

**LOGO** **PLATFORM TAILOR**

**Ortam Tanıtımı**

---

## İÇİNDEKİLER

<b>1. ÇALIŞMAYA BAŞLARKEN .....</b>	<b>3</b>
1.1. Logo Platform Tailor kullanım amacı .....	3
<b>2. PROJE OLUŞTURMA .....</b>	<b>3</b>
2.1. Yeni proje oluşturma adımları.....	3
2.2. Proje Ayarları .....	5
2.3. Bağlantı ayarları .....	5
<b>3. PROJEDE YERALAN KLASÖRLER VE KULLANIMLARI.....</b>	<b>7</b>
3.1. Tablolar .....	7
3.2. Nesnelere .....	9
3.3. Sorgular .....	13
3.4. Formlar .....	17
3.5. Raporlar .....	25
3.6. Ağaçlar.....	29
3.7. Profil .....	32
<b>4. DEPLOYMENT.....</b>	<b>35</b>
4.1. Eclipse üzerinde deploye işlemi.....	35
4.2. Ürün içerisinde deploye işlemi .....	36
4.3. Deploye sonrası işlemler .....	38

## 1. ÇALIŞMAYA BAŞLARKEN

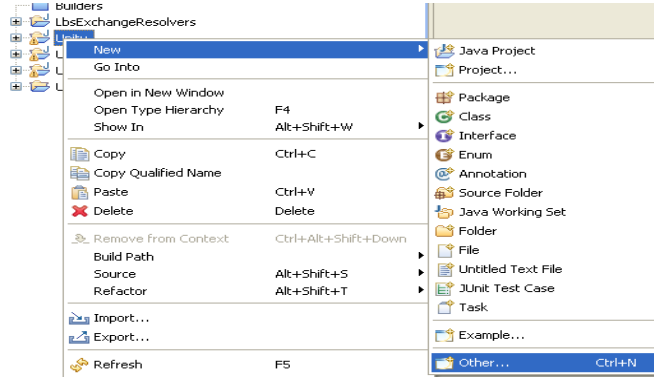
### 1.1. Logo Platform Tailor kullanım amacı

- *j-guar* ürününde kullanılan ağaç yapısı özelleştirilebilir, yeni modüller eklenebilir.
- *j-guar* ara yüzüne yeni formlar eklenebilir, mevcut formlar üzerine alanlar eklenebilir.
- Yeni sorgular eklenebilir.
- Yeni raporlar eklenebilir, mevcut raporlar genişletilebilir.
- *j-guar*'in başka sistemlerle entegrasyonu sağlanabilir. Bu amaçla standart Web servislerden yararlanabileceği gibi yeni Web Servisler de eklenebilir
- Excellent kullanımında projeye özel yeni metotlar eklenebilir.
- Yeni batch işlemler oluşturulabilmektedir.

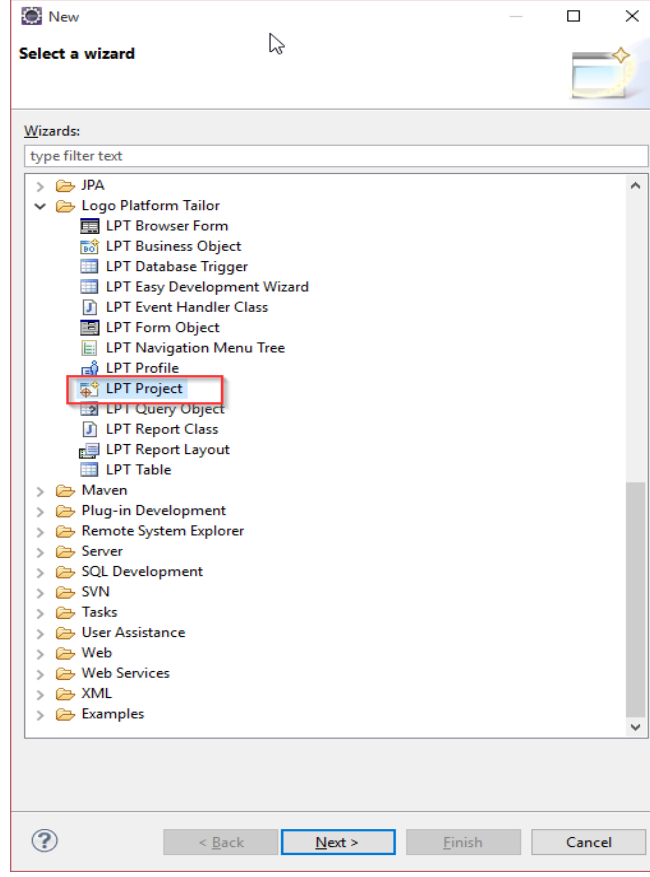
## 2. PROJE OLUŞTURMA

### 2.1. Yeni proje oluşturma adımları

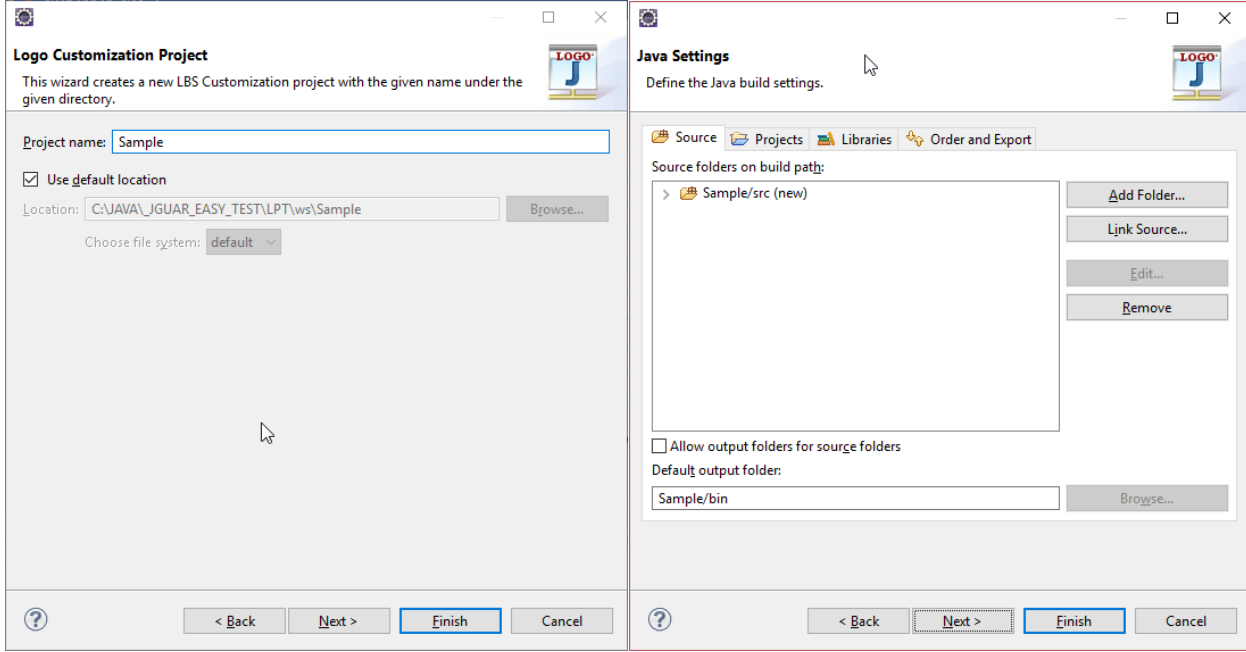
Uyarılama yapabilmek için ilk adım yeni bir uyarılama projesi oluşturmaktır. LPT'de proje oluşturmak için "Logo Customization Project" sihirbazı kullanılmaktadır. Package Explorer üzerinde sağ click ile açılacak olan listede önce "New" ardından da "Other" seçilir.

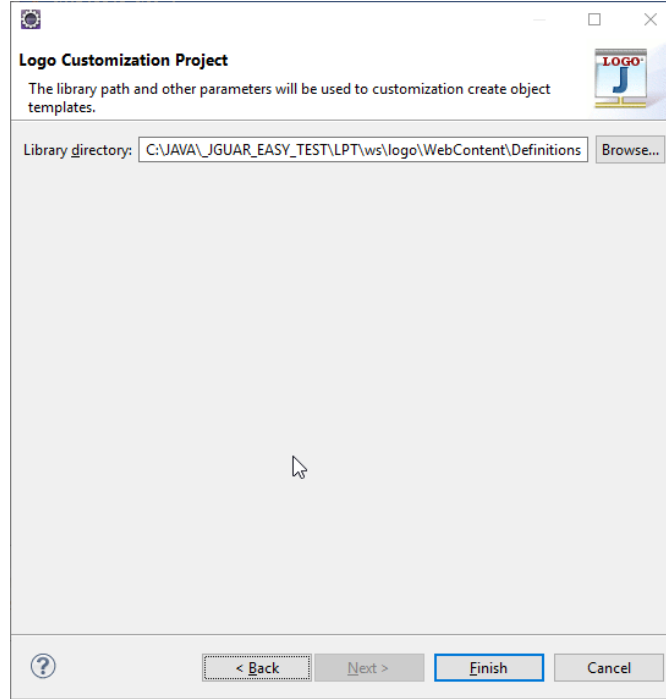


Gelen ekranda "Logo Platform Tailor" klasörünün altında uyarılama için kullanılacak sihirbazlar listelenmiştir, bu listeden proje oluşturma sihirbazının seçilmesi gerekmektedir.



Bu işlemin ardından sihirbaz üzerinde oluşturulmak istenilen uyarlama projesinin adı yazılarak devam edilir.

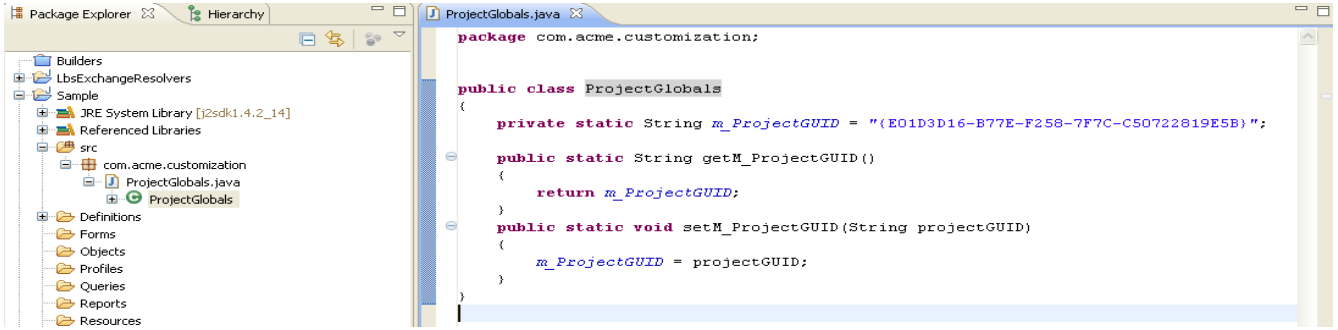




\* LbsDefinitions.jar J-GUAR ürününe ilişkin tüm nesne tanımlarını içeren ve versiyon bağımlı jar dosyasıdır.

## 2.2. Proje Ayarları

Project GUID alanında gördüğümüz değer proje'nin tekil kimliğini (unique ID) ifade etmektedir. Bu değere kod yazılırken ihtiyaç duyulacağı için proje içerisindeki SRC klasörünün içinde **"ProjectGlobals"** adlı bir dosyada tutulur.

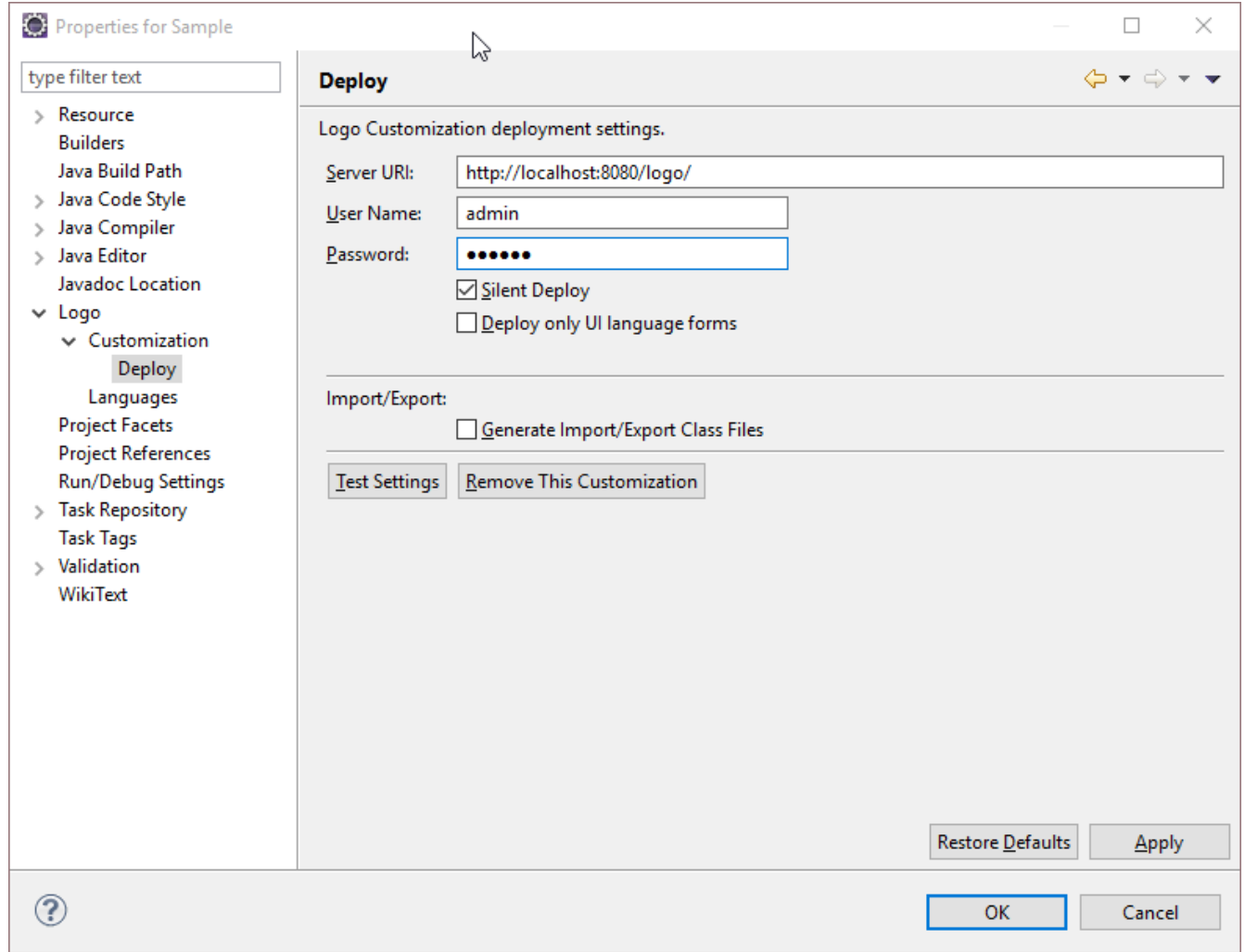


İleriki bölümlerde *ProjectGuid* kullanımına ilişkin örnekler yer almaktadır.

## 2.3. Bağlantı ayarları

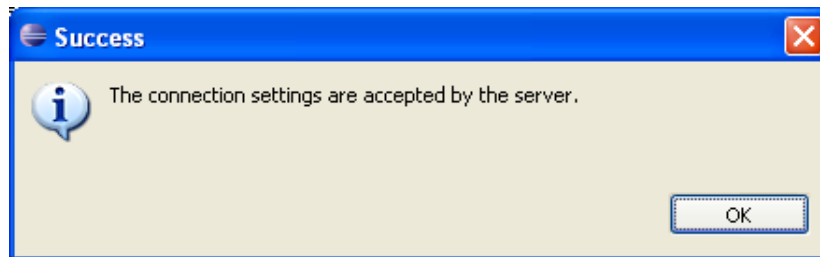
Uyarlama Projesinin, J-GUAR ürünüyle bağlantı kurulabilmesi amacıyla bağlantı ayarlarının yapılması gerekmektedir. Proje üzerinde sağ click yapılarak **"Properties"** seçilir, gelen proje özellikleri penceresinin solundaki **"Logo"** ve **"Customization"** seçenekleri genişletilerek **"Deploy"** özelliğine tıklanır. İlk açılışta bağlantı bilgileri boş olarak gelmektedir,

“Restore Defaults” düğmesi tıklanarak varsayılan bağlantı ayarları getirilebilir. Kullanıcı dilediği gibi bu ayarları düzenleyerek “Apply” düğmesini tıklar.



Ayrıca kullanıcı tanımlanan ayarlarla bağlantının kurulup kurulmadığını “Test Settings” düğmesine tıklayarak sınavabilir, bağlantının durumuyla ilgili kullanıcıya uyarı mesajı verilmektedir.

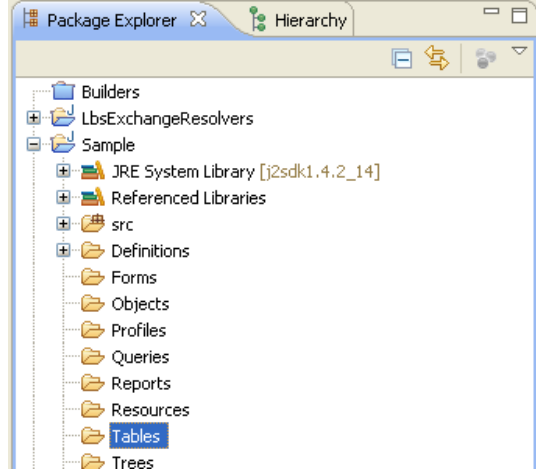
**Not:** Bağlantı ayarının sınavabilmesi için server’ın açık olması gerekmektedir.



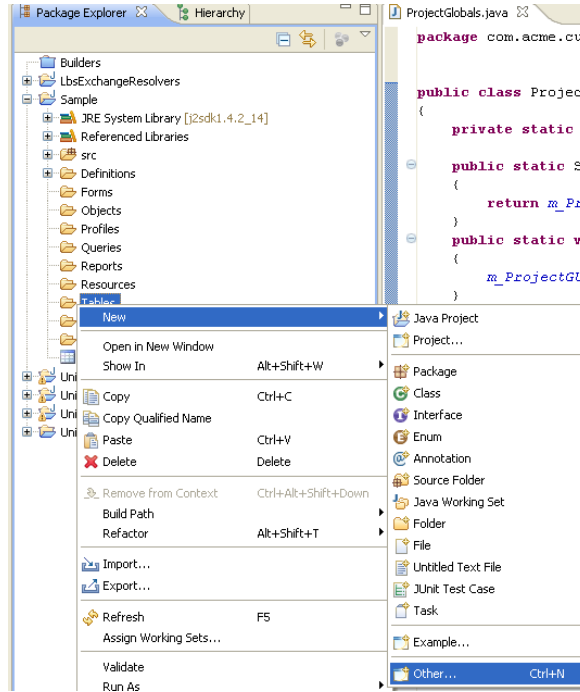
### 3. PROJEDE YERALAN KLASÖRLER VE KULLANIMLARI

#### 3.1. Tablolar

Tablolar uyarlama projesi içerisinde bulunan “Tables” klasöründe yer almaktadır.

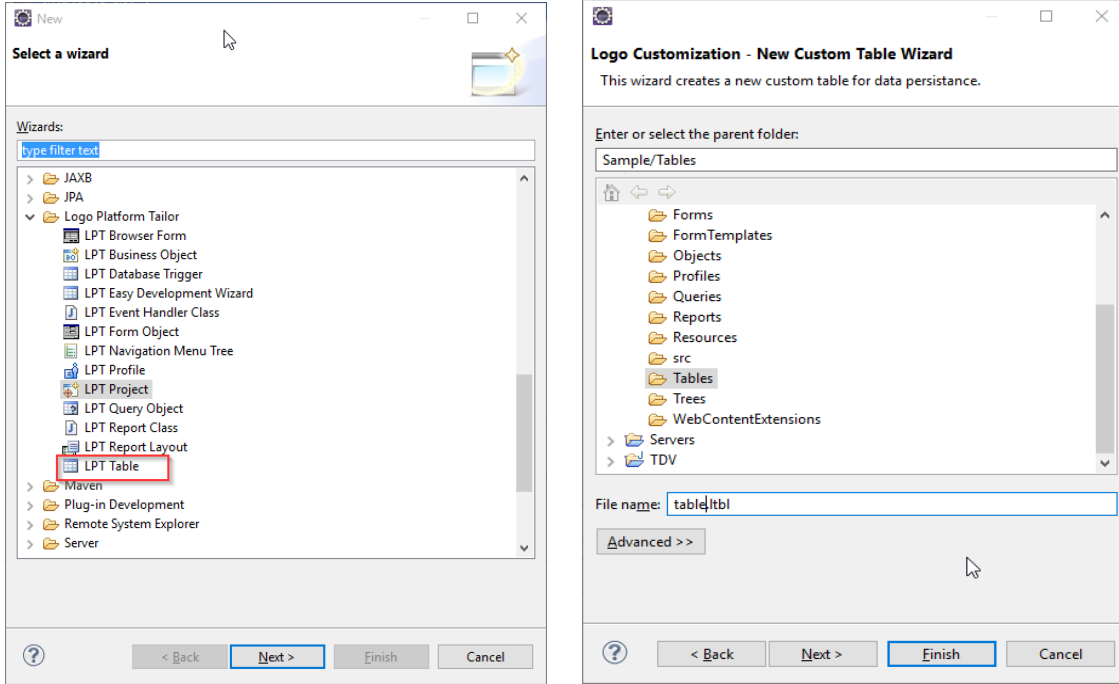


Tabloların nasıl yaratıldığını bir örnek üzerinde inceleyelim, örneğimizde makine tanımlamak için kullanılmak üzere tablo tanımı yapılmak istenmektedir, buna göre tables klasörü üzerinde sağ click yaparak menu’den “New” ardından da “Other”ı seçerek sihirbazların yer aldığı ekranı açalım.

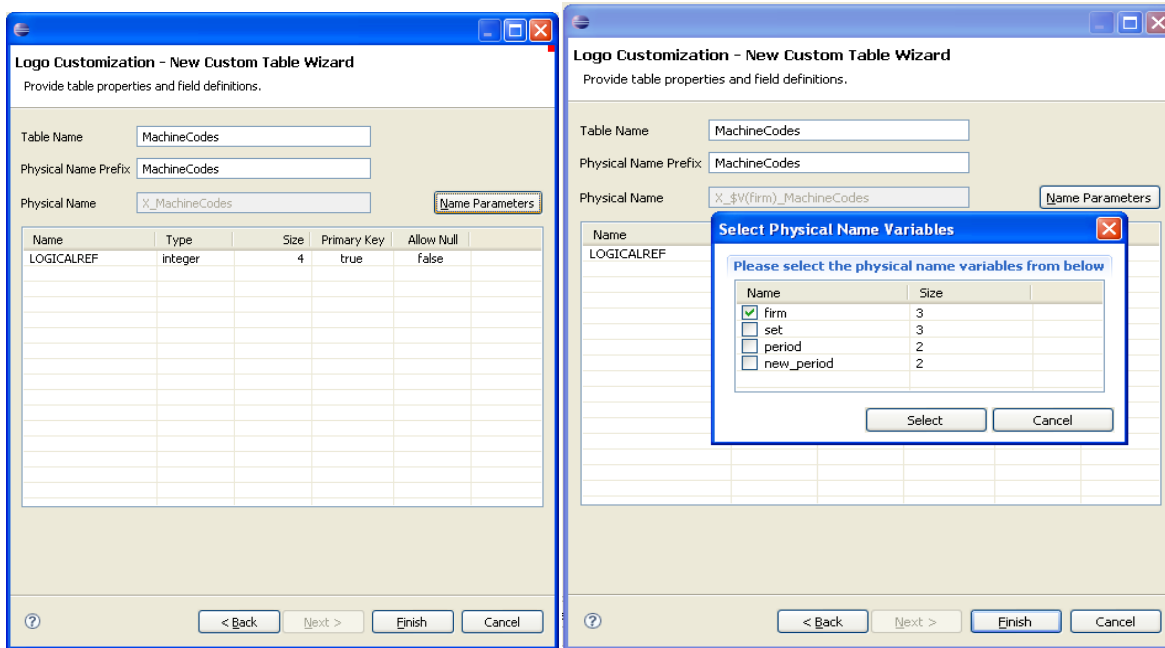


Açılan ekranda “Custom Table” seçilerek next düğmesi yardımıyla ilerleyelim gelen ekranda tablomuz için isim vermemiz gerekmektedir.

**Not:** “Bu aşamada verilen isim database üzerinde açılacak olan tablonun ismi değil sadece tablo tanımının ismidir”



Next düğmesi yardımıyla açılan sayfada database’de yaratılması istenilen tablonun ismi “**Physical Name Prefix**” alanına yazılır, tablonun firma ve/veya dönem tablosu olup olmadığına ilişkin bilgi “**Name Parameters**” düğmesine basılarak açılan küçük pencereden seçilir.



“**Logicalref**” kolonu j-guar ürününde tüm tablolarda yer alan primary key kolonudur, bu nedenle yaratılan her tablo için otomatik olarak oluşturulur.

Bizde “**Code**” ve “**Description**” isimli iki kolon ekleyip, tanımlamayı finish düğmesine basarak tablo oluşturma işlemini sonlandırılalım.



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

Table Name: MachineCodes  
Physical Name Prefix: MachineCodes  
Physical Name: X\_\$(firm)\_MachineCodes [Name Parameters]

Name	Type	Size	Primary Key	Allow Null
LOGICALREF	integer	4	true	false
CODE	string	20	false	false
DESCRIPTION	string	30	false	false

< Back Next > Finish Cancel

**Not:** "Tabloların database’de fiziksel olarak oluşabilmesi için j-guar Administrator’da uyarlama tablolarının sürüme göre güncellenmesi gerekmektedir."

### 3.2. Nesnelere

Formlar ve tablolar arasındaki iki yönlü veri hareketi nesnelere tarafından sağlanmaktadır. Bu cümleden yola çıkarsak Nesnelere tablolar ile formlar arasındaki katmanlı oluşturduğunu rahatlıkla söyleyebiliriz. j-guar ürününde nesnelere ikiye ayrılmıştır.

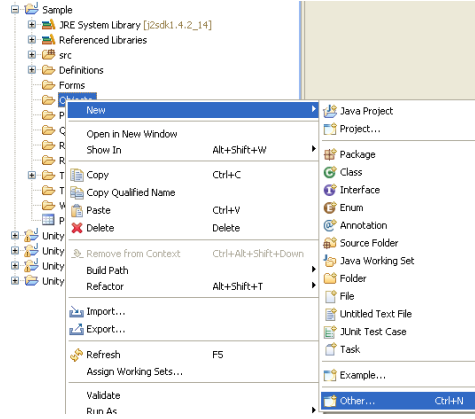
- Business Objectler
- Custom Business Objectler

İki nesne grubu arasındaki fark Business Objectler j-guar ürününe ait ürün nesnelere dir, Custom Business Objectler ise uyarlama yapan kişi tarafından oluşturulmuş yeni nesnelere dir.

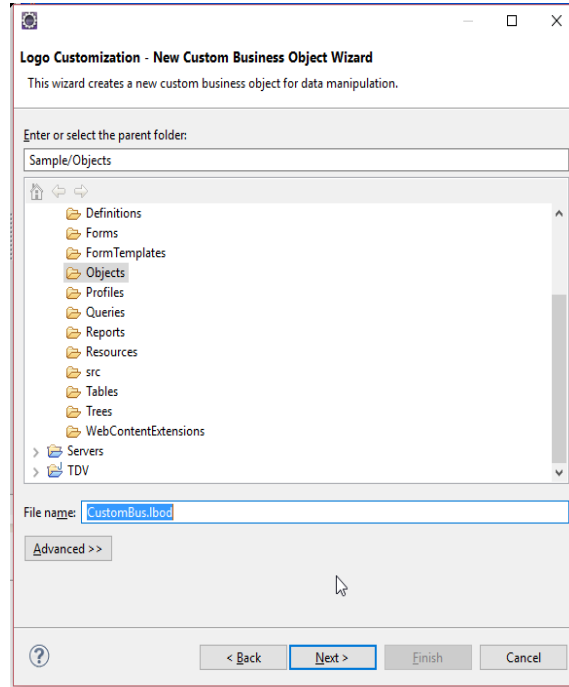
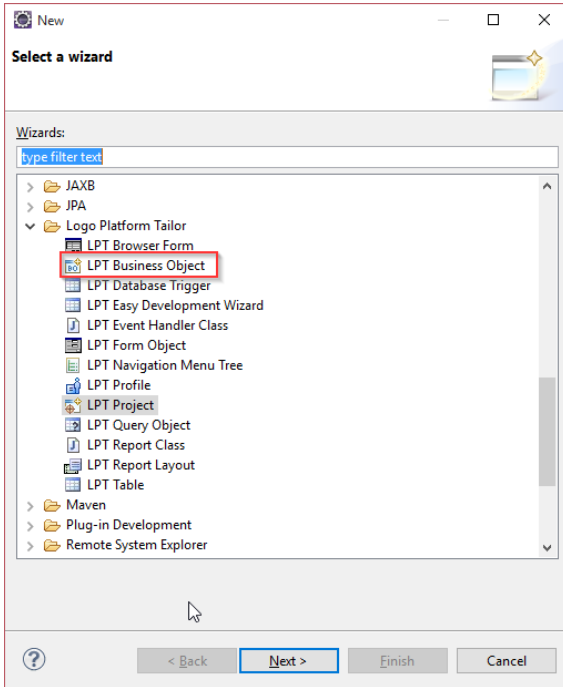
#### 3.2.1. Yeni bir nesne oluşturmak

3.1 bölümünde Makine kodları için bir tablo oluşturmuştuk şimdi bu tabloya ilişkin nesne tanımını yaparak konuyu örneklendirelim.

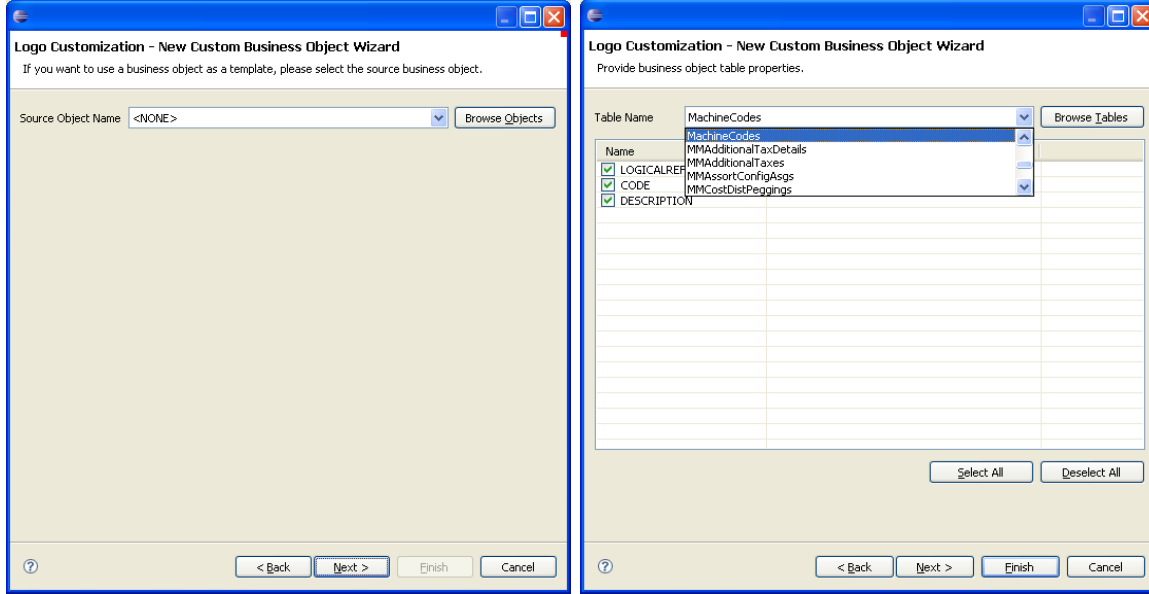
Objects klasörü üzerinde sağ click yaparak menu’den “New” ardından da “Other”ı seçerek sihirbazların yer aldığı ekranı açalım.



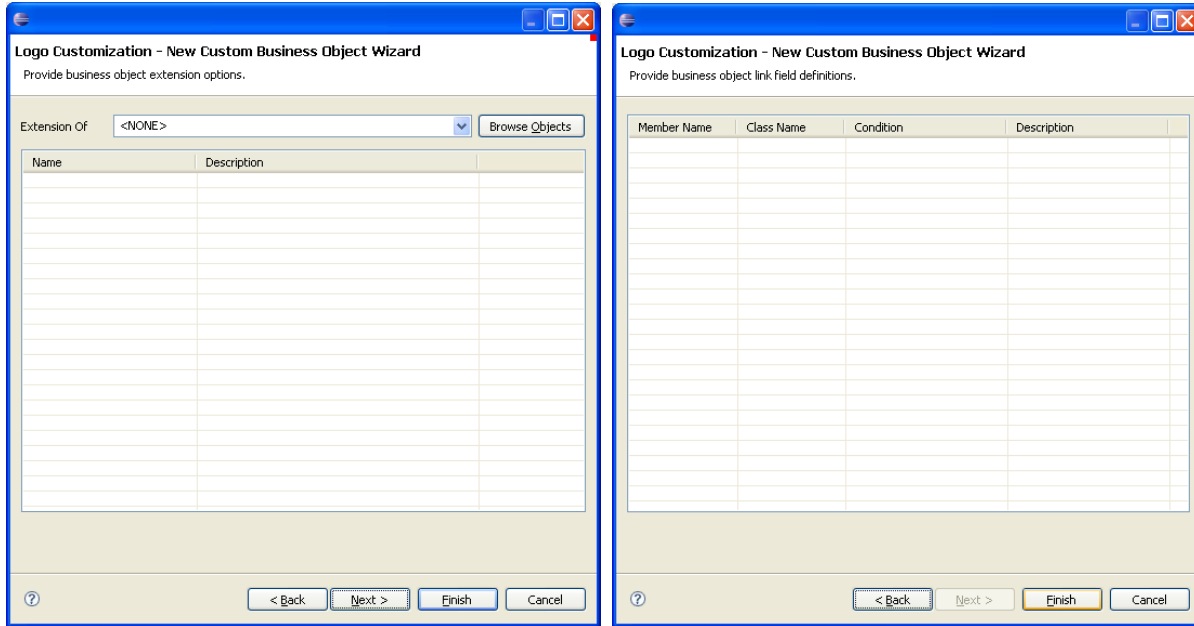
Uyarlama yapan kişi tarafından yeni oluşturulacak nesnelerin Custom Business object şeklinde olduğunu daha önce belirtmiştik. Bu nedenle seçimimizi LPT Business Object olarak yaparak “Next” düğmesine tıklayalım. Gelen ekranda bizden yaratmak istediğimiz nesneyi isimlendirmemiz istenecek, isimlendirmeyi yaparak “Next” düğmesine tıklayarak devam edelim.



Gelen Sayfada herhangi bir seçim yapmadan “Next” düğmesine basarak ilerleyelim. Şimdi bize yaratmak istediğimiz nesnenin hangi tabloyla bağlantılı olduğu sorulmaktadır. Bu aşamada 3.1 nolu madde yaratmış olduğumuz MachineCodes tablosunu seçelim, bu seçimin ardından ekrandaki grid’e tabloda bulunan kolonlar getirilecektir. Yaratmış olduğumuz nesneye dâhil olmasını istemediğiniz kolonlardaki check işaretini kaldırabilirsiniz yapmadan “Next” düğmesine basarak ilerleyelim



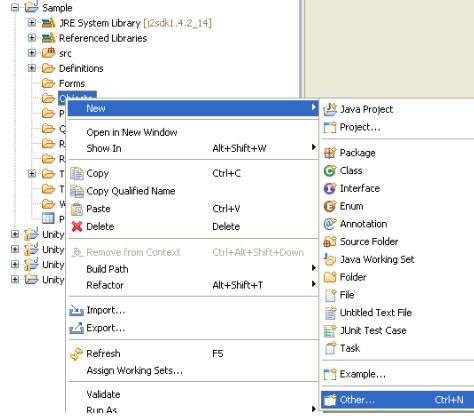
Gelen Sayfada herhangi bir şey yapmadan “Next” düğmesine basarak ilerleyelim. Sihirbazın son sayfasına ulaştık, bu sayfa üzerinde de hiçbir şey yapmadan” finish “düğmesine basarak sihirbazı sonlandıralım.



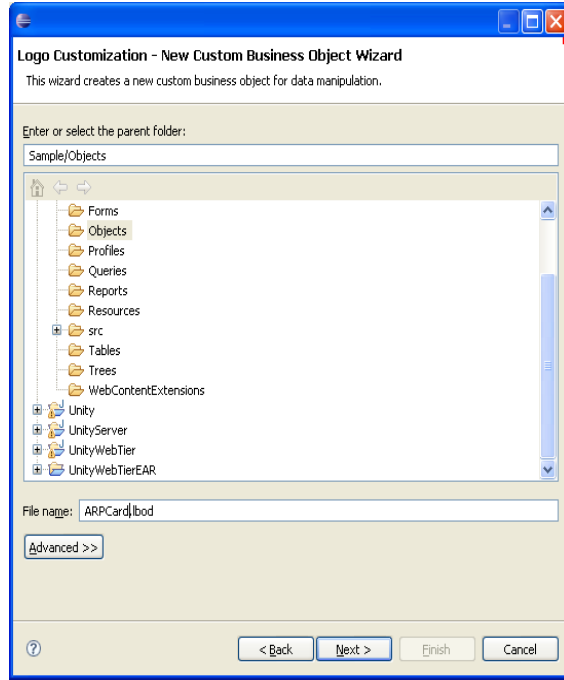
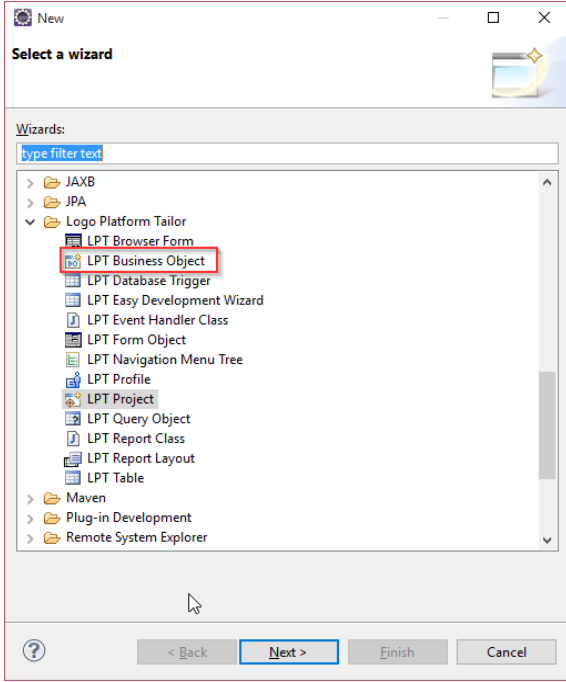
### 3.2.2. Mevcut bir ürün nesnesini genişletmek

Bu örneğimizde de cari hesap kartı nesnesine alan ekleyerek nasıl genişletebileceğimizi inceleyelim.

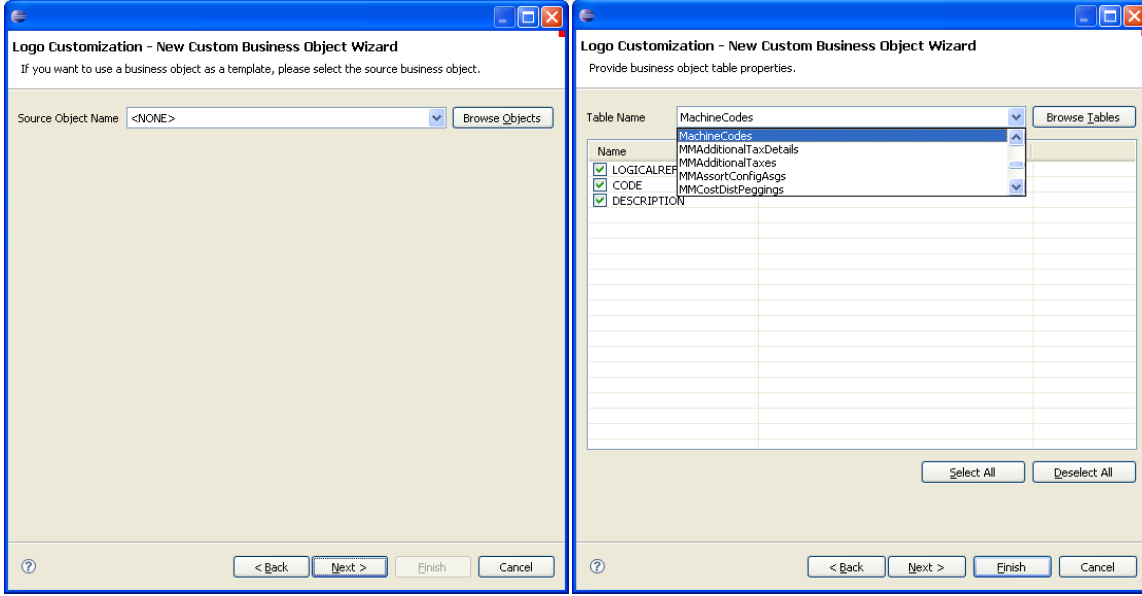
Objects klasörü üzerinde sağ click yaparak menu’den “New” ardından da “Other”ı seçerek sihirbazların yer aldığı ekranı açalım.



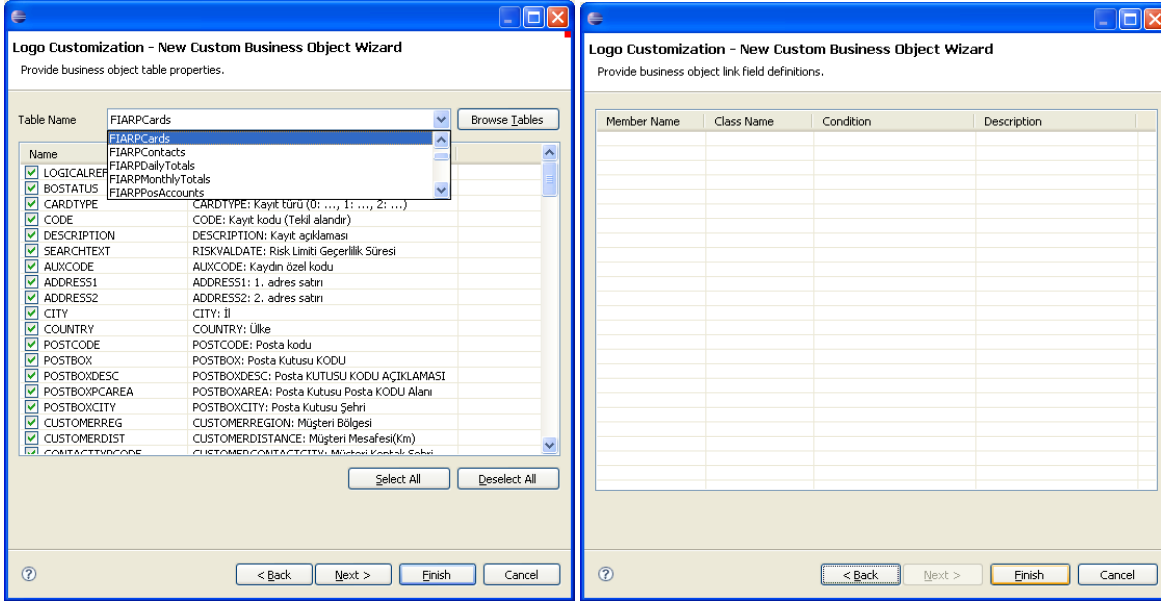
Bir önceki örneğimizde olduğu gibi yine LPT Business Object seçiminde bulunarak “Next” düğmesine basıyoruz. Gelen ekranda nesnemizi isimlendirerek “Next” düğmesine basarak bir sonraki sayfayı açıyoruz.



Gelen ekranda herhangi bir işlem yapmadan “Next” düğmesine basıyoruz. Şimdi nesneye bağlamak istediğimiz tabloya ilişkin seçim yapıyoruz. Bu seçimin ardından tabloya ilişkin kolonlar ekrandaki grid’e gelmektedir. “Next” düğmesine basarak devam ediyoruz.

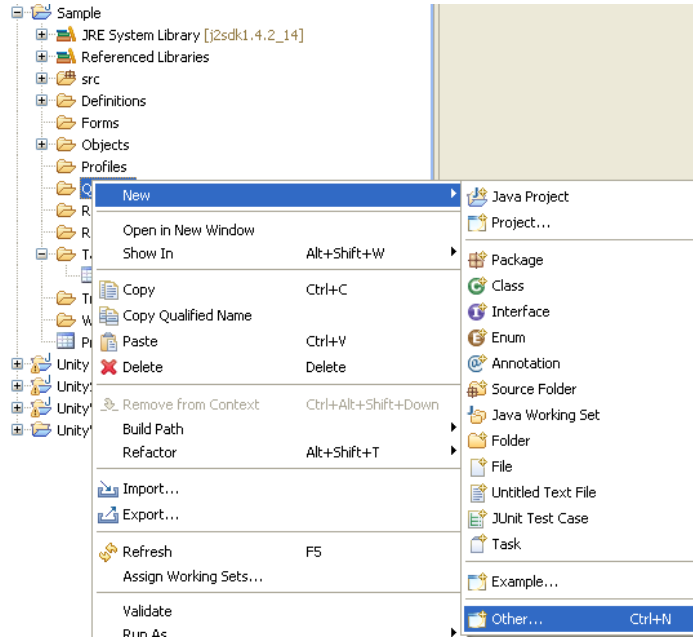


Bu sayfa da genişletilmesini istediğimiz Business Object'i seçiyoruz örneğimizde cari hesap kartı nesnesi yani FIBOARPCard. Seçimin ardından "Next" düğmesine basarak sihirbazın son sayfasına geliyoruz ve "Finish" düğmesine basarak sihirbazı sonlandırıyoruz.

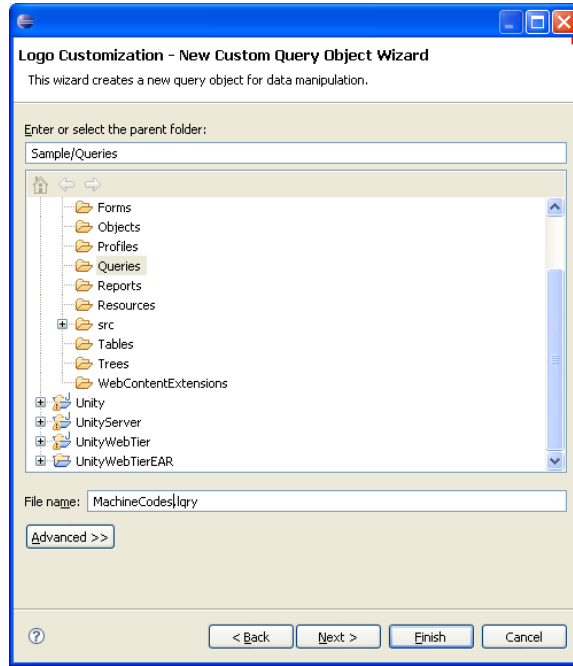
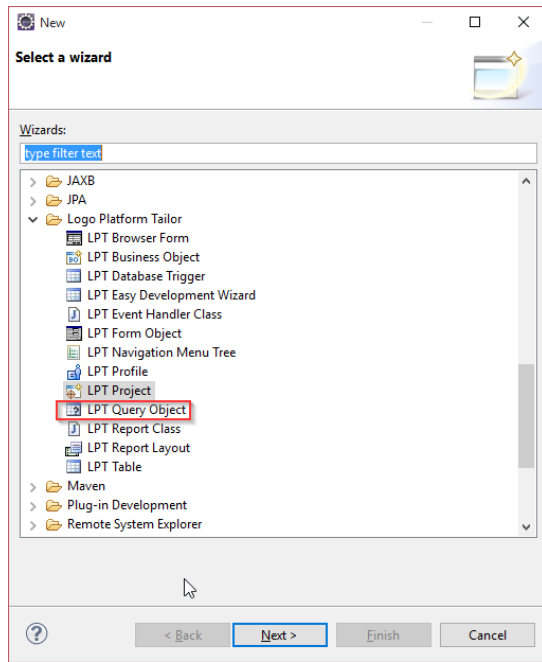


### 3.3. Sorgular

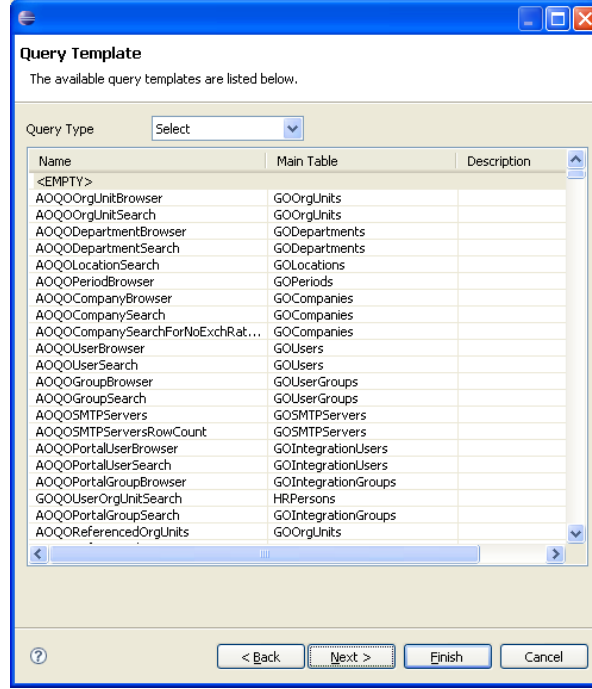
Uyarılma query nesnelere (Query Business Object) "Queries" klasörü içerisinde yer almaktadır. Yeni bir query nesnesi oluşturmak için sağ click yaparak menu'den "New" ardından da "Other"ı seçerek sihirbazların yer aldığı ekranı açalım.



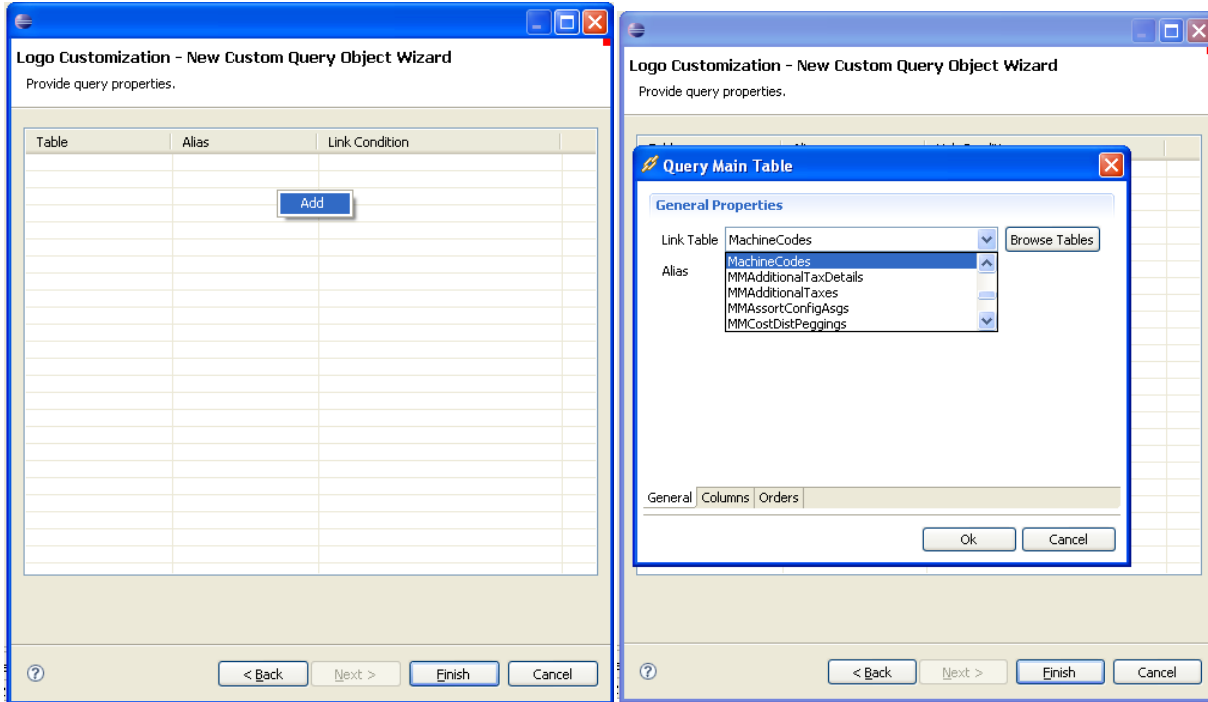
Açılan ekranda LPT Query Object seçeneğini seçerek “Next” düğmesine basalım, açılan sayfada bizden query nesnesi isimlendirmemiz beklenmektedir, isimlendirme işlemini yaptıktan sonra “Next” düğmesine basalım.



Gelen ekranda mevcut queryler listelenmektedir, bu queryler den birini şablon olarak kullanabildiğimiz gibi yeni bir query de oluşturabiliriz, bunun için “<Empty>” seçeneği seçili olarak “Next” düğmesine basalım.



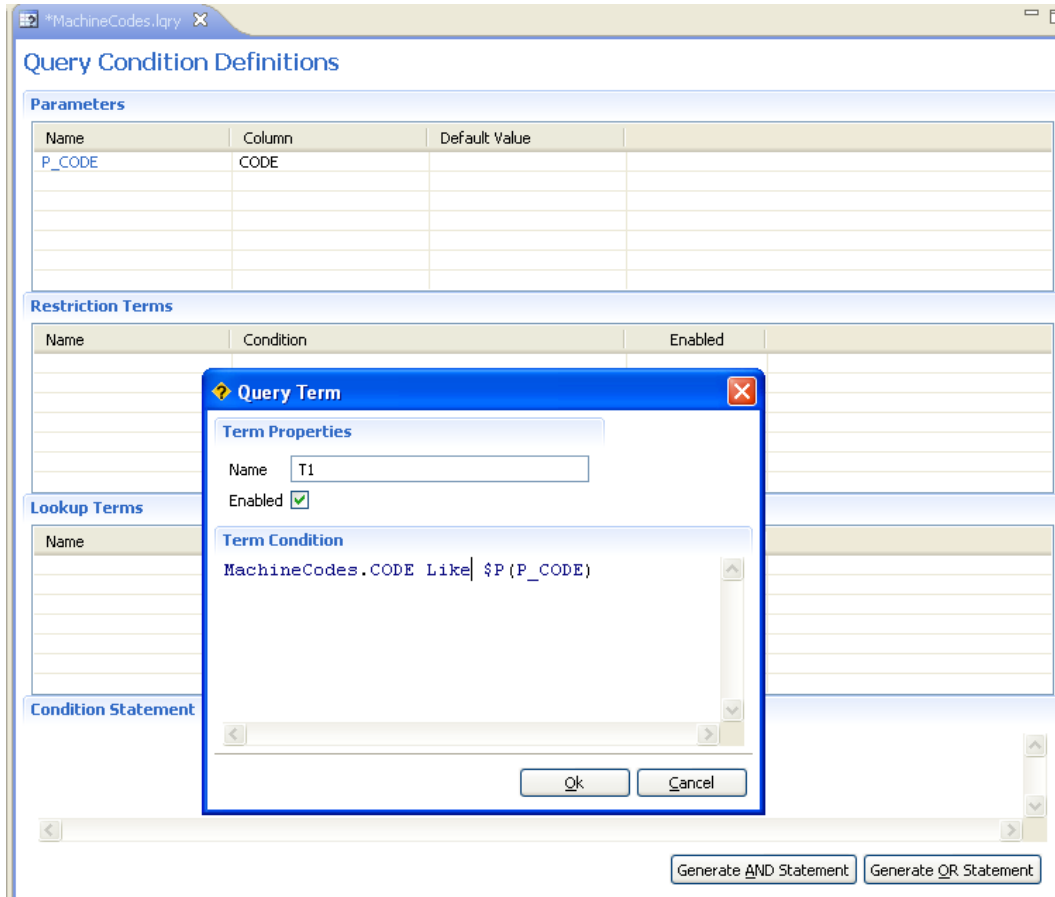
Gelen ekranda boş bir grid bulunmaktadır, sağ click yaptığımızda gelen menu'de "Add" seçili iken click'leyelim. Tablo seçebileceğimiz bir pop up gelecektir. Combobox'dan "MachineCodes" tablosunu seçelim. Ve "Ok" düğmesine tıklayarak pop up'ı kapatalım.



Son sayfada seçmiş olduğumuz tablo görüntülenecektir. "Finish" düğmesine basarak sihirbazı kapatalım.





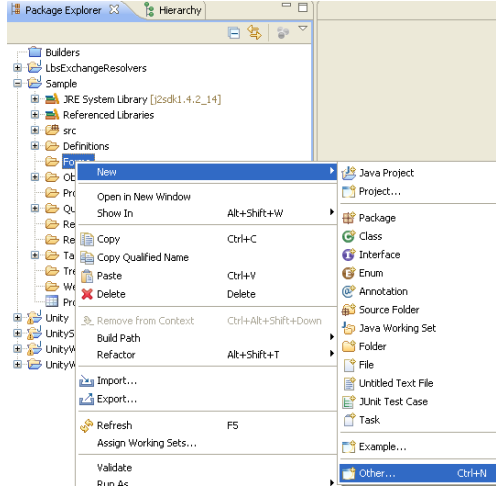


### 3.4. Formlar

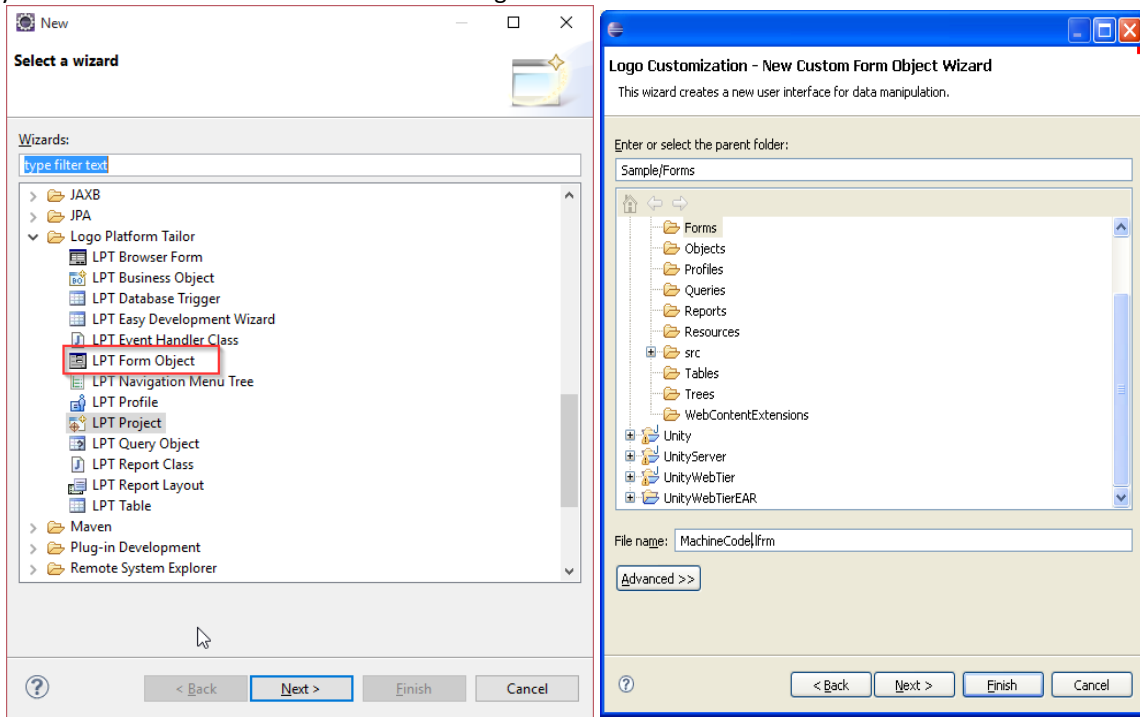
#### 3.4.1. Yeni form oluşturmak

Formlar uyarlama projesi içerisinde yer alan “**Forms**” klasörünün içerisinde bulunmaktadır. j-guar kartların ve browser’ların yer aldığı bir yapıya sahiptir, sırasıyla kart ve browser yapısı anlatılmaya çalışılacaktır.

Forms klasörü üzerinde sağ click yaparak menu’den “**New**” ardından da “**Other**”ı seçerek sihirbazların yer aldığı ekranı açalım.



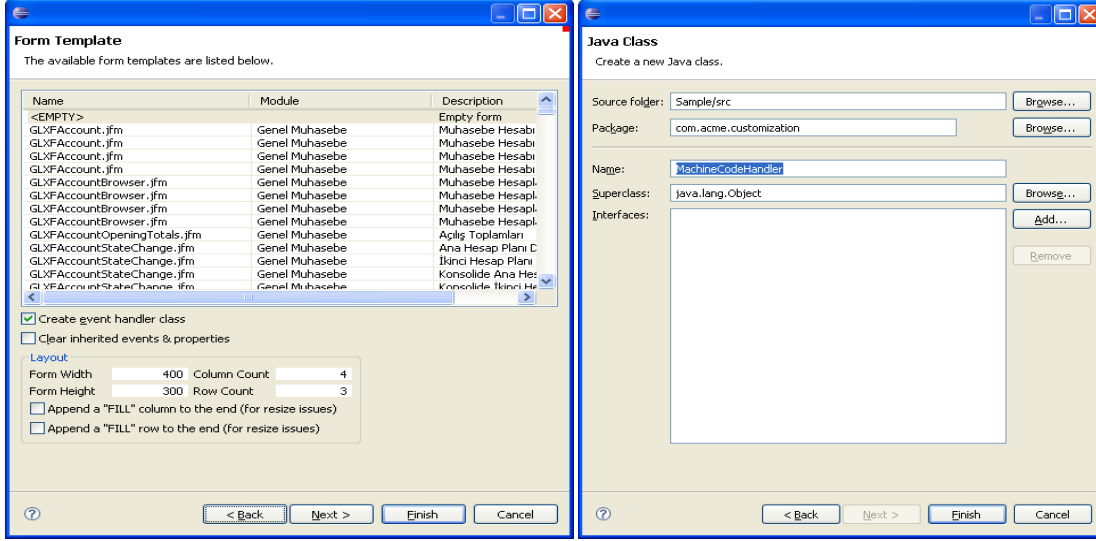
Sihirbazın ilk sayfasında “LPT Form Object(Table Layout)” seçeneği işaretlenerek “Next” düğmesine basılarak ilerlenir gelen sayfada form nesnesi isimlendirilerek “Next” düğmesine basılır.



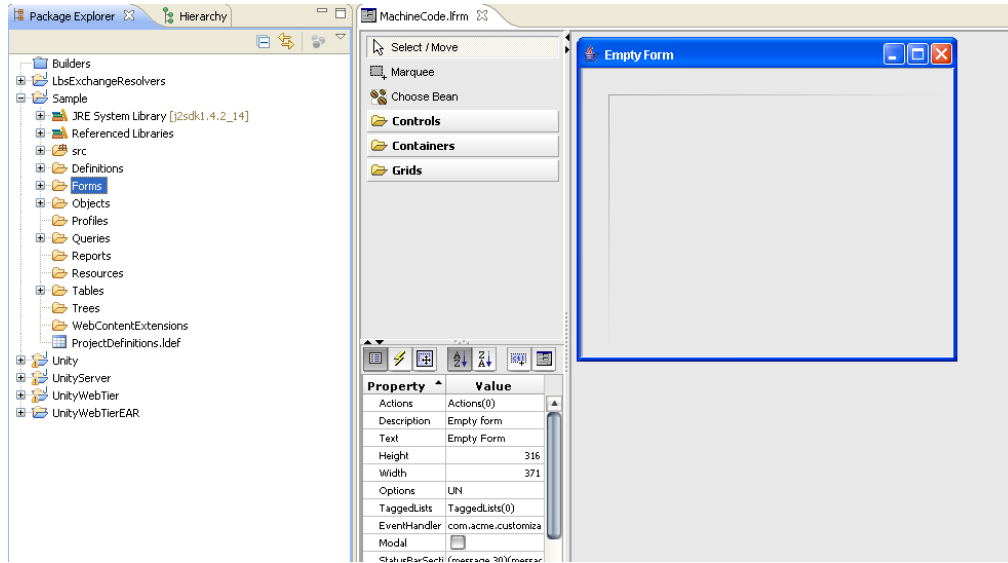
Sihirbazın gelen sayfasında, yaratılmak istenilen form için şablon kullanılmak istenip istenmediği seçilir (Form şablonları “LbsDefinitions.jar” dosyasında bulunmaktadır ), eğer boş bir ekran açılması isteniyorsa ön değer olarak gelen “<Empty>” seçeneği değiştirilmemelidir. Bu ekran üzerinde yer alan seçeneklerden bahsetmek gerekirse,

- **Create event handler class:** Bu seçenek form için kod yazılmasına olanak sağlayacak olan class dosyasının otomatik olarak yaratılmasına olanak sağlamaktadır. Ön değer seçili olarak gelmektedir, yani class dosyası yaratılır.
- **Clear inherited events & properties:** Eğer şablon kullanarak form yaratıyorsanız, yani ürünün formlarından birini uyarlıyorsanız, ilgili formun arkasında ki ürün kodları otomatikman yaratmış olduğunu bu yeni formda da çalışıyor olacaktır, eğer bu bahsedilen ürün kodlarının çalışması istenilmiyor ise bu seçenek işaretlenir.

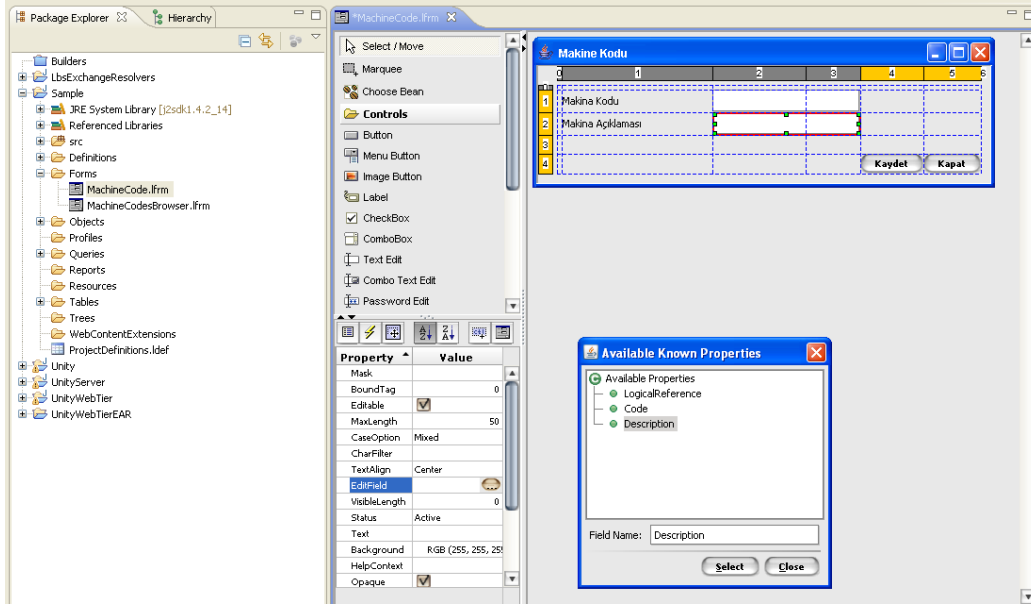
“Next “ düğmesine basılarak ilerlenir, “form için class dosyası yarat” seçeneği işaretli olduğu için yaratılacak olan class’ın isimlendirilmesi istenir, Bu işlem aynı zamanda sihirbazın son adımını oluşturmaktadır. “Finish” düğmesine basılarak sihirbaz sonlandırılır.



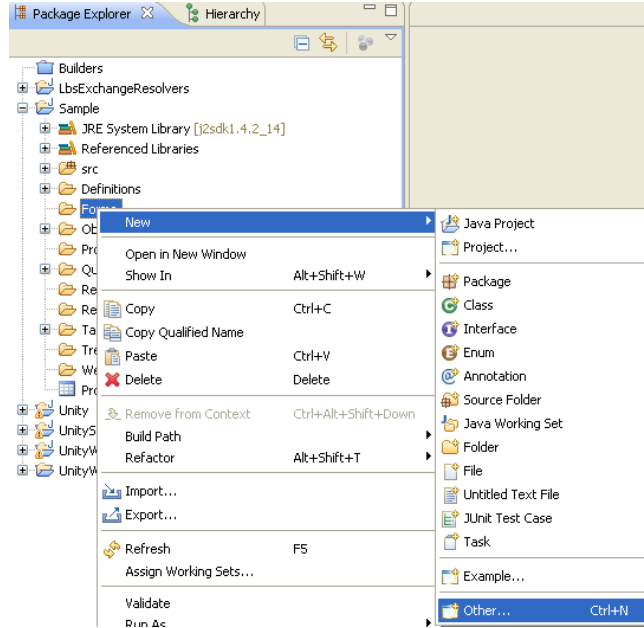
Oluşturulan form boş olarak ekrana gelir. Formun sol tarafında kullanılabilecek kontrollere ilişkin bir liste bulunmaktadır. Bu liste içerisinde yer alan kontroller sürükleyip bırakarak form’a yerleştirilebilmektedir.



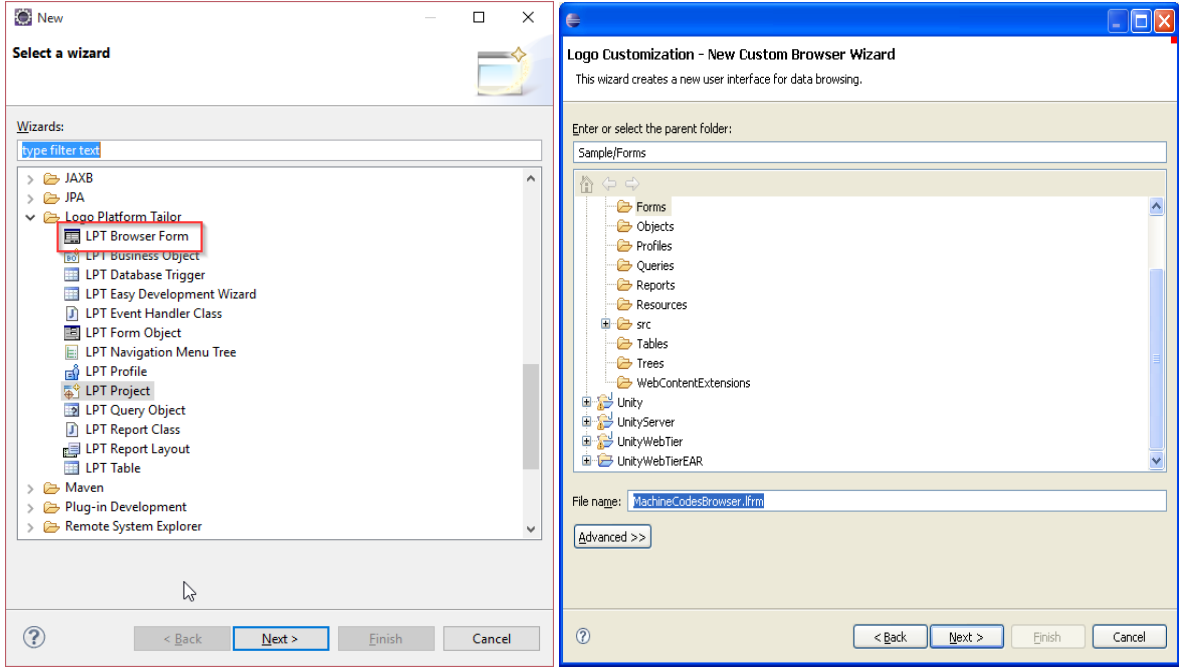
Makine kodları örneğimize devam ederek bu örnek için gerekli kart ve browser’ı yaratalım. Önce form’a 3.2.1 nolu madde de oluşturduğumuz nesneyi bağlayalım. Bunun için form dışında kalan bir yere tıklayalım sol tarafta gelen özellikler tabında “Edit Object Class” özelliğine “MachineCodes” nesnesini seçelim. Şimdi form’a iki adet label koyalım ve isimlendirelim. İki adet textbox koyalım ve özellikler tabından “editfield” özelliğine nesne üzerinde yer alan code ve description bilgilerini seçelim. İki adet düğme koyalım ve düğmeleri “kaydet” ve “kapat” şeklinde isimlendirelim. “Kaydet” düğmesi için özellikler tabında “Action” özelliği için “Save & Close” “Kapat” düğmesi için özellikler tabında “Action” özelliği için “Close” özelliğini seçelim. Kart’ımız böylelikle tamamlanmış olacaktır.



Şimdi Browser'ımızı oluşturmaya başlayalım, Forms klasörü üzerinde sağ click yaparak menu'den "New" ardından da "Other"ı seçerek sihirbazların yer aldığı ekranı açalım.



Gelen sihirbaz ekranı üzerinde "LPT Browser Form" özelliğini seçerek "Next" düğmesine basalım. Gelen ekranda browser'ın isimlendirmesini yaparak "Next" düğmesine basarak ilerleyelim.



Gelen Ekrandan;

- **Browser Title** alanına Başlık girilir.
- **Query Name** alanına daha önceden tanımladığımız query'i seçeriz
- Query Seçme işlemi sonrası tablomuzdaki tanımlı alanlar listelenir **select all** butonuna tıklarız
- **Select DBEntryDataClass** alanına daha önce tanımladığımız BO muzı tanımlarız
- **Select Object Class** alanına daha önce tanımladığımız BO muzı tanımlarız.
- **Select DBEntryFormClass** alanına Browser üzerinden açmak istediğimiz formun seçerek next butonuna tıklarız.
- Gelen ekran üzerinden Browserda görünmesini istediğimiz alanları seçeriz ve Finish butonuyla bitiririz.

**not:** hepsinin seçilip browser design-time üzerinden istenilen alanların kaldırılması daha sağlıklı olduğundan hepsini seçip Finish butonuna tıklarız.

**Logo Customization - New Custom Browser Wizard**  
Provide browser query options.

Browser Title:  1

Query Name:  2

Search:  Alias:

Alias	Column Name	Description
<input checked="" type="checkbox"/> LOGICALREF	LOGICALREF	
<input checked="" type="checkbox"/> col1	col1	
<input checked="" type="checkbox"/> col2	col2	

3

Identify Filter options  
 Create Event Handler Class

Select DBEntryDataClass:  4

Select ObjectClass:  5

Select DBEntryFormClass:  6

7

**Logo Customization - New Custom Browser Wizard**  
Provide browser filter and search options.

Query Name:

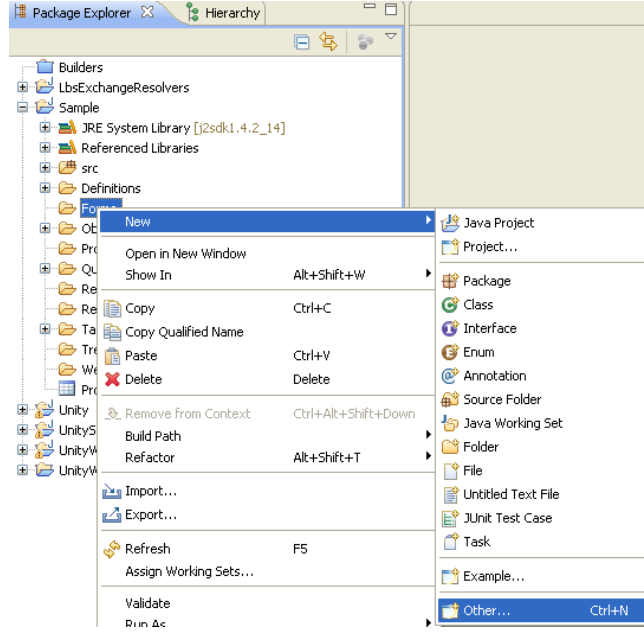
Search:  Filter Column:

Filter Column	Type	Generate Type
<input checked="" type="checkbox"/> LOGICALREF	Normal	Data-Bound
<input checked="" type="checkbox"/> col1	Normal	Data-Bound
<input checked="" type="checkbox"/> col2	Normal	Data-Bound

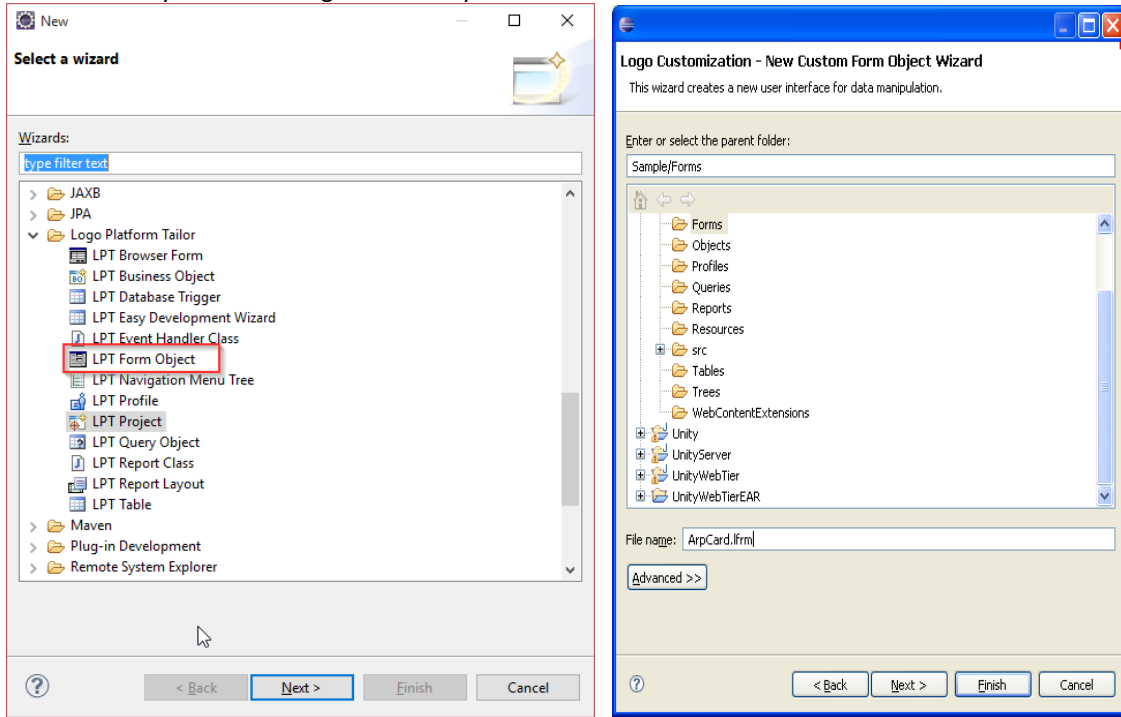
### 3.4.2. Mevcut ürün formunu uyarlamak

Konuyu örneklendirerek devam edelim, nesneler bölümünde cari hesap nesnesini genişletmiştik, şimdi bu nesneyi kullanarak cari hesap kartını uyarlayalım.

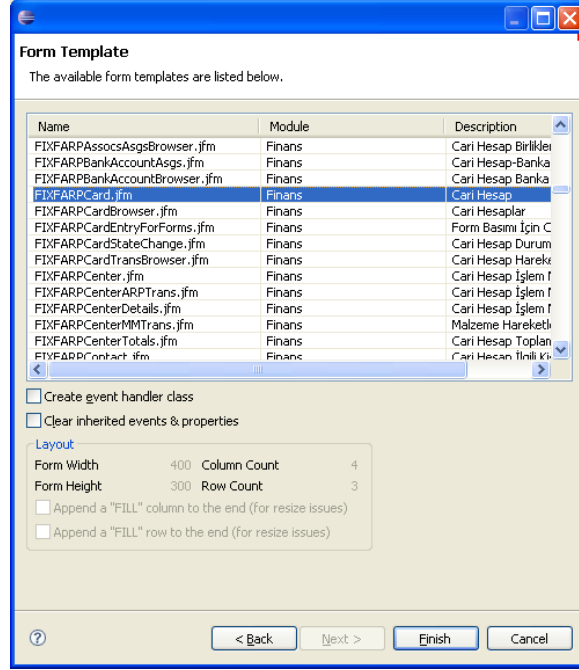
Forms klasörü üzerinde sağ click yaparak menu'den "New" ardından da "Other"ı seçerek sihirbazların yer aldığı ekranı açalım.



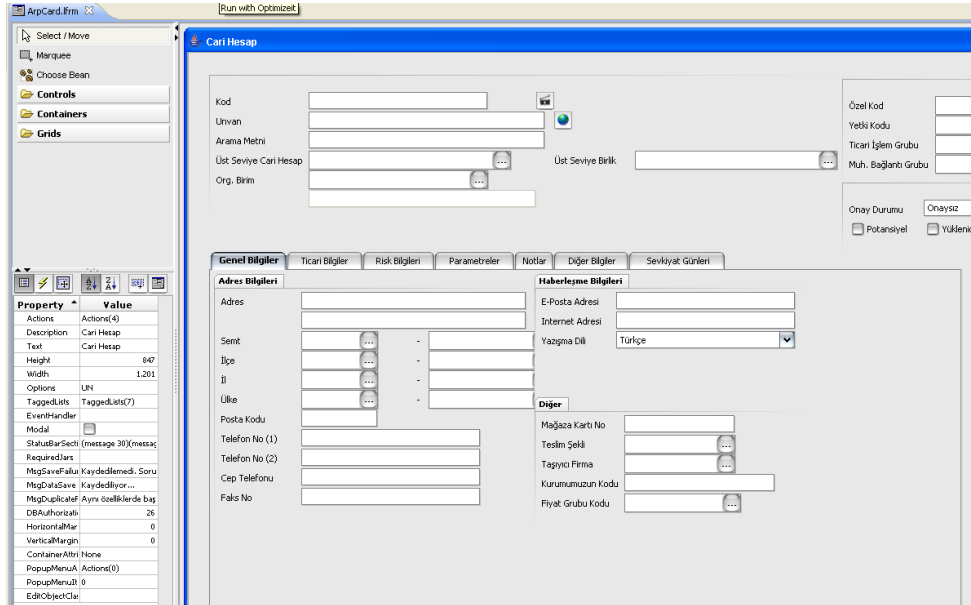
Açılan sihirbaz ekranında “LPT Form Object” seçili iken “Next” butonuna basarak ilerleyelim, gelen sayfada formumuzu isimlendirerek yine “Next” düğmesine basıyoruz.



Açılan sayfada uyarlamak istediğimiz ürün formunu seçiyoruz, örneğimiz için bu cari hesap kartı (FIXFARPCard.jfm) “Finish” düğmesine basarak sihirbazı sonlandırıyoruz.



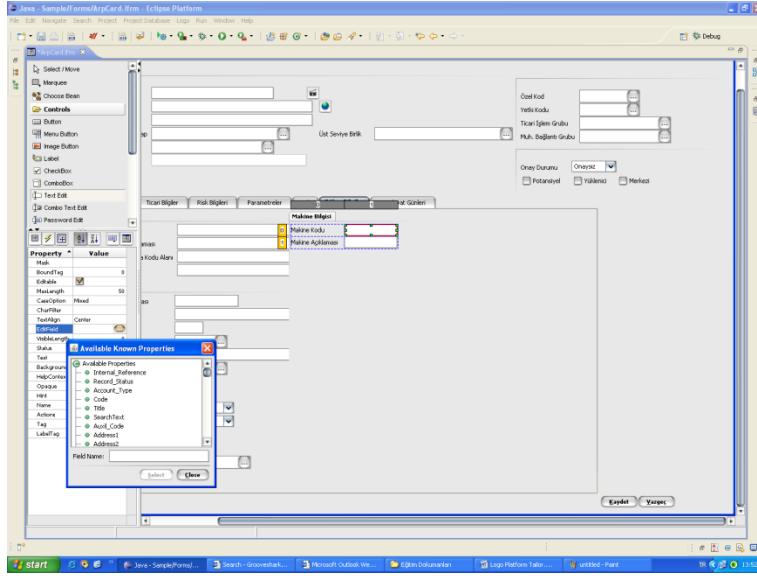
Şihirbazın sonlandırılmasıyla ürüne ait cari hesap kartını görüntüleyorz.



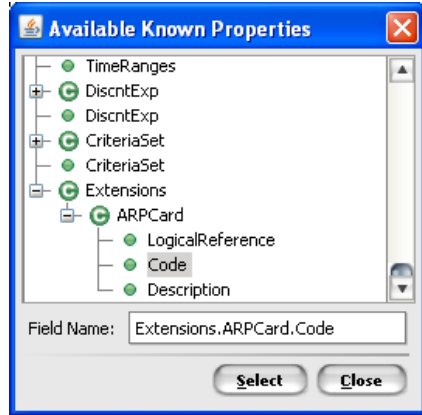
- Cari hesap kartında bulunan diğer bilgiler tabına makine kodu ve makine adı için label ve textbox'ları ekleriz.
- Nesnemizdeki alaları textbox'ımıza bağlarız.



- Textbox'ümüzü seçtikten sonra özellikler kısmındaki **"EditField"** alanına tıklayıp açılan popup ekranında nesnemize ait elemanların geldiğini görürüz.

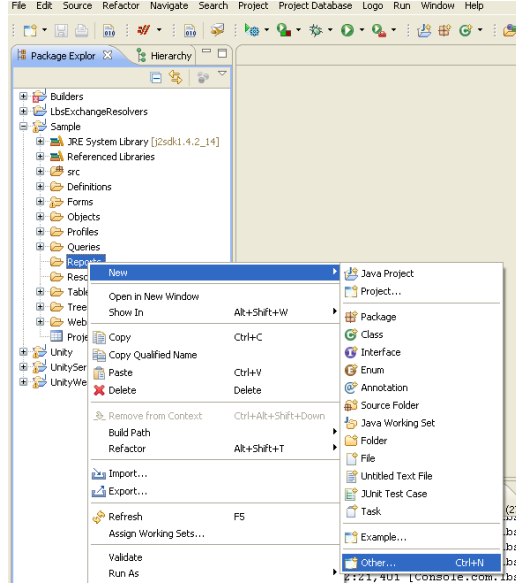


- Nesnenin en alt bölümünde **"Extensions"** yani uyarlama ile dâhil edilmiş elemanlar gelmektedir. İşte sırasıyla bu elemanları formda uyarlama ile eklediğimiz alanlara bağlayarak sonlandırıyoruz.



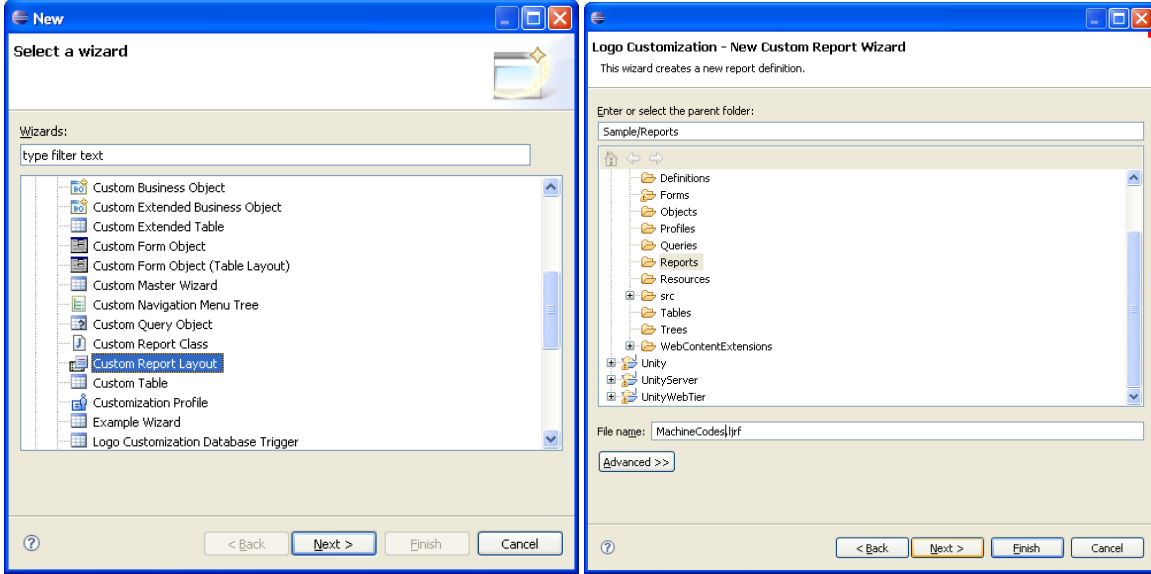
### 3.5. Raporlar

Reports klasörü üzerinde sağ click yaparak menüden **"New"** ardından da **"Other"**ı seçerek sihirbazların yer aldığı ekranı açalım.

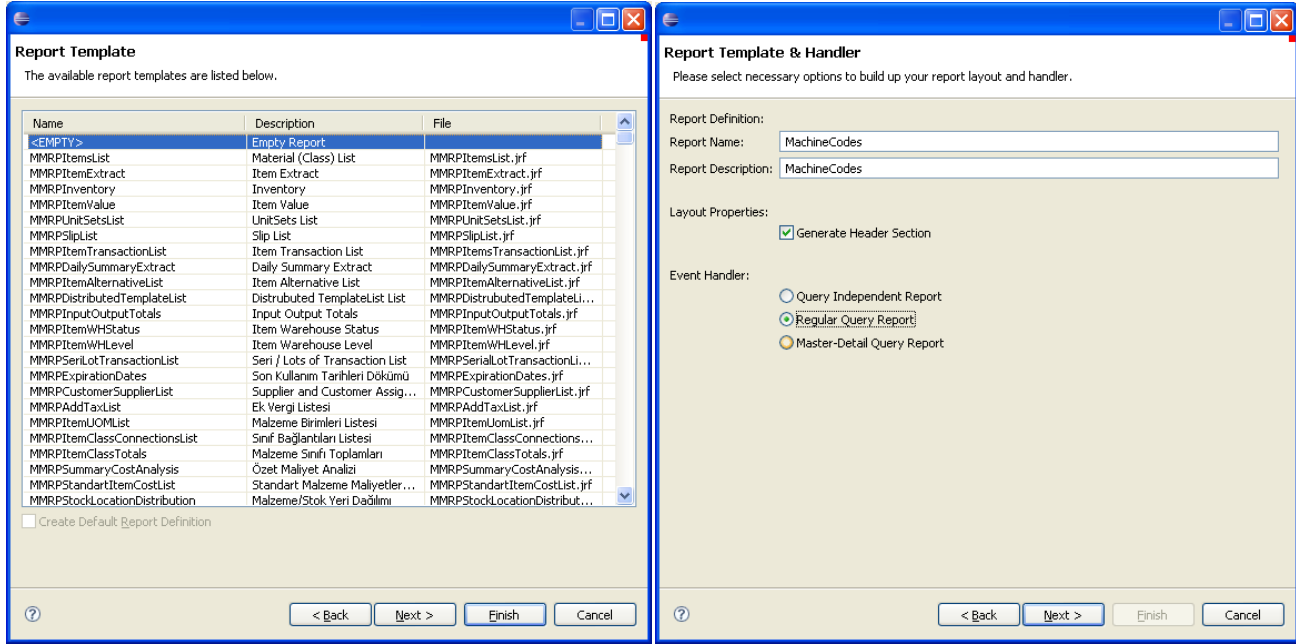


Makine kodlarına yönelik hazırladığımız örnek üzerinden giderek ilgili tabloya ilişkin kayıtların raporunun nasıl alınabileceğini inceleyelim.

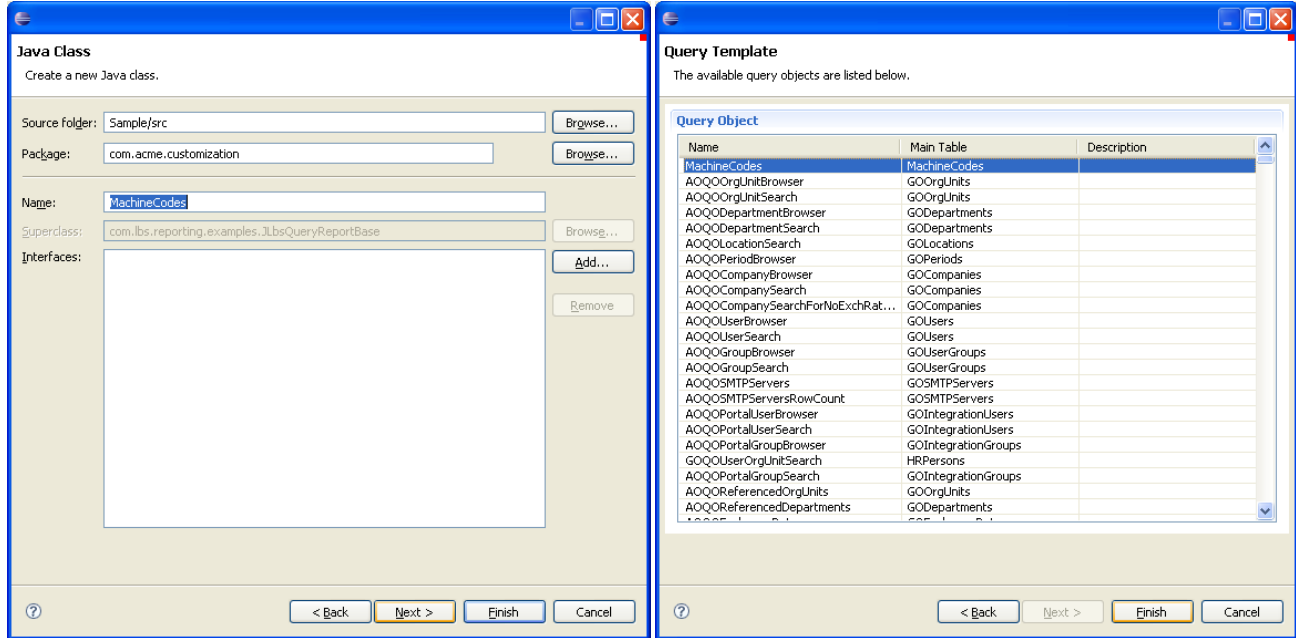
- Sihirbaz üzerinde “**LPT Report Layout**” seçiliyken “**Next**” düğmesine basarak ilerleyelim.
- Gelen sayfada raporumuzu isimlendirelim ve “**Next**” düğmesine basalım.



- Rapor için şablon seçilir, eğer şablon kullanılmayacaksa “<EMPTY>” seçeneği işaretli olarak “**Next**” düğmesine basılarak ilerlenir.
- Rapor tanımında kullanılmak üzere rapor adı ve rapor açıklamasının isimlendirilmesi beklenmektedir.
- Gelen kayıtların düz bir şekilde basılması için “**Regular Query Report**” seçeneği seçilir. Bu kısımda yapılan seçim class dosyasında kodlanmış şekilde yer alacaktır. “**Next**” düğmesiyle ilerlenir.

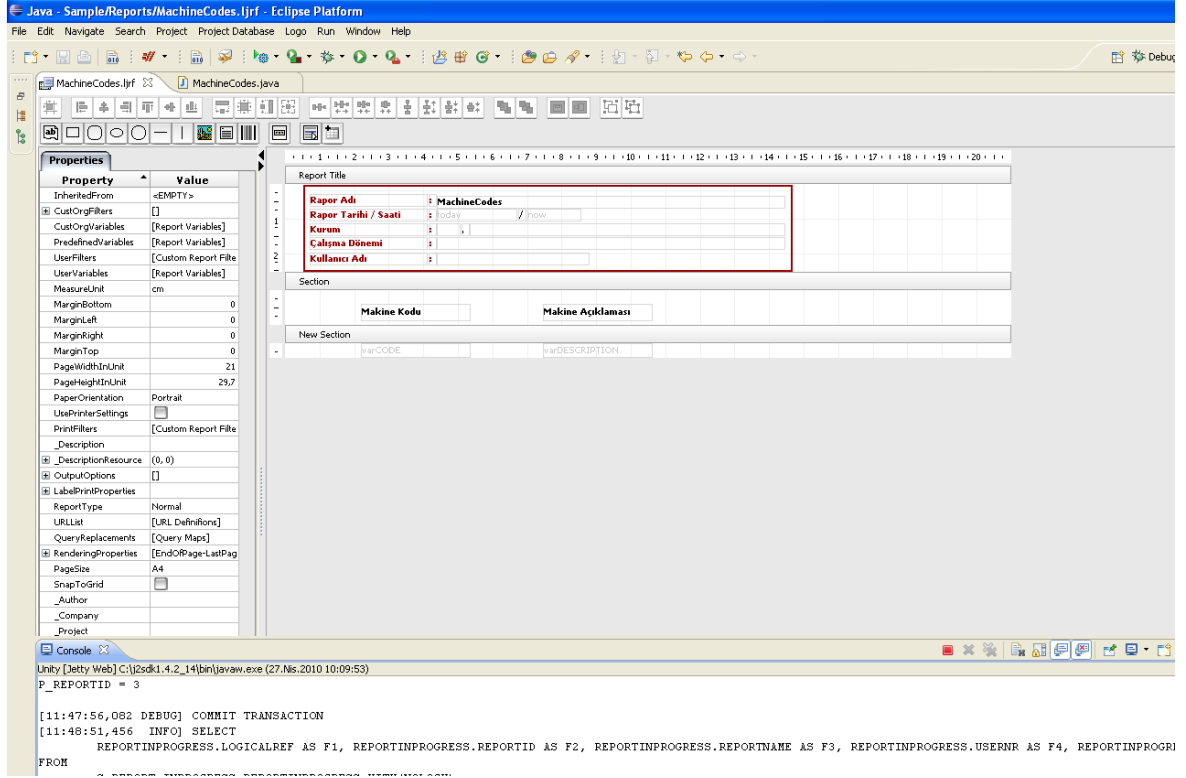


- Gelen ekrandan raporun class dosyasının isimlendirilmesi istenilir.
- “Next” düğmesiyle rapor sihirbazının son sayfasına ulaşılır bu sayfa üzerinde rapor ’un sorgusu seçilir. 3.3 nolu maddede hazırlamış olduğumuz sorguyu rapor sorgusu için seçiyoruz ve “Finish” düğmesine basarak sihirbazı sonlandırıyoruz.



- Sihirbaz sonlandırıldıktan sonra rapor tasarım ekranı açılmaktadır. Ekran boş olarak gelmektedir. İlk bölüm(section) başlık bölümü olsun. Bu bölüme makine kodu ve makine açıklaması için iki adet rectangle koyalım. Bölüm tipini (section Type) header olarak ayarlayalım.

- İkinci bir bölüm(section) ve bu bölüme iki adet datanın geleceği alan(field) ekleyelim.
- Rapor tasarımı üzerinde sağ klik yapıp **“import query fields”** seçeneğini seçelim. Böylece sihirbaz 'da seçmiş olduğumuz sorgunun alanları **“user variables”** alanına gelecektir. Bu variable'ları alan(fields) özelliklerinde bulunan **“VariableName”** için seçelim.



- Rapora ilişkin class dosyasına bakıldığında otomatik olarak bir kod bloğu oluşturulmuştur. Bu aşamada datanın geleceği bölüm(section) ID' si(örneğimize göre 2)kod da gerekli yerde düzeltilerek yazılırsa raporumuz hazır duruma gelmiş olacaktır.

```

package com.acme.customization;

import com.lbs.reporting.JlbsReportLayout;

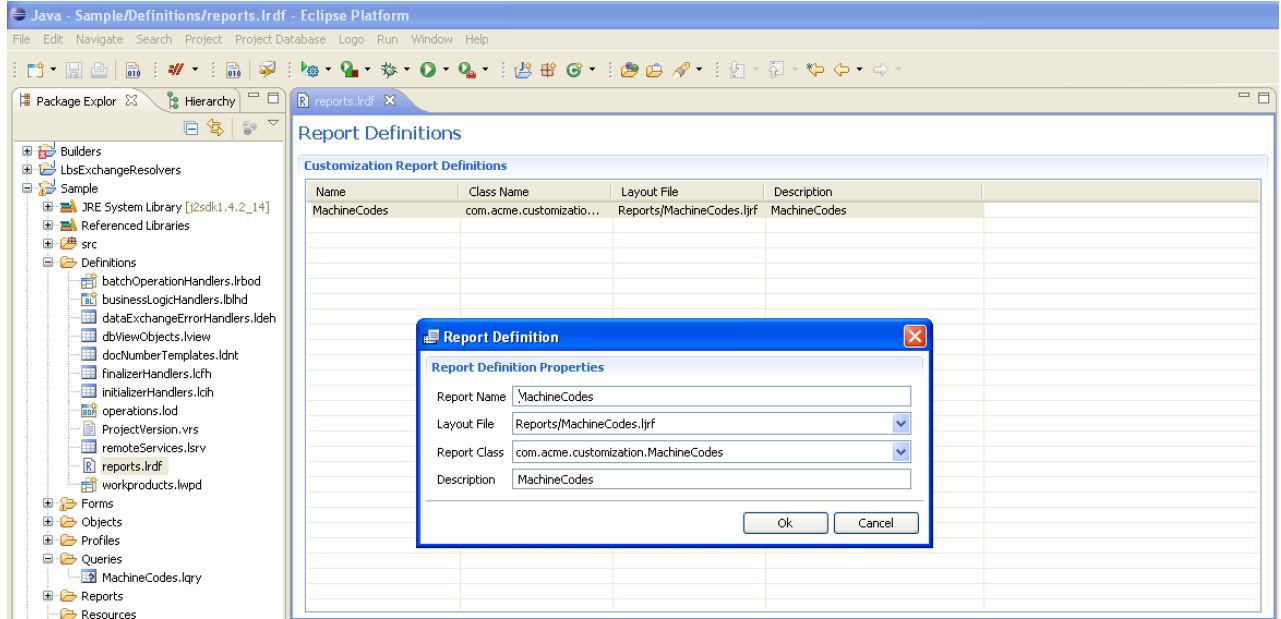
public class MachineCodes extends JlbsQueryReportBase {

    public MachineCodes(JlbsReportLayout layout, IReportRenderer renderer) {
        super(layout, renderer);
        // TODO Auto-generated constructor stub
        setMainQueryName("MachineCodes");
    }

    m_BodySection = 2;
}

```

Tanımlar(Definitions) klasöründe bulunan Reports.Irdf içerisinde raporun layout ve class bilgileri bulunmaktadır. Burada bulunan bilgiler sihirbaz tarafından otomatik olarak oluşturulmaktadır.



### 3.6. Ağaçlar

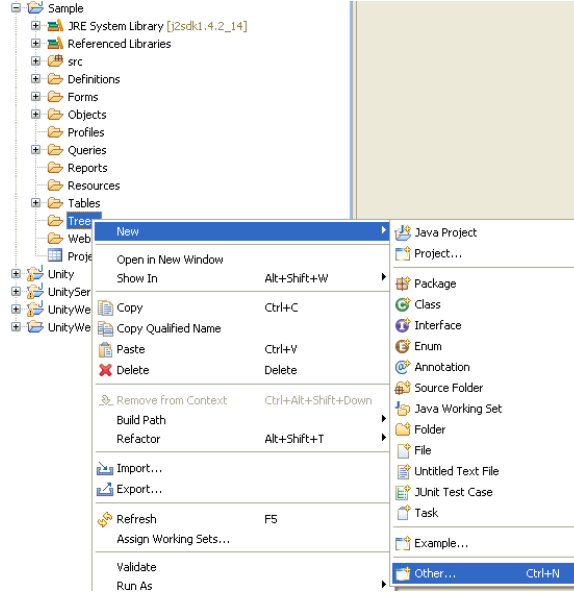
j-guar ürününde menü bir ağaç yapısının şeklindedir ve modüllere ayrılmış bulunmaktadır. Bu modüllerin altında genel yapı itibariyle

- Tanımlar
- Hareketler
- İşlemler
- Rapor

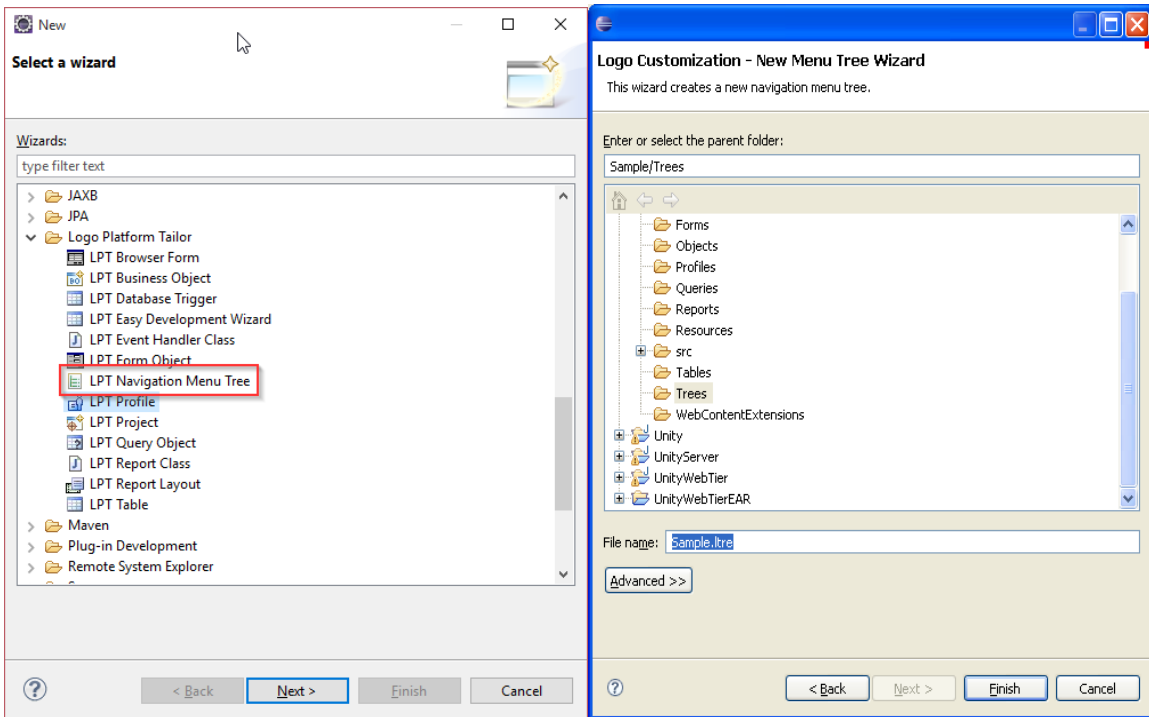
Şeklinde alt modüller yer almaktadır.

Bu ağaca uyarılama yapan kişi tarafından yeni modüller ve alt modüller eklenebilir.

**Trees** klasörü üzerinde sağ click yaparak menüden **"New"** ardından da **"Other"**ı seçerek sihirbazların yer aldığı ekranı açalım.

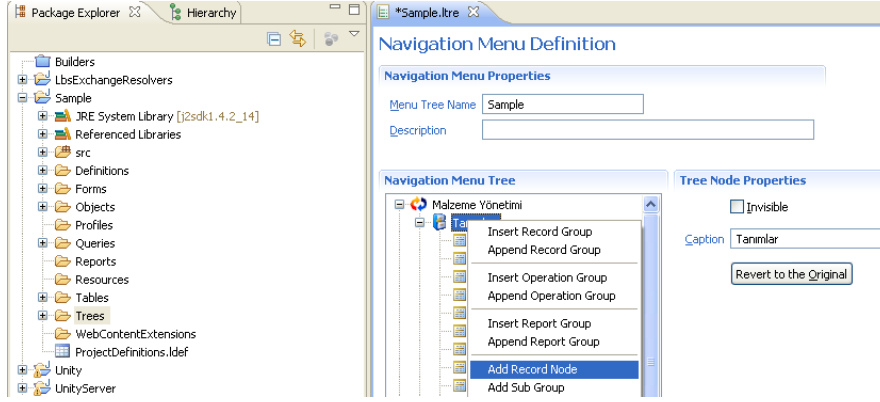


Sihirbazın ilk sayfasında “LPT Navigation Menu Tree” seçeneğin işaretleyerek “Next” düğmesine basarak ilerleyelim. Gelen sayfada ağacımızı isimlendirerek “Finish” düğmesine basarak sihirbazı sonlandıralım.



Ağaç yapısı ekrana gelecektir.

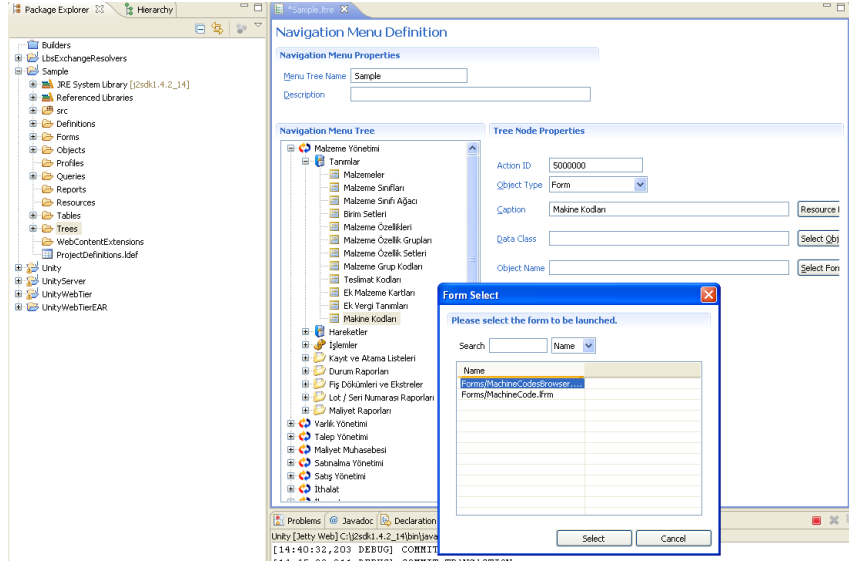
Örneğimize devam edelim makine kodları browser ekranını Malzeme Yönetimi altında bulunan tanımlar alt grubuna eleman olarak takalım bunun için ağaç üzerinde Malzeme Yönetimine tıklayarak altındaki grupları görünür hale getirelim ardında Tanımlar üzerinde sağ click yaparak gelen menüde “Add Record Node” seçeneğini seçelim.



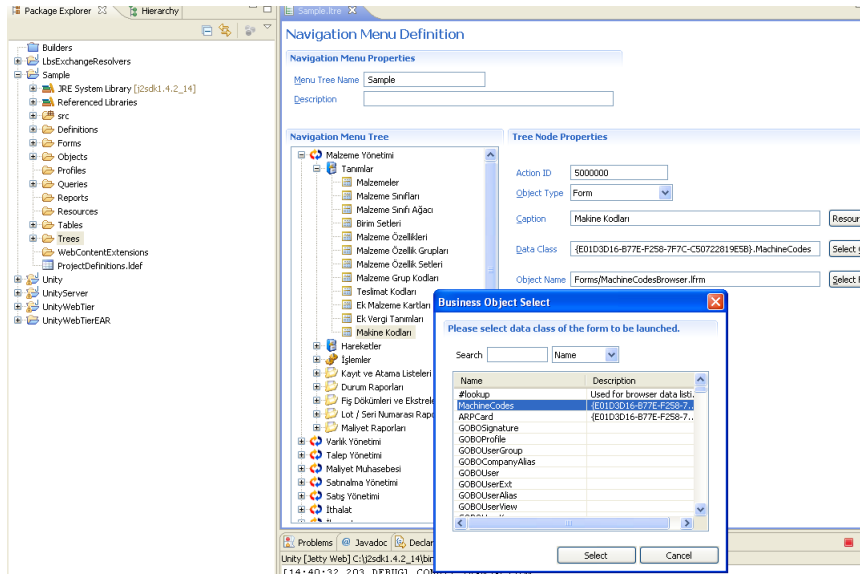
Açılan tanım ekranında “**Object Type**” özelliğini form olarak belirleyelim.

Başlık için “**Caption**” özelliğine Makine Kodları Yazalım.

Nesne seçimi yapalım bunun için “**Data Class**” özelliğine daha önce hazırlamış olduğumuz “**MachineCodes**” nesne tanımını seçelim.



Son olarak açılmasını istediğimiz form’u seçerek işlemimizi tamamlayalım.

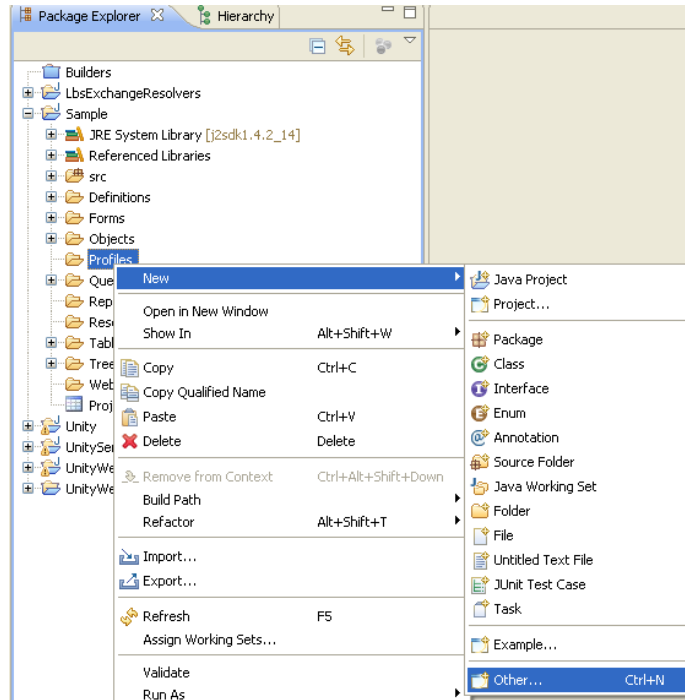


### 3.7. Profil

Profiller j-guar kullanıcılarına yönelik uyarlama içerikleri barındıran yapılardır. Her bir profil bir tree yapısına gereksinim duyar. Tree içerisinde tanımlanmış olan uyarlamalar sadece ilgili profil takılı olan kullanıcılar için geçerli olur.

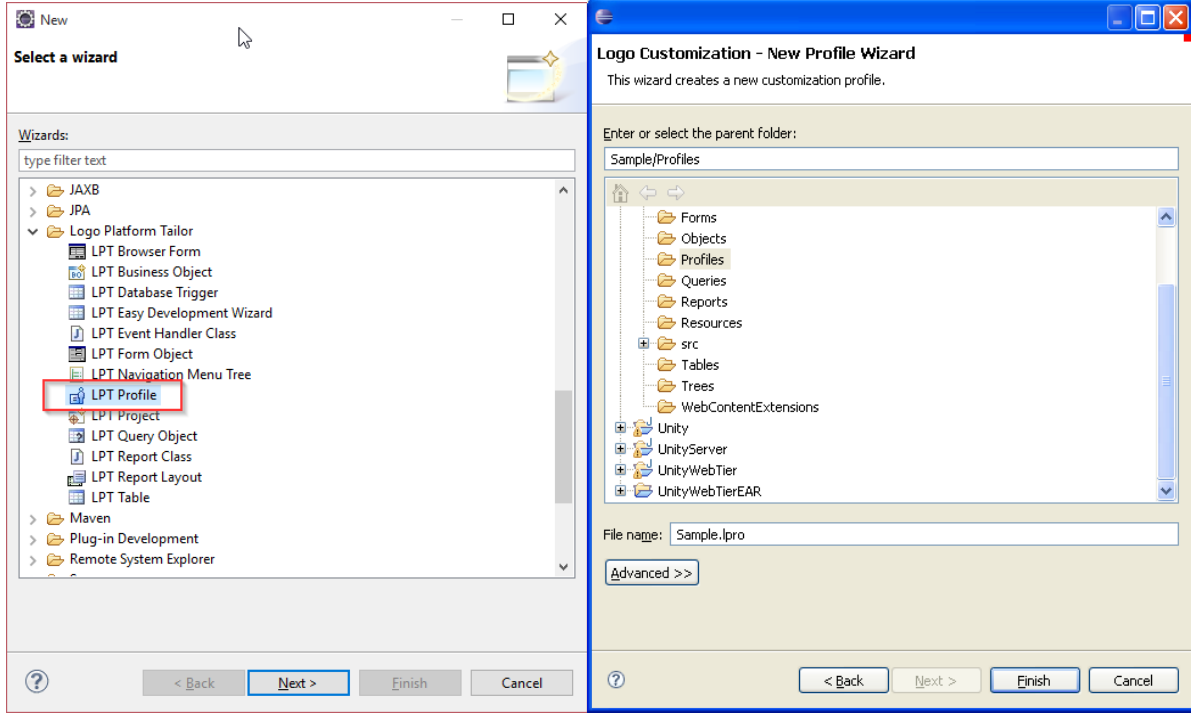
**Önemli Not:** bir uyarlama projesinde birden fazla profil tanımlanabilir.

Şimdi profiline nasıl tanımlanacağına bakalım. **Profiles** klasörü üzerinde sağ click yaparak menüden **“New”** ardından da **“Other”**ı seçerek sihirbazların yer aldığı ekranı açalım.

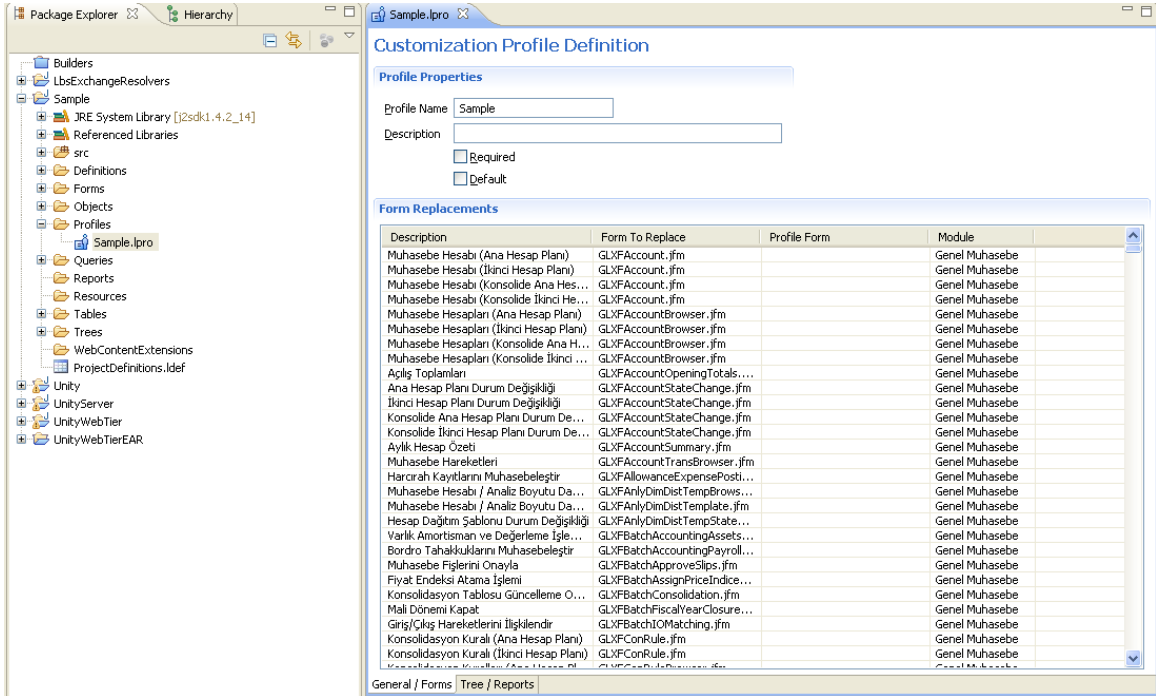


Gelen sihirbaz sayfasında **“LPT Profile”** seçeneğini işaretleyelim ve **“Next”** düğmesine basalım. Gelen sayfada profili isimlendirerek **“Finish”** düğmesine basalım ve sihirbazı sonlandıralım.



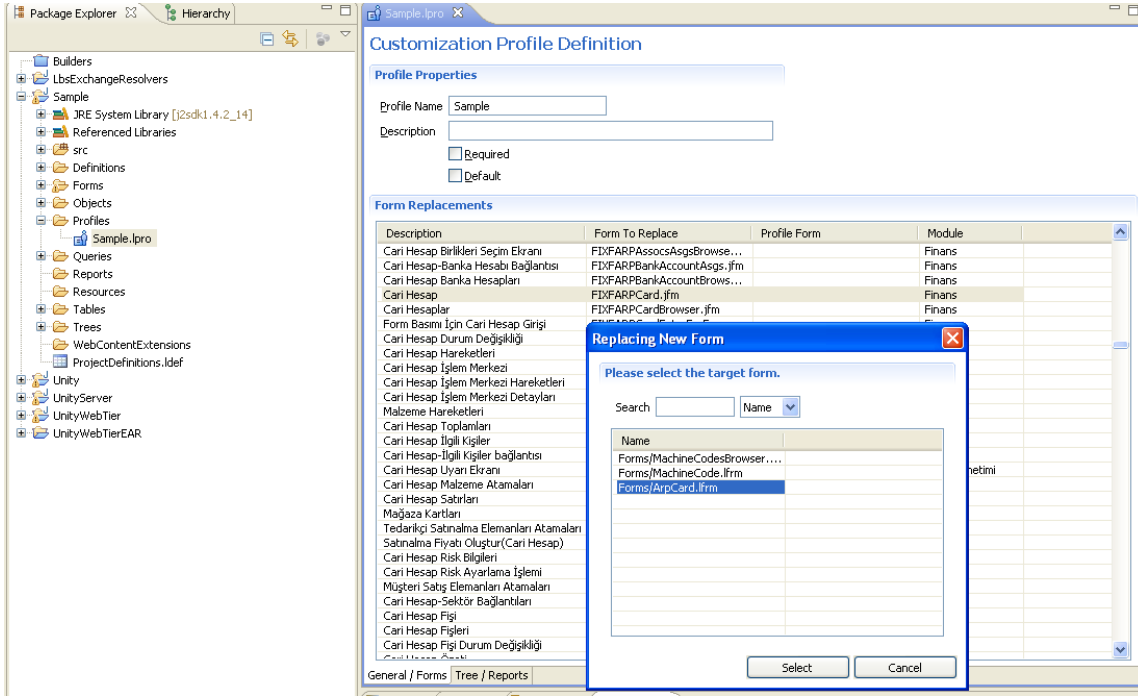


Sihirbaz kapandığında profil sayfası ekrana gelecektir.

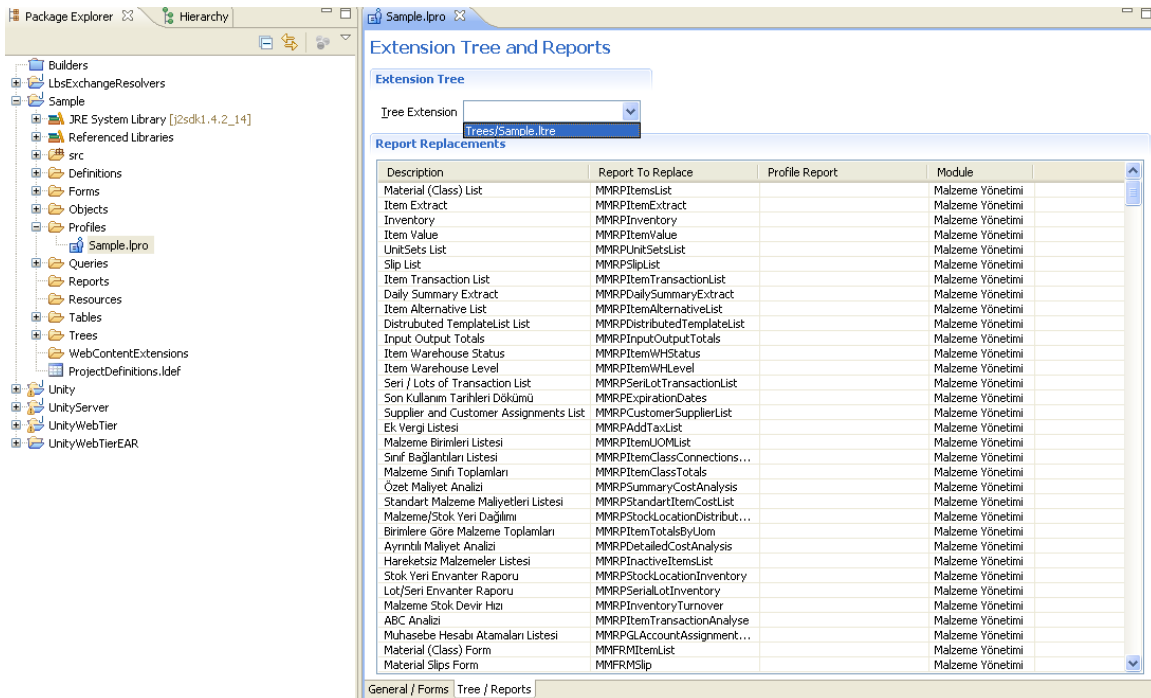


Profil sayfasında j-guar ürününde bulunan bütün form objeleri yer almaktadır. Bu formlar uyarlama olarak tanımlanmış olan formlarla değiştirilebilir.

Örneklendirelim. 3.4.2. nolu bölümde Cari hesap kartına ilişkin bir uyarlama yapmıştık şimdi üründe cari hesap kartı çağırıldığında orjinal kartın değil uyarlama kartımızın gelmesini sağlayalım. Öncelikle cari hesap kartını listede bulalım ve üzerinde çift klik yapalım açılan pop up ekranında uyarlama form listesi gelecektir, bu listeden uyarlama cari hesap kartımızı seçelim ve “Select” düğmesine basarak pop up’ı kapatalım.



Profiller için tree tanımını yapmamız gerekmektedir, bunun için grid'in altında bulunan tab'dan "Tree/Reports" seçeneğine basarak "Extension Tree and Report" sayfasını açalım ve burada "Tree Extensions" alanına daha önce tanımını yapmış olduğumuz tree'yi seçelim. Bu aşamada şunu belirtmek gerekir grid'de j-guar ürününe ait tüm raporlar yer almaktadır. Bu raporlar uyarılama raporlar ile değiştirilebilir.

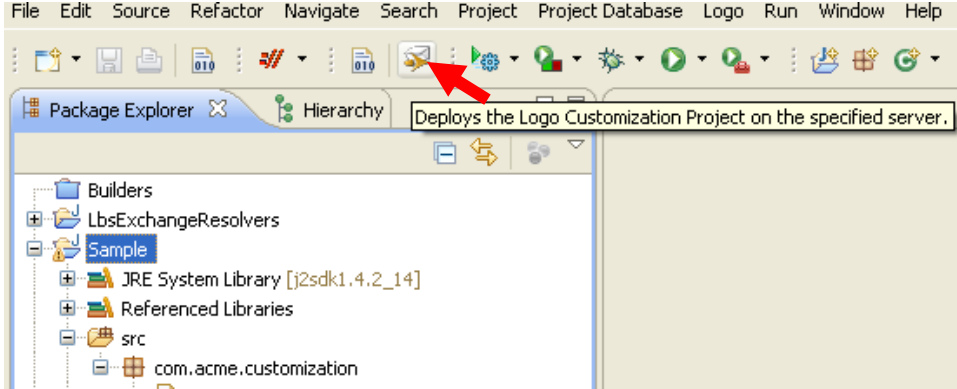


## 4. DEPLOYMENT

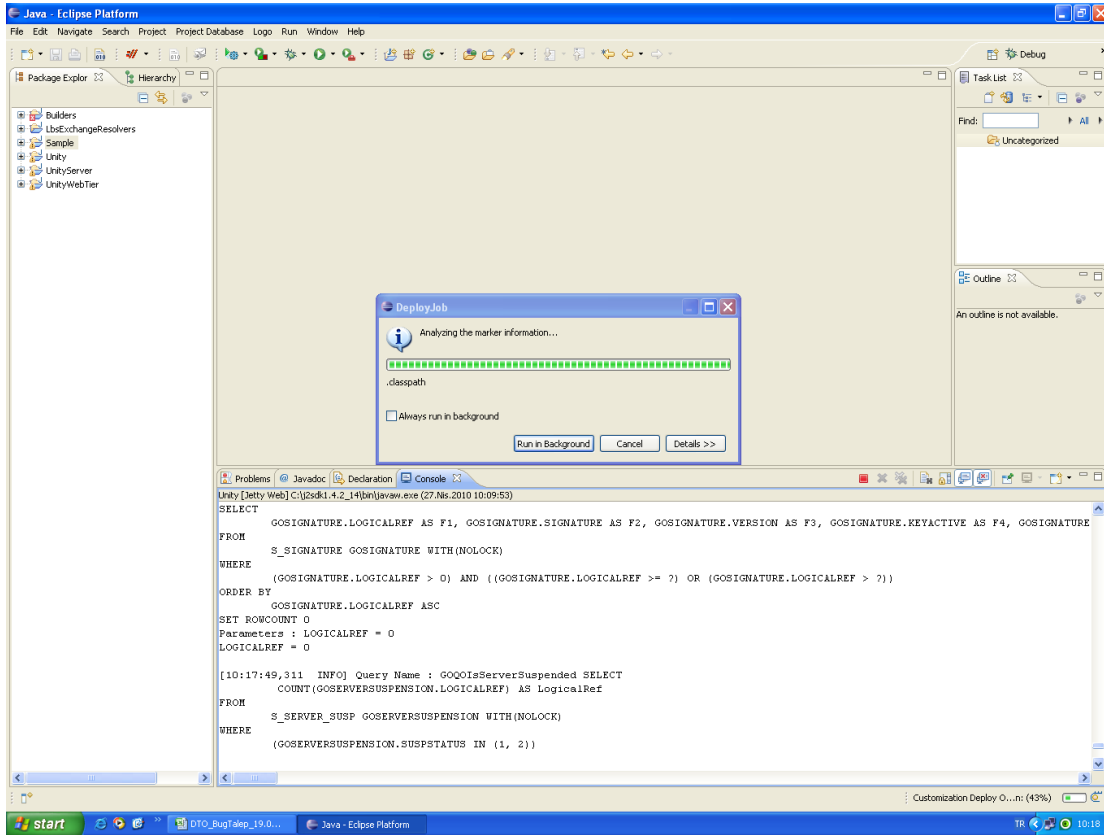
### 4.1. Eclipse üzerinde deploya işlemi

Yapmış olduğumuz ayarlamaların server'a nasıl deploy edileceğini inceleyelim.

Eclipse üzerinden deploy yapabilmek için izlenmesi gereken yol bağlantı ayarlarının kontrol edilmesiyle başlar bağlantının sağlanmasıyla ilgili detaylı bilgi 2.3 nolu maddede anlatılmıştır. Bağlantı kontrolü yapıldıktan sonra package explorer üzerinden projeye tıklanarak aktif hale geçirilir ve deploy tuşuna basılır.

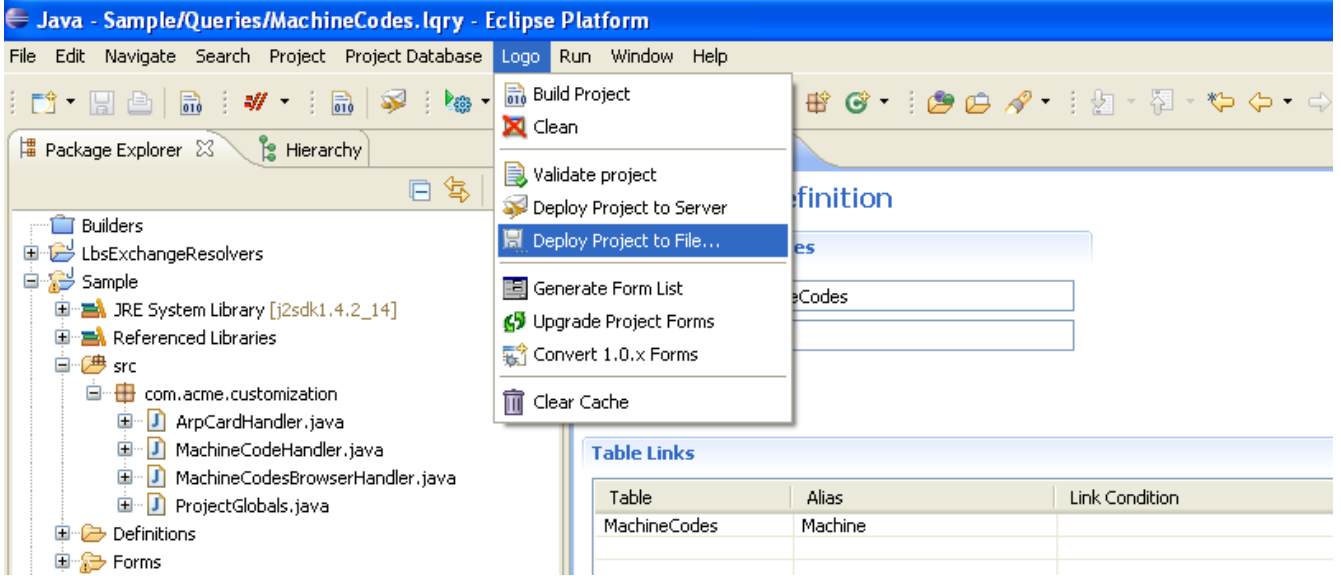


Deploy işlemi başladıktan sonra eclipse'in sağ alt kısmında uyarlamanın sever'a gönderimi ile ilgili ilerleme durumu takip edilebilir.

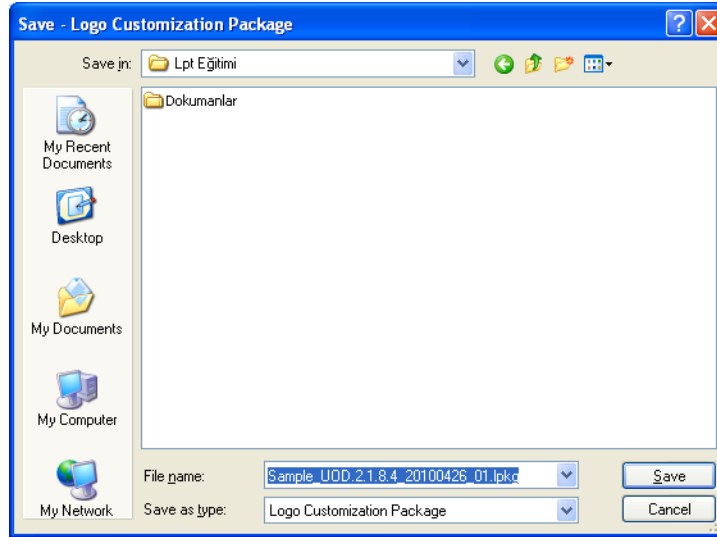


## 4.2. Ürün içerisinde deploye işlemi

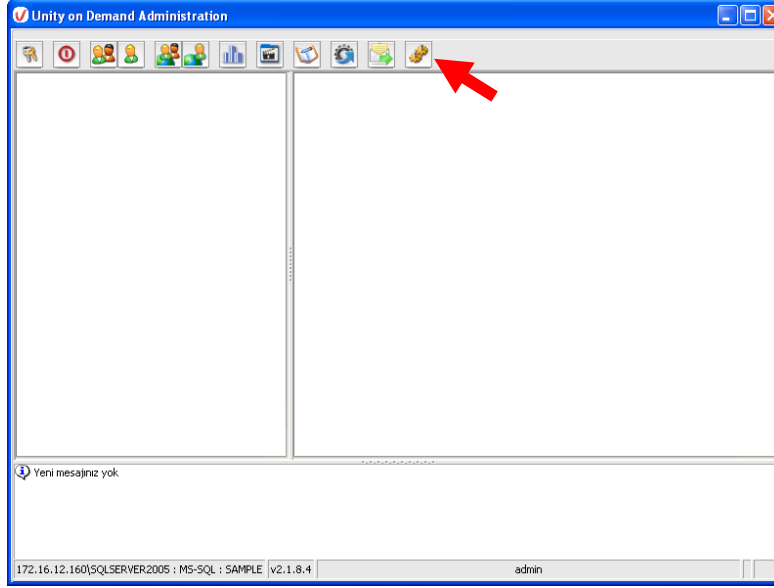
Ürün içerisinden deploy yapabilmek için öncelikle lpkg uzantılı uyarılama dosyası oluşturulması gerekmektedir. Bunun için eclipse'in logo menüsünden "Deploy project to file" elemanı seçilir.



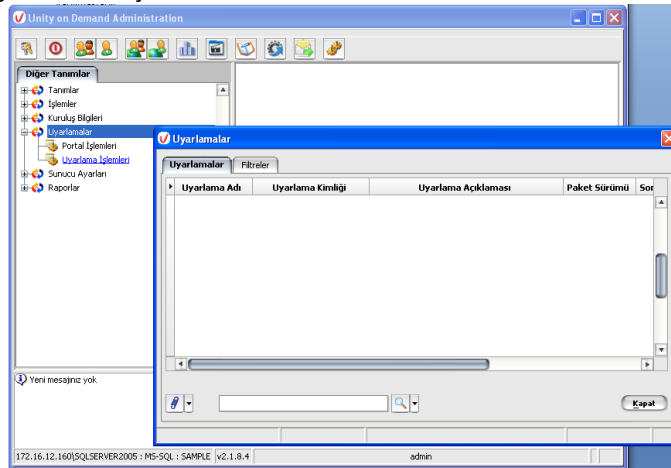
Lpkg dosyasının kaydedileceği yer belirlenir, yeri gelmişken lpkg dosyası için isimlendirme otomatik olarak yapılmaktadır.



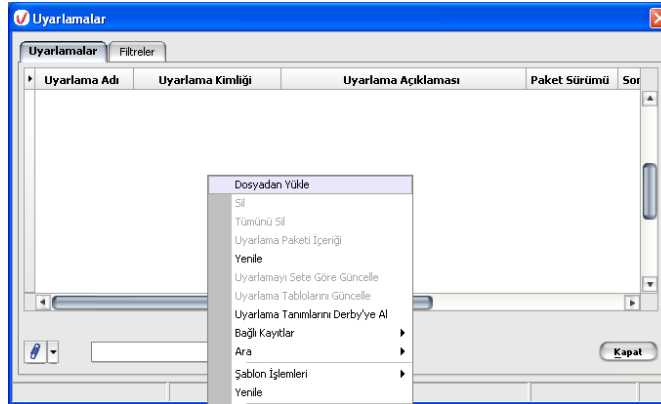
Lpkg uzantılı dosya oluşturulduktan sonra j-guar Administrator uygulaması açılır. Uygulamada çark imgesi olan diğer tanımlar düğmesine basılır.



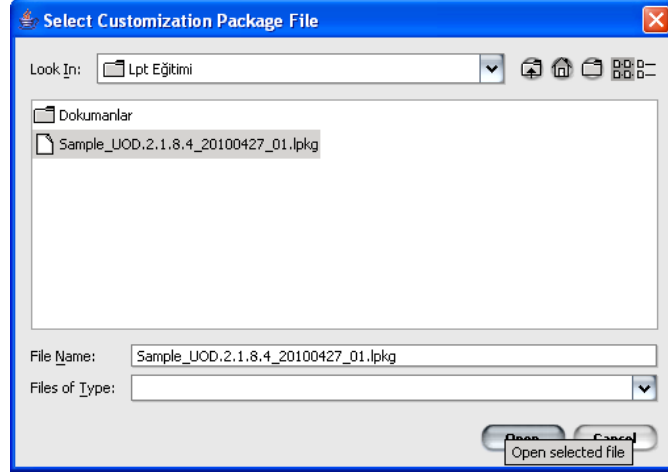
Sol tarafta gelen diğer tanımlar tabında “Uyarlamalar” ardından da “Uyarlama İşlemleri” elemanına tıklanır. Böylelikle uyarlama paketlerin bulunduğu browser açılır..



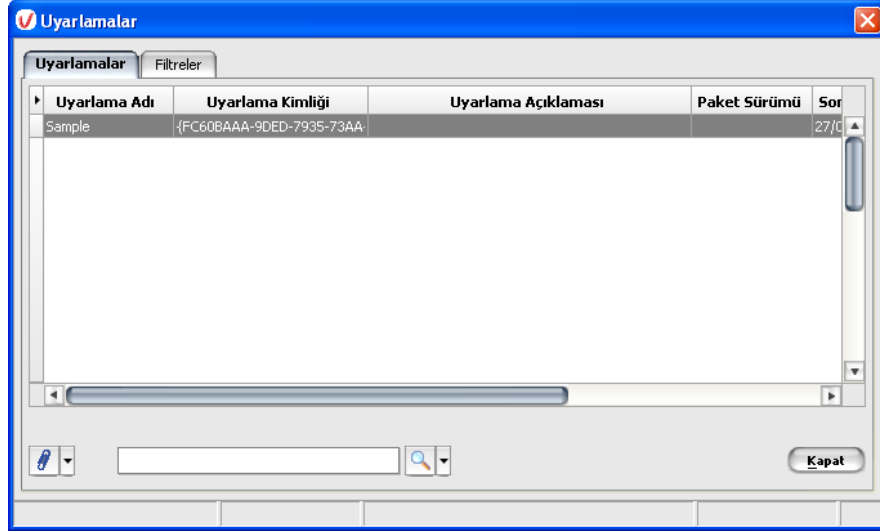
Uyarlamalar browser’ı üzerinde sağ click yapılarak “Dosyadan Yükle” seçeneği ile seçilir.



Lpkg uzantılı uyarlama dosyamızı seçerek open düğmesine basılır.

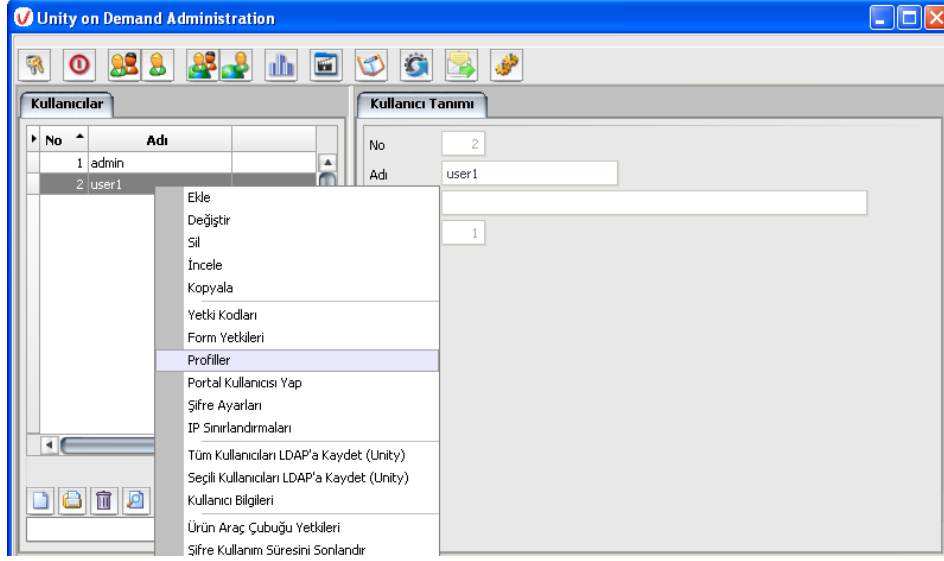


Kısa bir süre içerisinde uyarlama dosyası browser’da yer alacaktır. Böylelikle uyarlama deploy edilmiş olacaktır.

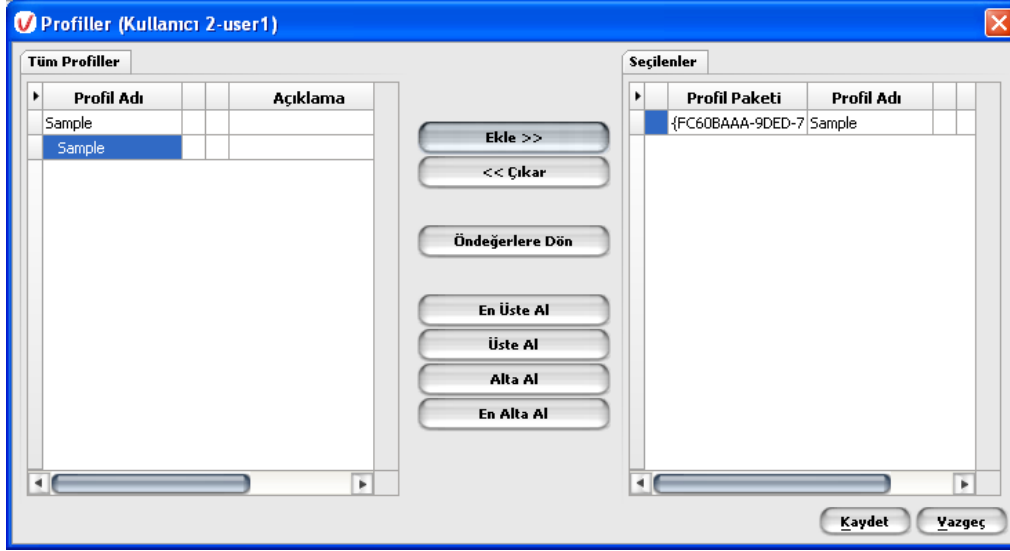


#### 4.3. Deploy sonrası işlemler

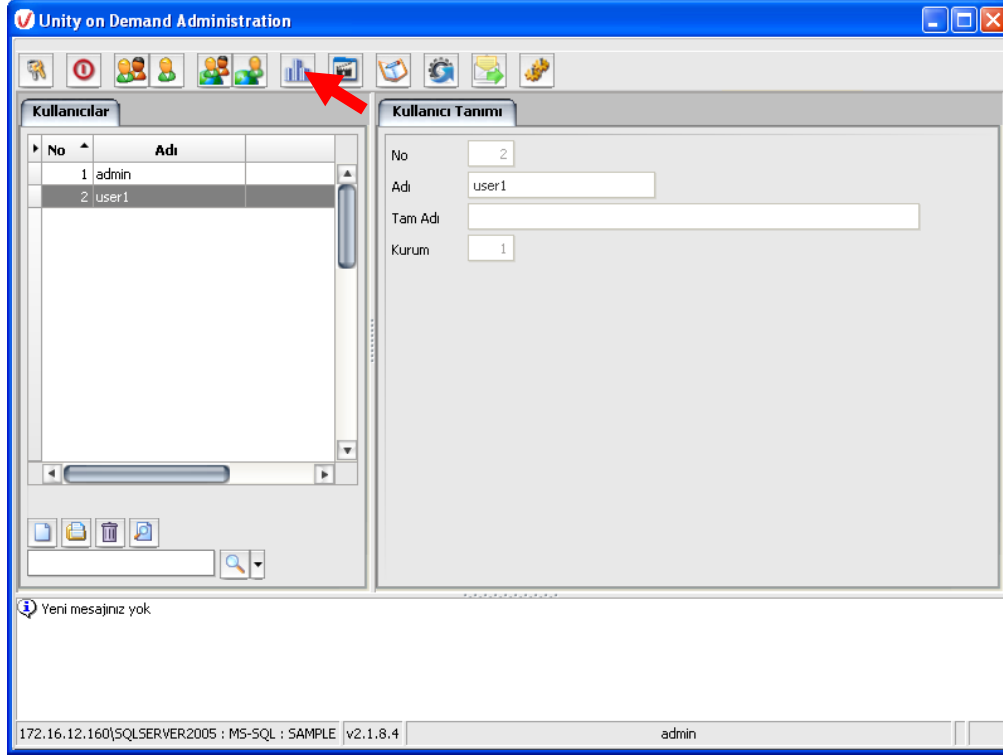
İster eclipse’den yapılınsın ister ürün üzerinden yapılınsın deploy sonrası yapılacak işlemler ortaktır. Öncelikle deploy etmiş olduğumuz uyarlamaları bir kullanıcıya profil olarak takmamız gerekmektedir. Kullanıcılar üzerinde sağ klik yapılarak “Profiller” seçilir.



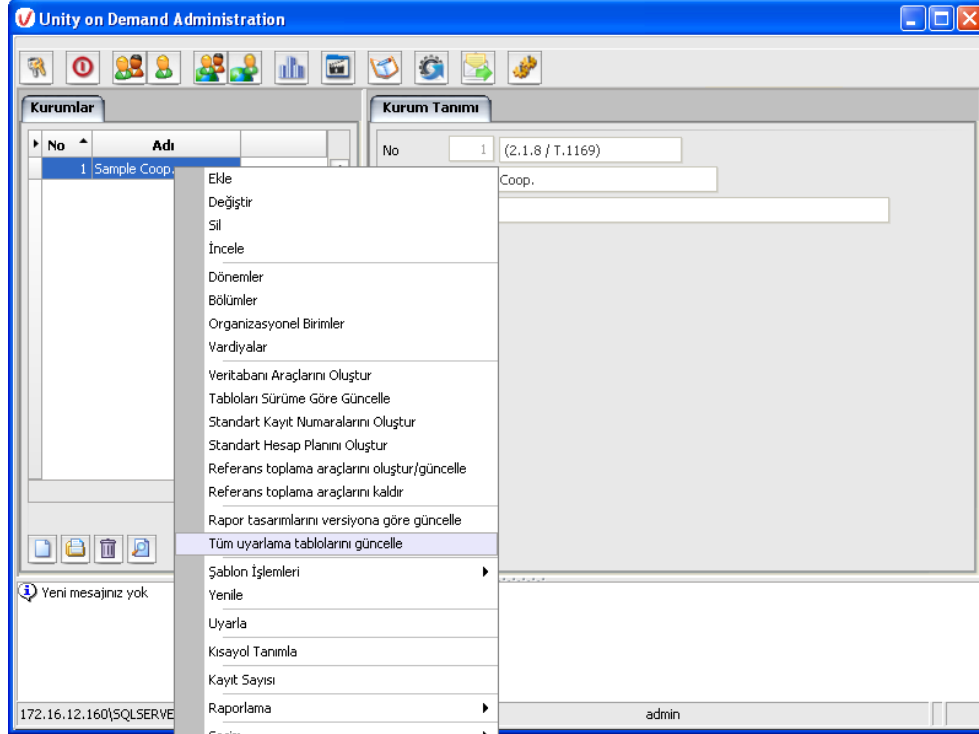
Açılan profil browser'ında sol gridde tüm uyarlama profilleri, sağ grid'de ise sadece seçili kullanıcıya bağlanmış uyarlama profilleri görüntülenmektedir. Sol griddeki "Sample" profilini seçtikten sonra ekle düğmesine basarak kullanıcıya profili ekliyoruz ve kaydet düğmesine basarak profil ekleme işlemini sonlandırıyoruz.



Eğer uyarlama olarak yeni bir tablo eklenmişse bu tablonun database'de açılması için deploy etmek yeterli değildir. Deploy yapıldıktan sonra uyarlama tablolarının güncellenmesi gerekmektedir. Bunun için Kurumlar düğmesine basalım.



Ekran gelen kurum tanımımız üzerinde sağ klik yaparak “Tüm Uyarlama Tablolarını Güncelle” seçeneğini seçelim.



Uyarlama Tabloları güncellenecektir uyarısı görüntülenecektir. “Tamam” düğmesine basarak işlemi başlatalım, alt kısımda bulunan alanda tabloların güncellenme durumlarıyla ilgili bilgi verilmektedir. Tabloların güncelleme işlemi esnasında yaşanan sıkıntılar bu alanda kullanıcıya iletilmektedir.



