

正则表达式

1 简介

正则表达式资源文档在实体的验证规则中使用，用于描述字符串必须匹配的一组标准。

正则表达式的属性如下所述。

2 通用

2.1 名称

名称可用于引用实体验证规则中的正则表达式。

2.2 文档

仅用于记录目的；在正在建模的最终用户应用程序中不可见。

3 表达式

表达式定义了以正式的国际标准化正则表达式语言检查字符串的标准。

例如，用于检查荷兰邮政编码的表达式可以是：`[1-9][0-9][0-9][0-9] ?[A-Za-z][A-Za-z]`。

以下是两个邮政编码的示例：**3072AP** 和 **7500 AH**。

这些标准包括：

- 第一个字符是 1 至 9 之间的数字
- 第二、第三和第四个字符是 0 至 9 之间的数字
- 最后两个字符是字母，由最后两个子表达式 `[A-Za-z]` 表示，指明最后两个字符应该在 A-Z 或 a-z 范围内
- 在数字和字母之间可以有一个空格，由空格和问号组成的子表达式表示；问号指明空格为可选

以下各部分总结了可在 Mendix 中使用的正则表达式。该描述也适用于 `isMatch()` 等函数中使用的正则表达式字符串。

3.1 子表达式

正则表达式由一系列子表达式组成。如果字符串的所有部分都以相同顺序与这些子表达式匹配，则字符串与正则表达式匹配。

正则表达式可以包含以下类型的子表达式：

- `[]` - 中括号表达式匹配中括号内指示的单个字符，例如：
 - `[abc]` 匹配 “a”、“b” 或 “c”
 - `[a-z]` 指定匹配从 “a” 到 “z” 范围内的任何小写字母

`{{% alert type= “info” %}}` 这些形式可混合使用：`[abcx-z]` 匹配 “a”、“b”、“c”、“x”、“y” 或 “z”，相当于 `[a-cx-z]`。如果 - 字符是方括号内的最后一个或第一个字符，或者用反斜杠 (\) 转义，则将其视为文字字符。`{{% /alert %}}`
- `[^]` - 匹配不包含在方括号中的单个字符，例如：
 - `[^abc]` 匹配除 “a”、“b” 或 “c” 以外的任何字符
 - `[^a-z]` 匹配从 “a” 到 “z” 中任何非小写字母的单个字符

`{{% alert type= “info” %}}` 如上所述，文字字符和范围可混合使用。`{{% /alert %}}`
- `{m,n}` - 与前一个元素匹配至少 *m* 次且不超过 *n* 次，例如：
 - `a{3,5}` 仅匹配 “aaa”、“aaaa” 和 “aaaaa”
- `{n}` - 与前面的元素正好匹配 *n* 次，例如：
 - `[1-9][0-9]{3} ?[A-Za-z]{2}` 是另一种编写表达式的方法，用于检查上面示例中的荷兰邮政编码
- `.` - 点可匹配任何单个字符；如果要匹配一个点，可以在点前面加上 \ (反斜杠) 进行转义
- 原义字符 - 这是一种在正则表达式语言中没有特殊含义且与自身匹配的字符；这实际上是除 `\[] () { } ^ - $? * + | .` 以外的任何字符，例如：

- 荷兰邮政编码示例中的空格是一个只匹配自身的原义字符

{{% alert type= “info” %}} 如果需要匹配其中一个非原义字符，在其前面加上反斜杠 (\)。 {{% /alert %}}

- \w - 单词：字母、数字或下划线；\w 是 [A-Za-z0-9_] 的缩写
- \d - 数字， [0-9] 的缩写

3.2 量词

子表达式在字符串中出现的次数由子表达式后面的量词表示。如果不存在量词，子表达式必须恰好出现一次。

可使用以下量词：

量词	描述
?	前面的子表达式不应出现或只应出现一次。
*	前面的子表达式可出现多次。
+	前面的子表达式应出现一次或多次。
	没有量词意味着前面的子表达式只应出现一次。