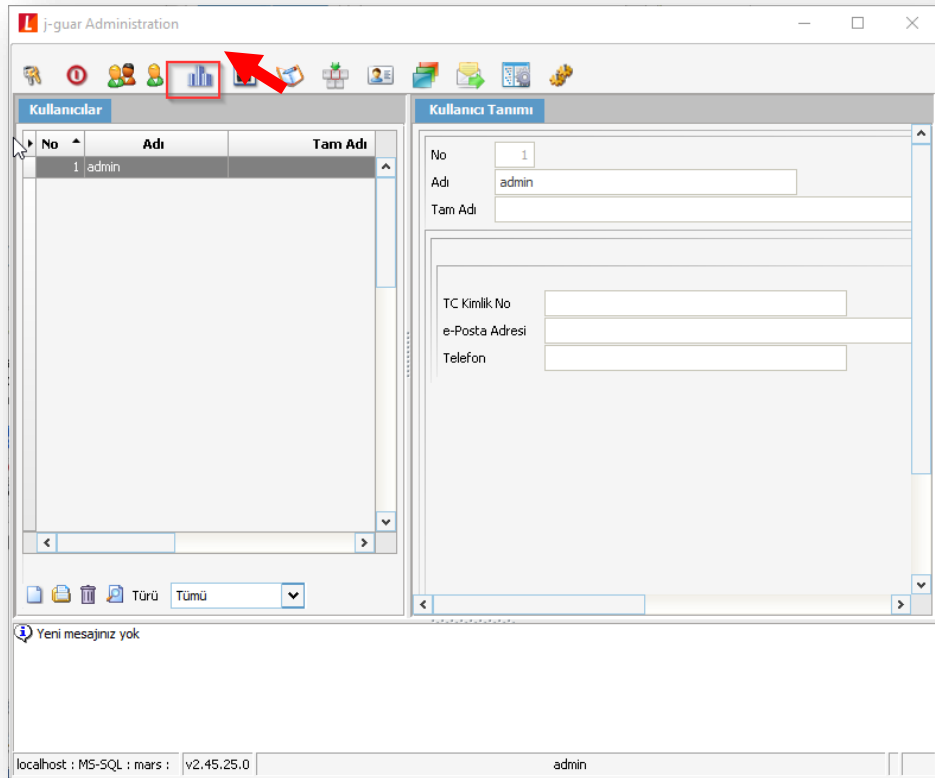
The operations you do after deploying are the same whether you deploy over Eclipse or in the product. First, you need to add the customizations you deployed to a user as profile. To do so, right click on Users and then click **Profiles** option.



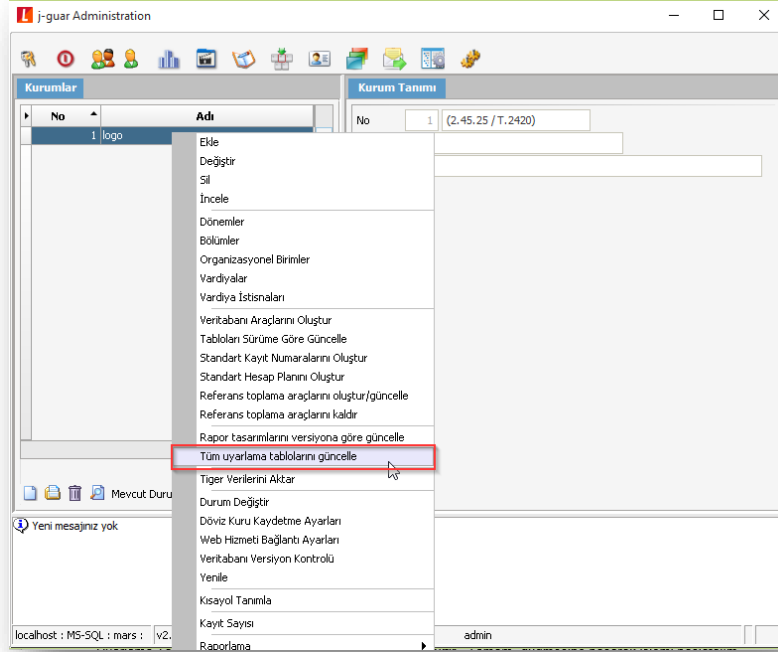
On the left pane of Profiles window, all customization profiles are listed. On the right pane, customization profiles that are linked to the selected user are listed. After selecting **CourseProject** profile on the left, we click **Add** button to add the profile to the user and click **Save**. Then we end profile addition.



If a new table is added as customization, it is not sufficient to perform deployment for this table to be created in database. After deploying, you must update customization tables. To do so, click **Companies**  button.



Right click on your company definition and select **Update All Customization Tables** option.



Message board at the bottom shows update status of the tables. All problems encountered when updating tables are also shown in this field.

