

09 - 模块 9：约束数据

模块 9：约束数据

模块简介（模块图块）：

在此模块中，您将学习如何约束返回的数据。

讲座 9.1：简介

到目前为止，您看到了许多功能，并且应用程序已经发展到一个非常可用的水平。但是您仍然缺少一些信息，该是深入探讨 Mendix 的时间了。到目前为止，您一直从数据库中检索所有信息，但您可能记得，Summerhill 医院的人希望根据正在访问它的用户来查看特定的数据。接下来，将注意力集中在约束数据上。本模块将教您所有关于用 XPath 约束数据的知识。我们将探讨：

- 可以使用 XPath 的地方
- 查询和标记
- 运算符
- 约束
- 函数
- 变量

完成本模块后，您将成为 Mendix 中的数据选择向导！

讲座 9.2：写入 XPath 查询和标记

要选择 Mendix 中的数据，您可以使用 XPath，这是一种查询语言，它完全基于用于选择 XML 文件中数据的 XPath 语言。在 Mendix XPath 中找到的许多运算符也存在于 XML XPath 中。虽然标记可能相似，但它们的含义在 Mendix XPath 中略有不同，因此，我们来看一个查询并进行解压。在 Java 中选择数据时，您可以使用如下所示的查询。

```
//UserManager.Employee[Name = 'Jones']
```

//UserManager.Employee 表示 XPath 查询的开始。它表明您正在使用哪个实体，在本案例中，是 **UserManager** 模块中的**员工**实体。要开始查询，您可以使用 [] 符号。要过滤的属性将位于这些括号内。您可以使用 = 符号将属性与值进行比较。这是此查询的最后一部分的工作方式。

在 Studio Pro 中编写查询时，您可以在以下情况下使用 XPath：

- 在微流中**检索活动**
- 数据小组件中的 **XPath 数据源**
- 安全性中的 **XPath 约束标签**

您可以通过手动添加 [] 字符来开始和结束查询。输入 [后，就可以像在微流表达式中一样使用 Ctrl + Space。查询可能如下所示。

```
[Name = 'Jones']
```

现在您已经看到第一个 XPath 查询，是时候介绍您在这些查询中可以找到的标记了：

标记	定义
//	XPath 查询的开始
.	模块和实体名称之间的分隔符
/	实体和关联之间的分隔符，反之亦然
[]	约束
()	约束分组

讲座 9.3: XPath 运算符

在 Mendix Studio Pro 中编写 XPath 时，您可以使用比较运算符来选择所有可用对象的子集。有效运算符的列表如下：

运算符	描述	示例
=	等于	价格 = 9.8
!=	不等于	价格 != 9.8
<	小于	价格 < 9.8
<=	小于等于	价格 <= 9.8
>	大于	价格 > 9.8
>=	大于等于	价格 >= 9.8
或	或	价格 > 1 或 价格 < 0
与	与	价格 > 1 和 价格 < 0

Java 代码中，XPath 还支持使用 +、-、* 和 div 符号来表示加、减、乘和除。

讲座 9.4: XPath 约束

现在，您已经了解 XPath 的基本理论，您可以打开它，查看使用 XPath 的约束数据的所有功能。使用 XPath 时，您可以在实体上使用任何属性或关联。使用关联时，您可以使用 / 运算符访问关联另一端的实体。从此之后，您可以使用该实体的任何属性或关联。生成查询时，Mendix 将使用链接表，将实体从生成的查询拖放到两个关联之间。这显著加快

了数据检索，因为链接表明显小于实体。一旦到达要过滤的实体，您就可以使用属性上的比较运算符。

```
//Administrator.Account[UserManager.Employee_Account/UserManager.Employee/Firstname = 'Henry']
```

如果您需要更多的选择能力，有一些约束函数可以让您做更复杂的比较。约束函数的完整列表可在相关[文档](#)中找到。您可以在以下列表中找到一些关键函数和简短描述：

- **contains(StringAttribute, String)** 可用于确定属性是否包含特定字符串。
- **starts-with(StringAttribute, String)** 可用于确定属性是否以特定字符串开头。
- **ends-with(StringAttribute, String)** 可用于确定属性是否以特定字符串结尾。
- **string-length(StringAttribute)** 返回 StringAttribute 的长度。
- **day-from-datetime(DateAttribute)** 返回 DateAttribute 的本月日期
- **true()** 返回“True”，用于将布尔属性与“True”进行比较。 [HasAccount = true()]
- **false()** 返回“True”，可用于将布尔属性与“False”进行比较。 [HasAccount = false()]

XPath 使用一组关键字来确定属性是否具有值。这些关键字为空和 NULL。要确定关联是否存在，可以在 XPath 查询中使用该关联。例如，下面将返回具有与其关联的帐户的所有员工对象：

```
[UserManager.Employee/UserManager.Employee_Account]
```

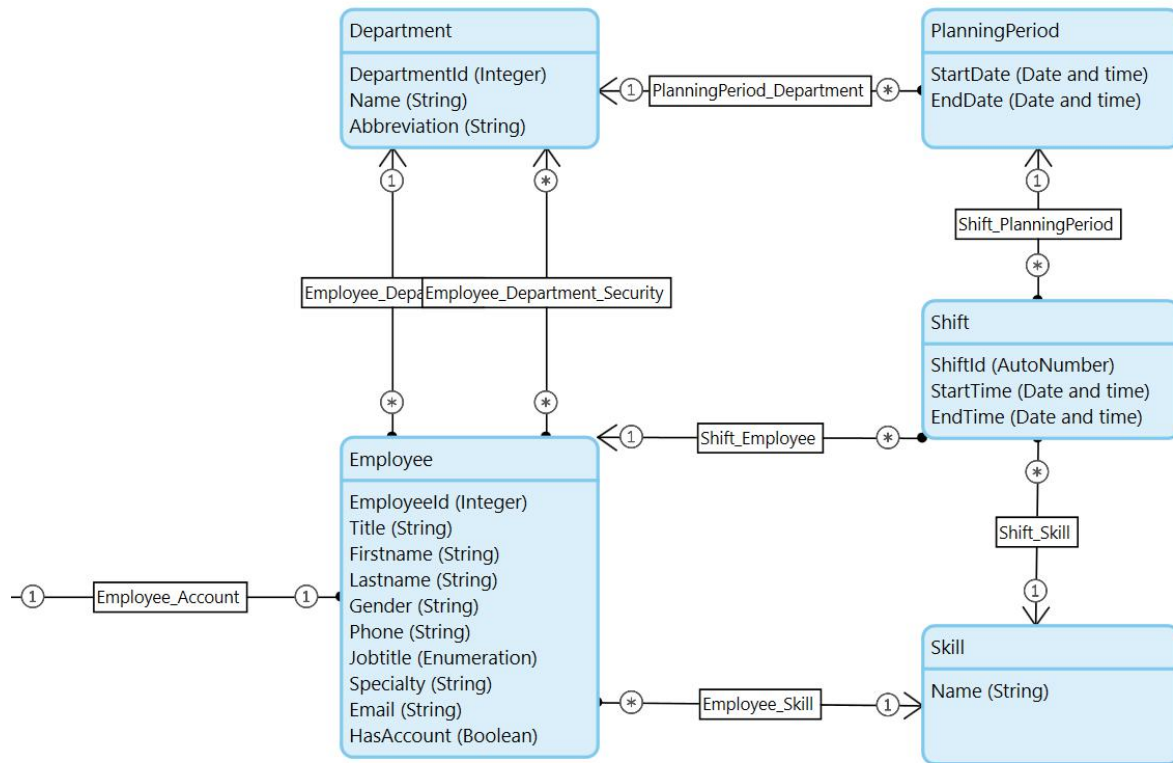
最后，在 XPath 查询中可以使用一些系统变量：

- **对象相关**
 - [%Current_User%] 当前用户的 ID
 - [%Current_Object%] 当前上下文中的对象的 ID
- **用户角色**每个用户角色都有一个常数。常数为 [%UserRole_{userrolename}%]，例如 [%UserRole_Administrator%]
- **与时间相关**与时间有关的常数，如 [%CurrentDateTime%] 和 [%BeginOfCurrentDay%]，也可用。可以在文档中找到完整列表。

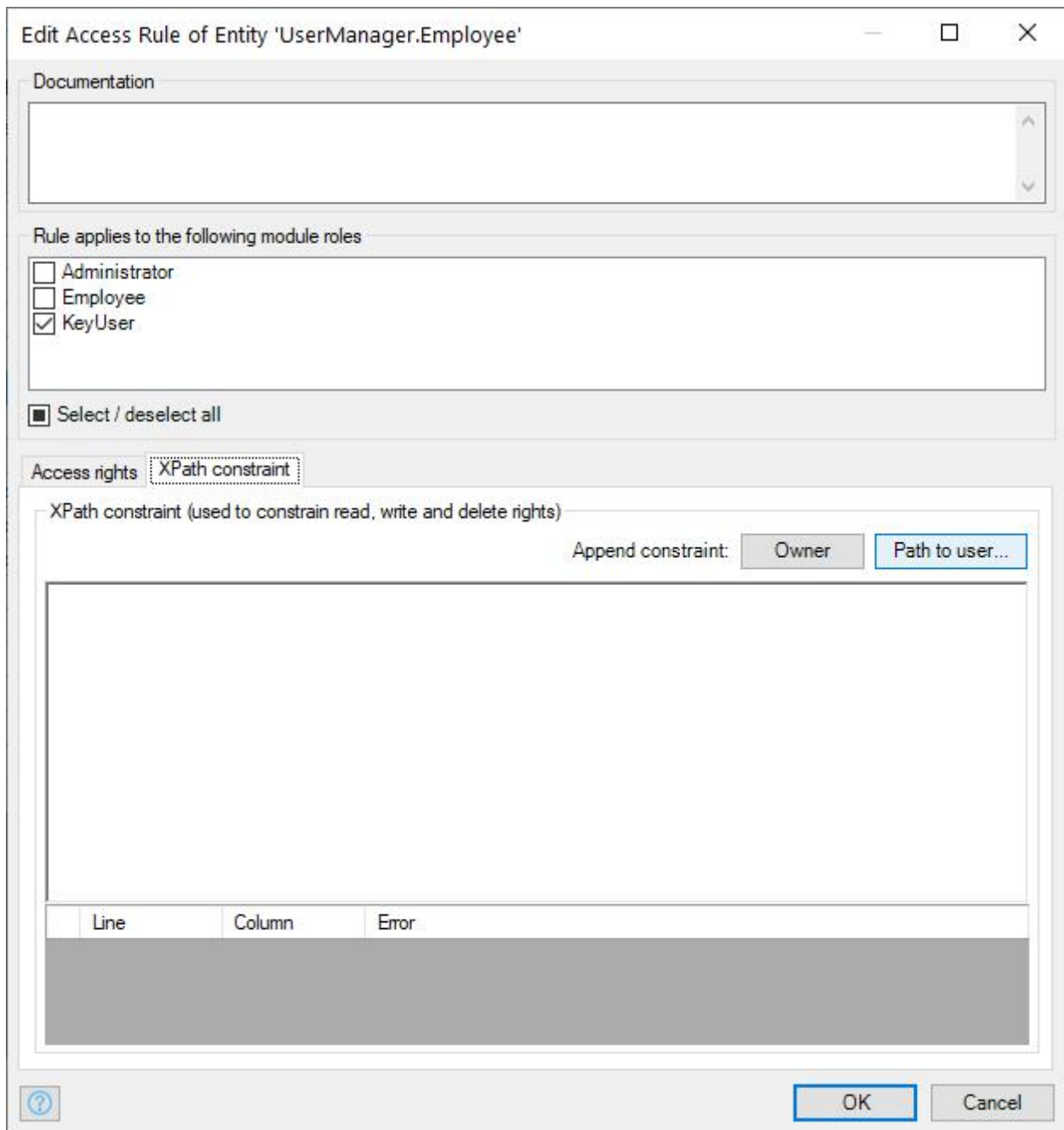
讲座 9.4.1：限制关键用户的访问权限

有了这个理论，现在是时候敲定您的应用程序的安全性了。Summerhill 的人谈到，希望他们的关键用户只能根据他们所属的部门访问一个用户子集。这意味着我们需要一个附加关系来配置该访问权限。我们现在就开始吧！

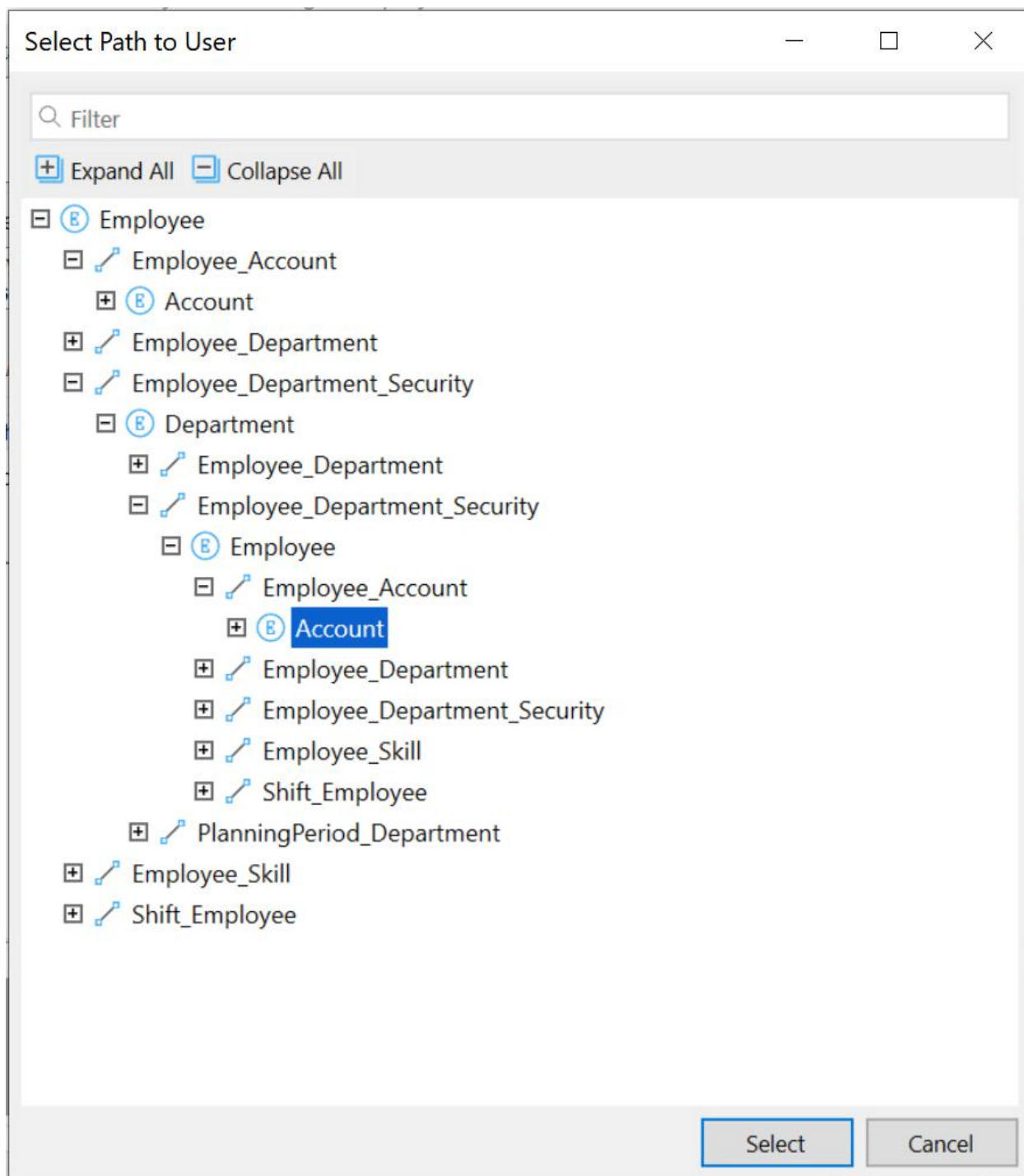
1. 打开域模型，将第二个关系从**员工**添加到**部门**。将关系设置为多对多，并赋予其名称 **Employee_Department_Security**。



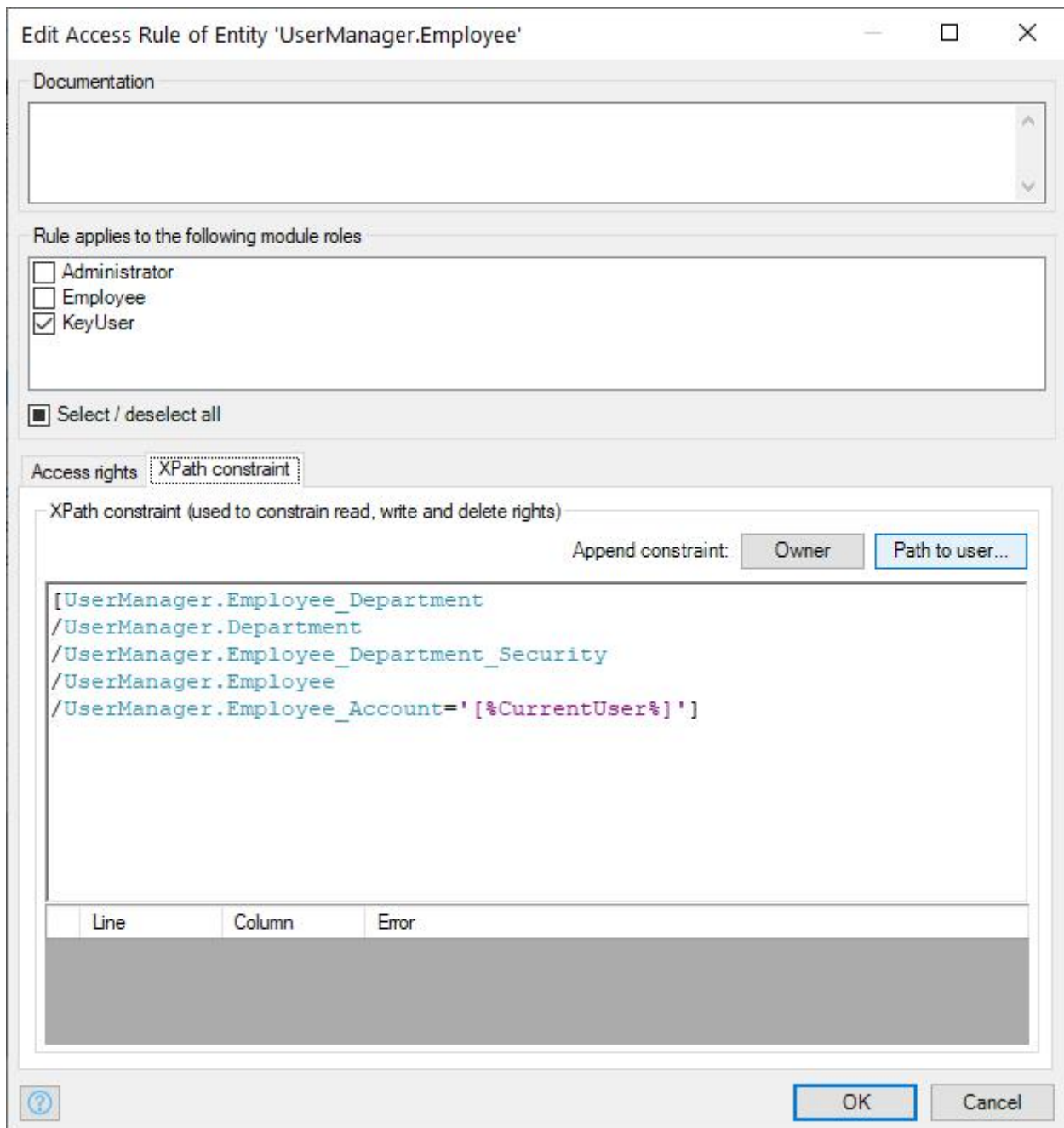
2. 打开模块安全性，然后打开实体**员工**和模块角色 **KeyUser** 的实体访问规则。切换到 **XPath** 约束标签，然后单击**通往用户的路径**。



3. 使用树状视图和**员工与部门**之间的两个关系，加上**员工与帐户**之间的关系，得到下图中的关系。单击**选择**。



XPath 现在应如下所示:



4. 更改为访问权限标签，并确保 **UserManager.Employee_Department_Security** 成员被设置为**读取**。这将防止 **KeyUsers** 更改他们的访问权限。
5. 单击**确定**。
6. 打开 **Employee_NewEdit** 页面，然后从**工具箱**为您的页面添加一个**输入参考集选择器**。确保它位于**员工实体**的**数据视图**中。

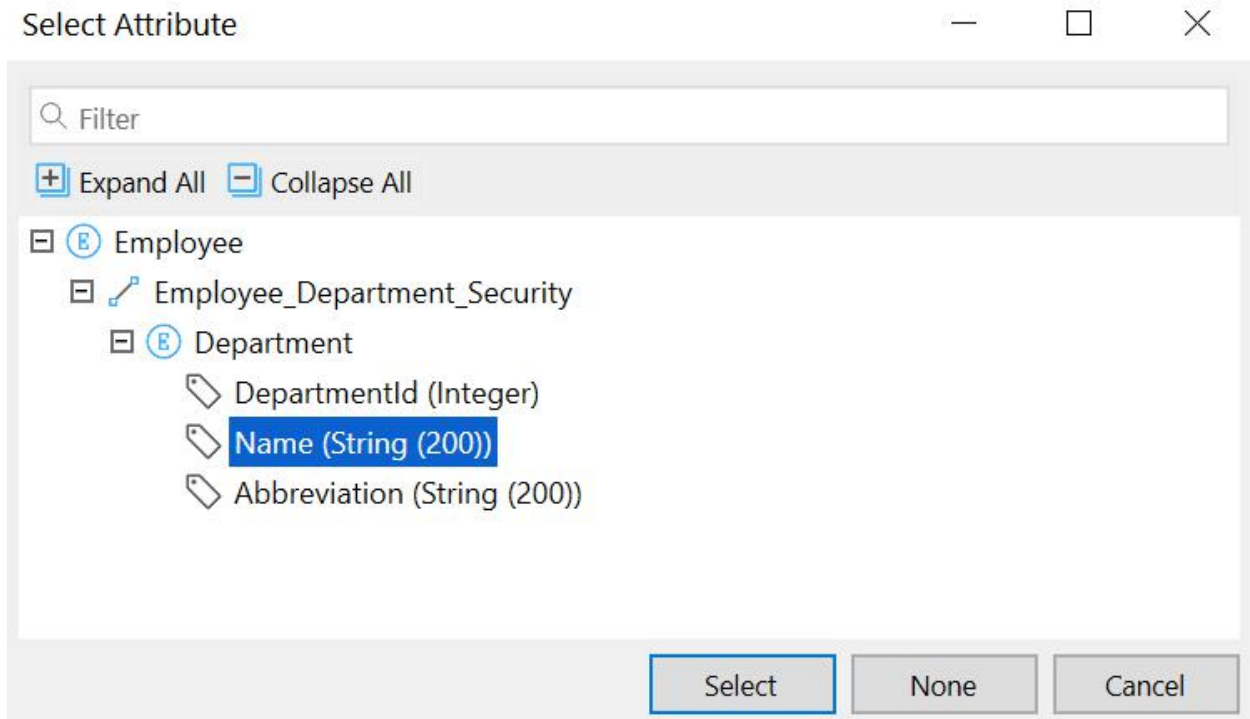
Manual - 12

[Employee, page parameter]

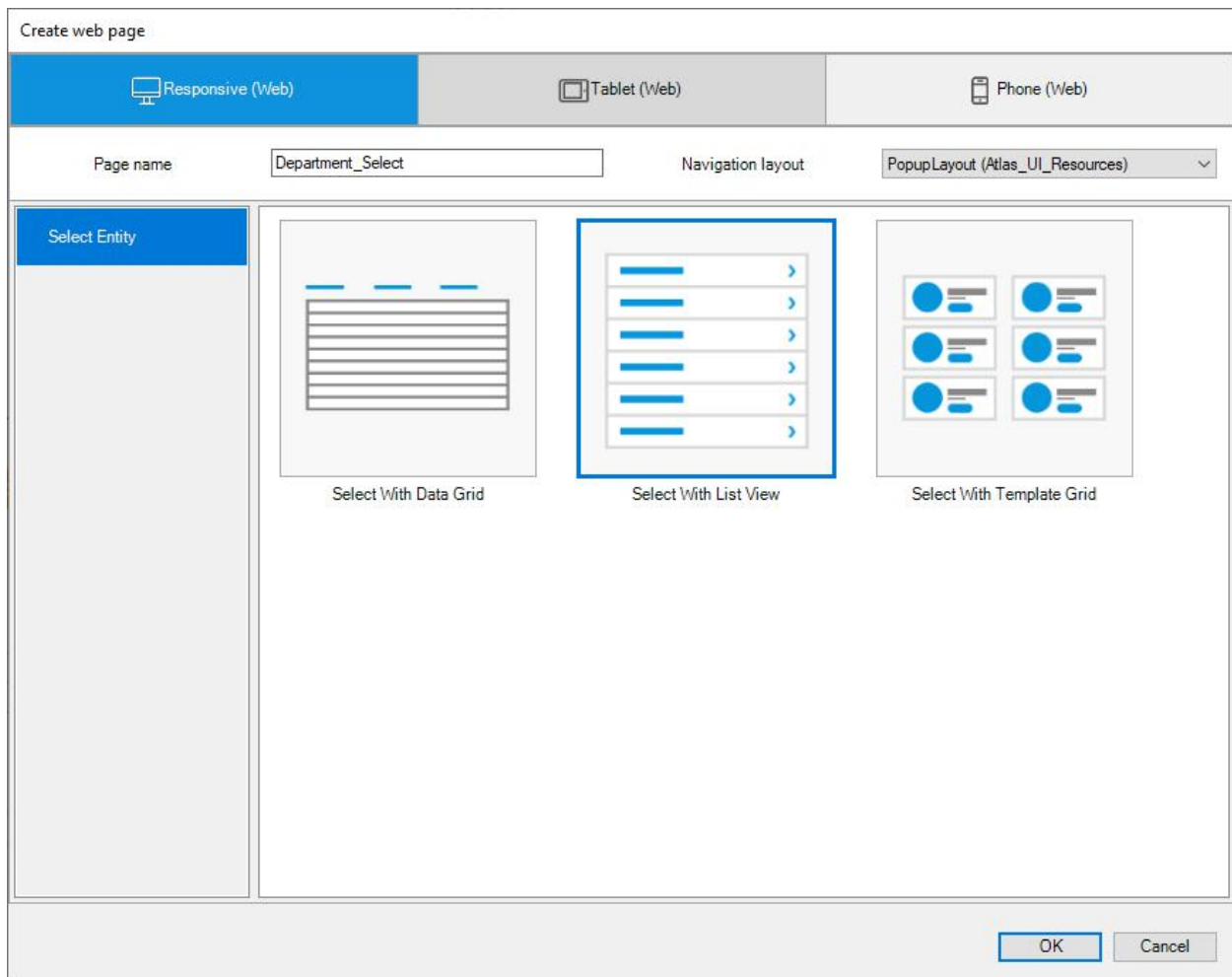
Auto-fill	Auto-fill
Employee... <input type="text" value="[Employeeid]"/>	Has acco... <input type="radio"/> Yes <input type="radio"/> No [HasAccount]
Title <input type="text" value="[Title]"/>	<div style="border: 1px solid #ccc; padding: 2px;"> [Account, over association 'Employee_Account'] </div>
Firstname <input type="text" value="[Firstname]"/>	Username <input type="text" value="[Name]"/>
Lastname <input type="text" value="[Lastname]"/>	Last login <input type="text" value="[LastLogin]"/>
Gender <input type="text" value="[Gender]"/>	Blocked <input type="radio"/> Yes <input type="radio"/> No [Blocked]
Phone <input type="text" value="[Phone]"/>	Active <input type="radio"/> Yes <input type="radio"/> No [Active]
Jobtitle <input type="text" value="[Jobtitle]"/>	User role <input type="text" value="[UserRoles/UserRole/Name]"/>
Specialty <input type="text" value="[Specialty]"/>	Language <input type="text" value="[User_Language/Language/Code]"/>
Email <input type="text" value="[Email]"/>	Time zone <input type="text" value="[User_TimeZone/TimeZone/Code]"/>
Departme... <input]"="" type="text" value="[Employee_Department/Department/"/>	Change Password
Label <input type="text" value="[No attribute selected]"/> 1	
Skill <input type="text" value="[Employee_Skill/Skill/Name]"/>	

Save
Cancel

7. 使用您刚刚创建的新关联，将数据源设置为**部门名称**。
8. 单击**确定**。



9. 将标题更改为**管理部门**。
10. 右键单击小组件并选择**生成选择页面**。
11. 选择用**列表视图选择**，然后单击**确定**。



现在您已经有了这些基本的设置，是时候考虑其他实体和您应为 **KeyUser** 添加的约束了。例如，**Shift** 实体的 XPath 会是什么？那么技能实体呢？有什么区别吗？为了回答这些问题，让我们来推理一下。

部门

既然大家都知道医院有哪些部门，就没有理由不允许 **KeyUser** 访问数据。但是，将访问权限更改为只读是有意义的，因为他们没有更改详细信息的原因。这听起来更像是**管理员**的工作。

PlanningPeriod

由于规划期间与部门相关，在此处限制访问是有意义的。那么问题来了，它是否应该被限制在 **KeyUser** 的部门还是 **KeyUser** 管理的部门。在这种情况下，第二个选项最有意义。

班次

由于班次与 **PlanningPeriod** 紧密相关，此处遵循相同的逻辑是有意义的。

技能

这些技能与医院所有员工共享，因此不应施加任何限制。您还需要一个 **KeyUser** 能够管理这些内容，这样他们就可以在这里拥有读/写权限。

现在您已经确定了您应授予什么权限，是实施的时候了！

12. 将 **KeyUser** 对部门实体的访问权限更改为完全读取，不写入。

13. 更新 **PlanningPeriod** 的 XPath 约束，以基于安全关联限制访问。

Documentation

Rule applies to the following module roles

- Administrator
- Employee
- KeyUser

Select / deselect all

Access rights XPath constraint

Append constraint:

```
[UserManager.PlanningPeriod_Department
/UserManager.Department
/UserManager.Employee_Department_Security
/UserManager.Employee
/UserManager.Employee_Account=' [%CurrentUser%] ']
```

Line	Column	Error
------	--------	-------

OK Cancel

14. 使用上述屏幕截图和您对于 XPath 的知识，为班次、KeyUser 的组合设置类似的 XPath 规则。

由于您现在限制了访问权限，KeyUser 只能访问自己部门的员工；他们无法查看所有员工。这是由您刚才写入的第一个 XPath 引起的。要让他们查看其他部门的员工，请添加一个额外的规则。

15. 单击**新建**添加新规则，选择**员工**作为实体，KeyUser 作为**模块角色**。将默认权限设置为**读取**，并使用标有**读取的全部设为**按钮，将所有属性都设置为读取。

New Access Rule of Entity 'UserManager.Employee'

Documentation

Rule applies to the following module roles

Administrator
 Employee
 KeyUser

Select / deselect all

Access rights XPath constraint

Create and delete rights
 Allow creating new objects Allow deleting existing objects

Member read and write rights
Default rights for new members None Read Read, Write

Member	Access rights
EmployeeId (Integer)	Read
Title (String (200))	Read
Firstname (String (200))	Read
Lastname (String (200))	Read
Gender (Enumeration 'Enum_Gender')	Read
Phone (String (200))	Read
Jobtitle (Enumeration 'Enum_Jobtitle')	Read
Specialty (String (200))	Read

Set all to

祝贺您，您已经配置了您的第一个实体访问约束条件！现在，Summerhill 医院的人可以确定由关键用户管理哪个部门。

讲座 9.4.2：允许用户编辑自己的数据

现在，您的工具箱中已包含此新功能，再次应用，确保员工能够编辑自己的数据。

1. 打开模块安全性，进入**实体访问**标签。
2. 单击**新建**创建一个新规则，选择**员工**实体，然后单击**确定**。
3. 选择**员工**作为模块角色，然后设置所有读/写访问权限。

4. 切换到 **XPath 约束** 标签，然后单击**通往用户的路径**。
5. 打开 **Employee_Account** 关系，然后选择**帐户**实体。单击**选择**。操作完成后，单击**确定**。
6. 设置属性访问权限的时间。使用下面的屏幕截图配置访问权限，然后单击**确定**。

Edit Access Rule of Entity 'UserManager.Employee'

Documentation

Rule applies to the following module roles

Administrator
 Employee
 KeyUser

Select / deselect all

Access rights XPath constraint

Create and delete rights

Allow creating new objects Allow deleting existing objects

Member read and write rights

Default rights for new members None Read Read, Write

Member	Access rights
EmployeeId (Integer)	Read
Title (String (200))	Read, Write
Firstname (String (200))	Read, Write
Lastname (String (200))	Read, Write
Gender (String (200))	Read, Write
Phone (String (200))	Read, Write
Jobtitle (String (200))	Read
Specialty (String (200))	Read
Email (String (200))	Read, Write
HasAccount (Boolean)	Read
UserManager.Employee_Department (UserManager.Department)	Read
UserManager.Employee_Skill (UserManager.Skill)	Read, Write
UserManager.Employee_Department_Security (List of UserManager.Department)	Read
UserManager.Employee_Account (Administration.Account)	Read

Set all to



OK

Cancel

其他需要被赋予 XPath 约束的实体是 **PlanningPeriod** 和班次。对于所有这些实体，您应配置 XPath 规则，将实体限制为相应员工的部门。所有属性都应被设为完全读取，无写入。

7. 打开 **PlanningPeriod** 访问规则，然后根据以下示例设置 XPath。操作完成后，单击确定。

Documentation

Rule applies to the following module roles

- Administrator
- Employee
- KeyUser

Select / deselect all

Access rights XPath constraint

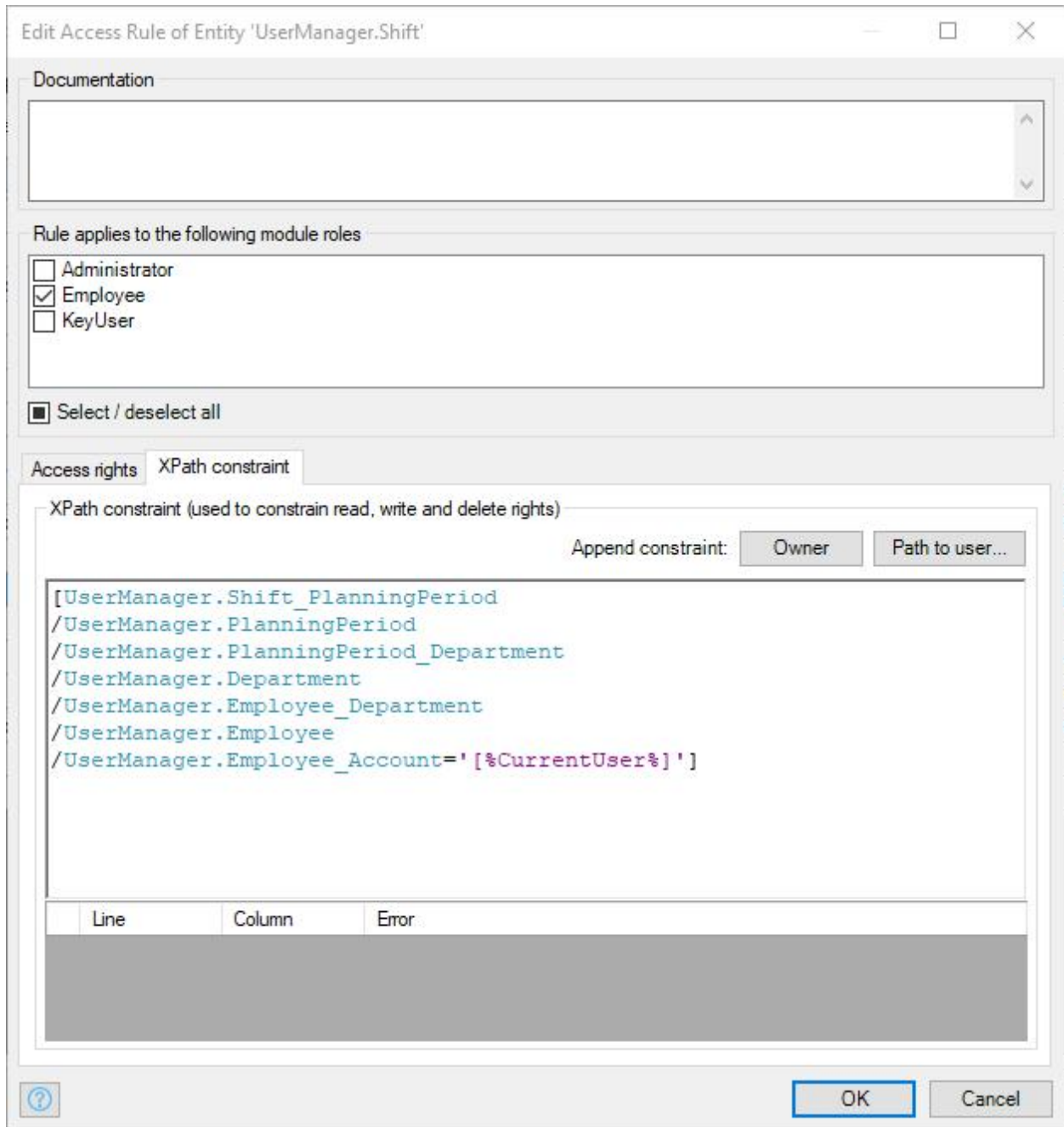
XPath constraint (used to constrain read, write and delete rights)

Append constraint:

```
[userManager.PlanningPeriod_Department
/usermanager.Department
/usermanager.Employee_Department
/usermanager.Employee
/usermanager.Employee_Account='[%CurrentUser%]']
```

Line	Column	Error
------	--------	-------

8. 现在针对班次重复此过程。确保 XPath 使用班次和 **PlanningPeriod** 之间的关系。这样，员工就可以看到自己部门的班次。您可以自己编写 XPath，或者使用通往用户的路径...来设置。



另一个 XPath 约束已配置。好极了！Summerhill 的人们不断地有深刻印象。

总结

又完成了一次讲座，为您的应用程序增加了更多功能。本模块教授了有关 XPath 约束以及如何使用它们来选择特定对象的所有内容。您学会了如何利用关联来做到这一点，您也了

解了 XPath 的亮点以及如何使用它们来编写相当高级的查询。现在，让我们将注意力集中在 Mendix 的一个特别有用的领域：数据报告。