

## XPath

### 1 简介

Mendix XPath 是旨在检索数据的 Mendix 查询语言之一。XPath 使用路径表达式来选择 Mendix 对象及其属性或关联的数据。

XPath 查询既可在 Studio Pro 中写入（例如，当希望对微流活动中检索的数据指定约束时），也可直接在 Java 操作的 .java 文件内的代码中编写。请注意，并非所有运算符都受 Studio Pro 支持，并且 Studio Pro 和 Java 环境之间查询的语法可能有所差异。

XPath 查询的示例有：

- `//Sales.Customer` 检索所有客户。
- `//Sales.Customer[Name='Jansen']` 检索所有名字为“Jansen”的客户。
- `avg(//Sales.Order[IsPaid = true()]/TotalPrice)` 检索所有已付款订单总价的平均值。

在 Studio Pro 中，不用写入完整查询，只需写入约束。实体由上下文隐式确定。因此，只需在客户上下文中写入 `[Name='Jansen']`，而不是 `//Sales.Customer[Name='Jansen']`。

在 Java 中，的确需要写入整个查询，包括双斜杠 (`//`) 和实体名称。

### 2 XPath 元素

常见的 Mendix XPath 查询由几个元素组成。

A	B	C	D
聚合函数（可选）	要检索的实体（必需）	约束（可选）	要检索的属性（可选）
avg	<code>//Sales.Order</code>	<code>[IsPaid = true()]</code>	<code>/TotalPrice</code>

元素 B 描述了每个查询的核心，并包含对被检索对象的描述。该段始终以两个正斜杠“`//`”开头，并包含想要访问的实体的名称，该实体的前面是包含以句点分隔的实体的模块。例如，`//Sales.Customer` 将返回模块“Sales”中实体“Customer”的所有对象。

查询的元素 C 为可选项，并包含一个或多个约束，以限制被检索的数据。

考虑以下查询：

```
//Sales.Customer[Name='Jansen']
```

约束在方括号之间清晰可见，并将检索的对象限制为属性“名字”等于“Jansen”的对象。具有除 Jansen 之外任何其他名字的对象都将从列表中排除。对应于单个查询的可能约束在数量上没有限制。有关如何添加和操控这些约束的更多信息，请参见“XPath 约束”。

查询的元素 D 为可选项，并指定了被检索实体的属性。因为所有数据都存储在对象中，所以该选项在 Studio Pro 自身内部很少使用，这使得处理单个属性的列表变得繁琐且太过复杂。尽管如此，各种 Java 操作仍使用此类列表。此外，该功能可与 A 部分结合使用，以轻松创建某些属性的聚合。

查询的元素 A 为可选项，并指定了聚合。元素 A 可为以下函数之一：avg、count、max、min 和 sum。除了“count”之外，其中每个函数都要求在元素 D 中指定特定属性。

### 3 标记

有关详细信息，请参见“XPath 标记”。

### 4 运算符

有关详细信息，请参见“XPath 运算符”。

### 5 函数

以下 XPath 函数可用：

- XPath 函数：
  - avg
  - count
  - max
  - min
  - sum
- 约束函数：
  - contains
  - starts-with
  - ends-with
  - not
  - true

- false