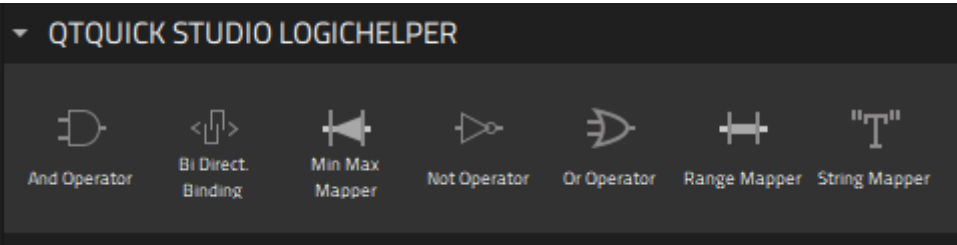


逻辑助手

若要让 UI 执行某些操作，可能需要为条件编写 JavaScript 表达式或将数字转换为字符串。为了简化此操作，Qt Design Studio提供了一组称为**逻辑助手**的组件。



逻辑帮助程序可用于使用布尔 AND、NOT 和 OR 运算符绑定属性值，以及映射数字和数字范围。此外，还可以双向同步两个组件的属性值。

逻辑帮助程序是不可见的组件，可用于控件，如**滑块**或**复选框**。要使用逻辑帮助程序，请将其从**组件**>Qt Quick Studio **逻辑帮助程序**拖放到**导航器**。如果在**组件**中找不到逻辑帮助程序，则需要将Qt Quick Studio **逻辑助手**模块添加到项目中，如**添加和删除模块**中所述。

以下部分更详细地介绍了不同类型的逻辑帮助程序。


布尔助手

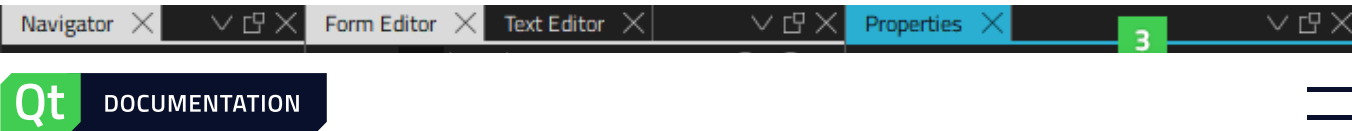
可以使用逻辑帮助程序通过布尔 AND、OR 和 NOT 运算符绑定属性值。

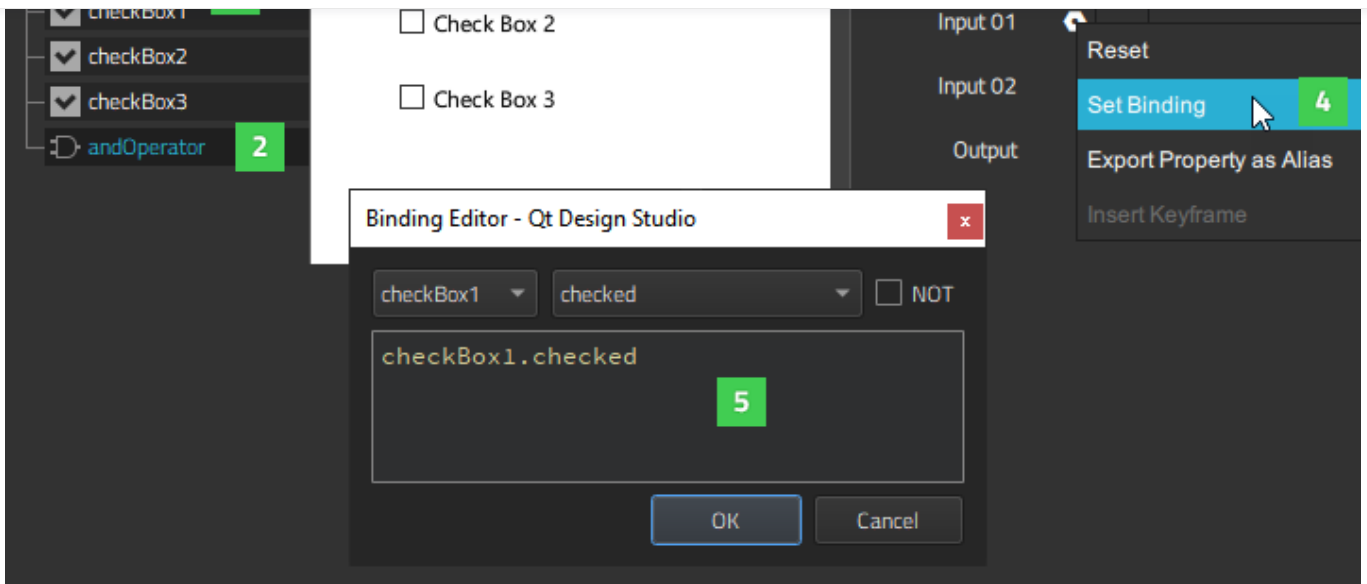
和运算符

And 运算符组件计算两个布尔输入。对输出的计算就像两个输入都是一样。true>true

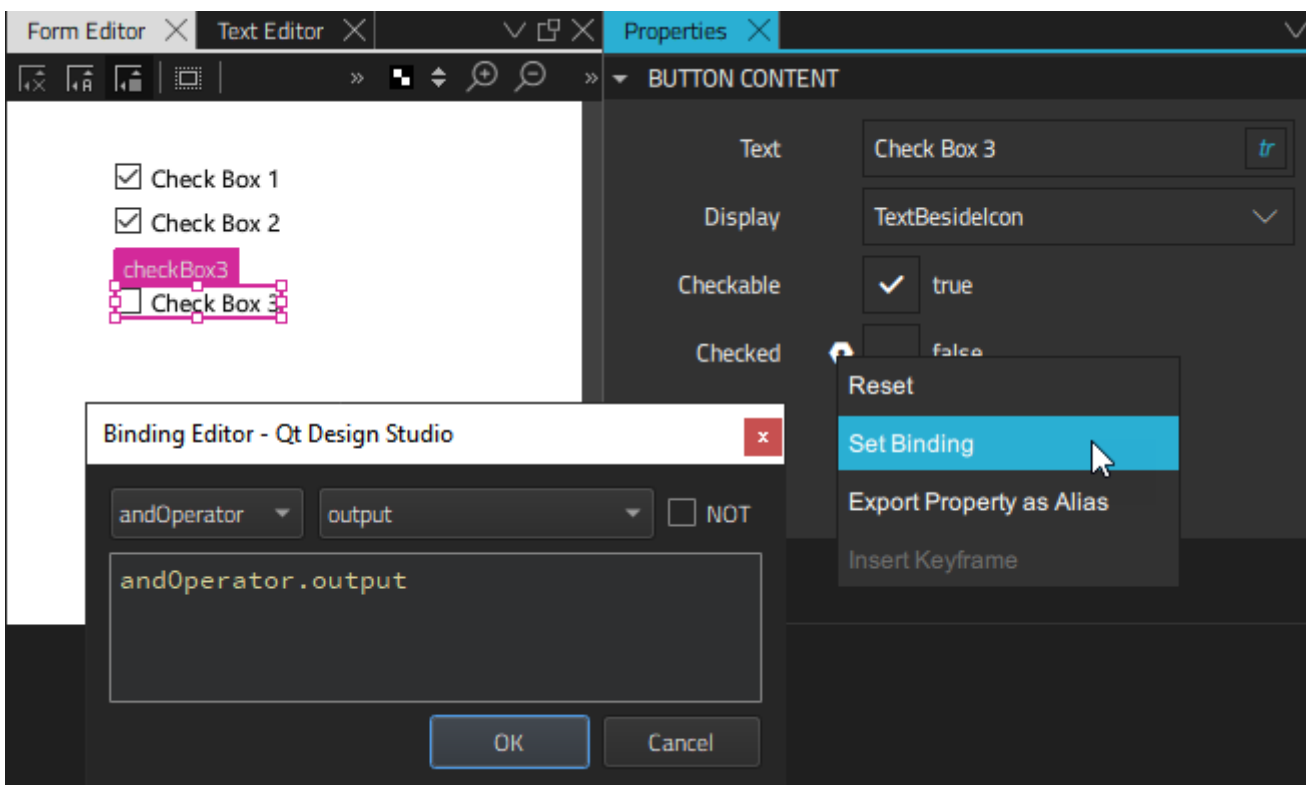
例如，我们可以使用两个复选框的选中状态来确定第三个复选框的选中状态。首先，我们将**复选框**组件的三个实例和**And 运算符**组件的一个实例拖放到**Navigator**（1）中。然后，我们选择**And 运算符**组件实例（2）并在属性（3）中设置其**属性**。

我们选择  输入01字段旁边的以显示“**操作**”菜单，然后选择“设置绑定（4）”以打开“**绑定编辑器**”（5）。在那里，我们将 AND 运算符的属性值绑定到第一个复选框的属性值。然后，我们在**输入 02**字段中执行相同的操作，在该字段中我们将属性绑定到第二个复选框的属性。input01checkedinput02checked

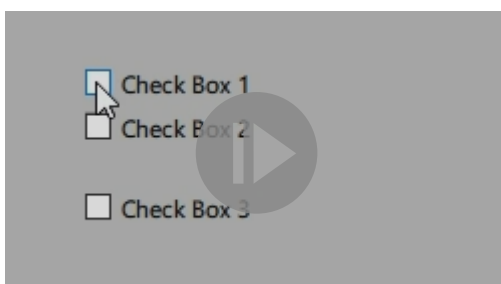




最后，我们选中第三个复选框，并将其Checked属性绑定到 AND 运算符的Output属性。



当我们预览UI时，所有复选框最初都是未选中的。但是，当我们选中第一个和第二个复选框时，第三个复选框也会被选中。

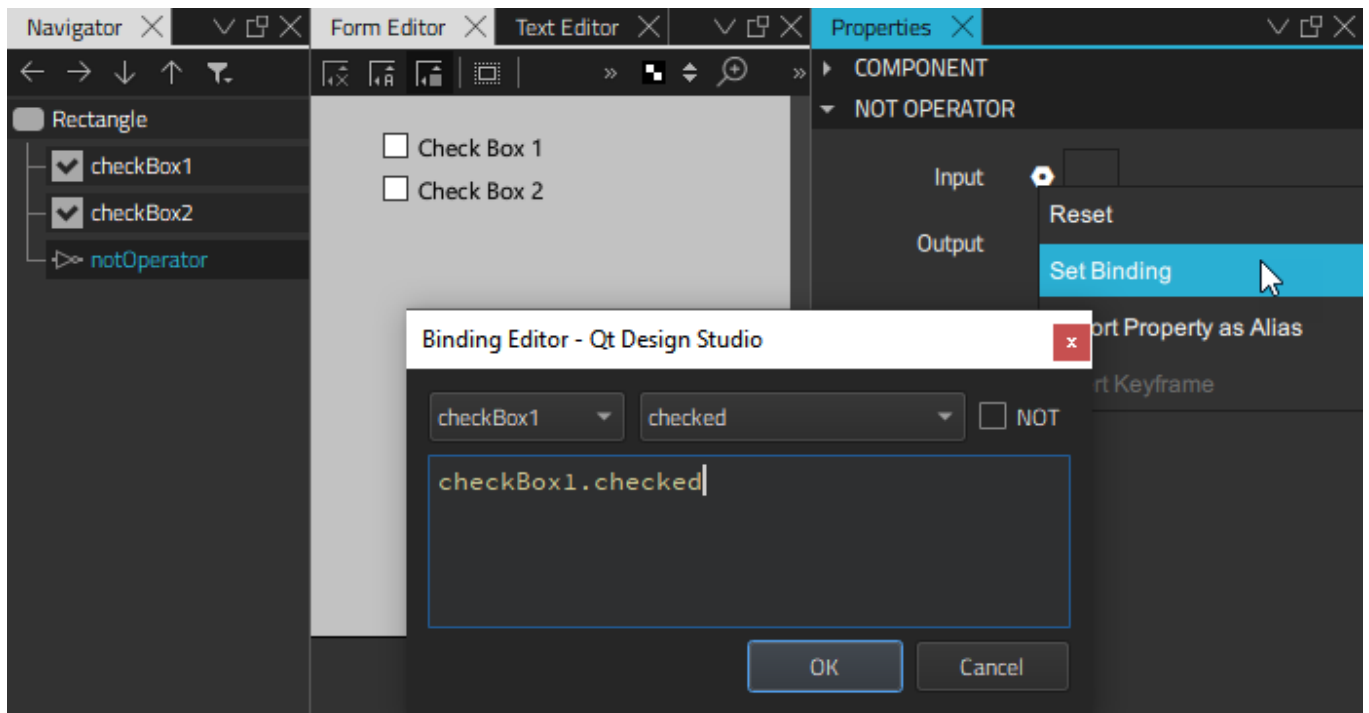


手术室运算符

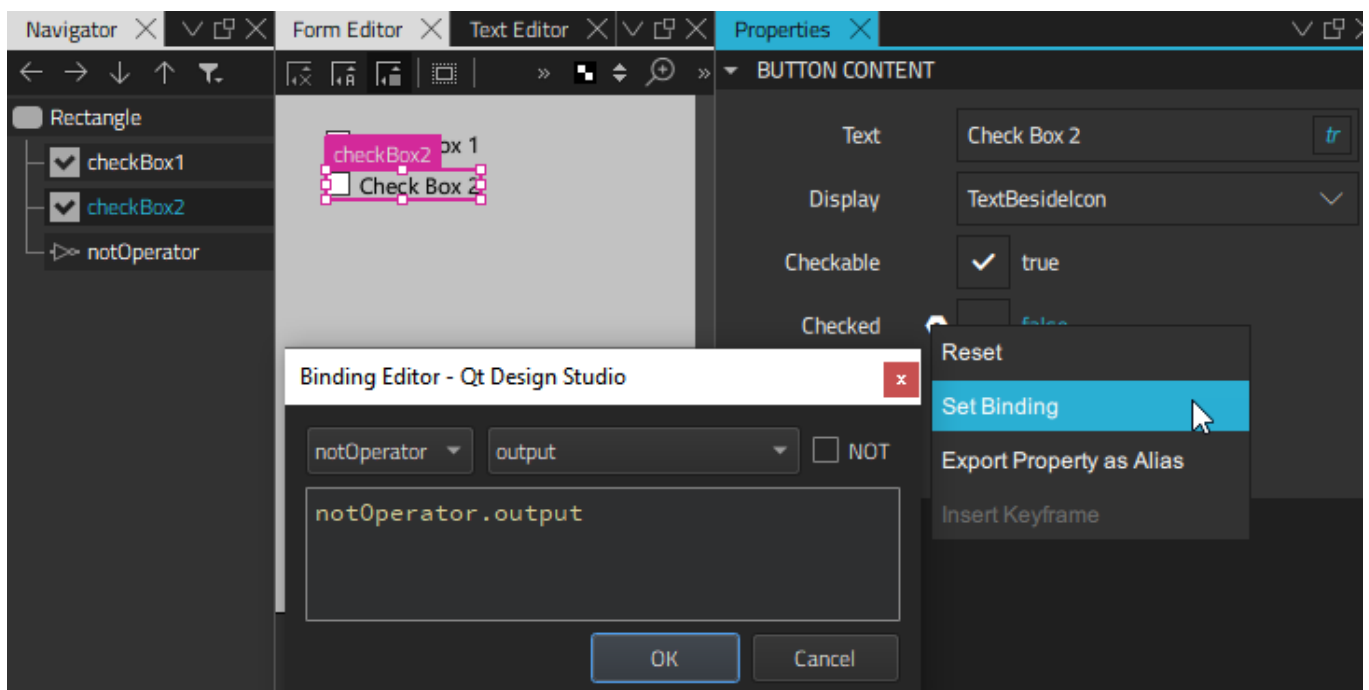
非操作员

如果不满足条件，则计算“**非运算符**”组件。true

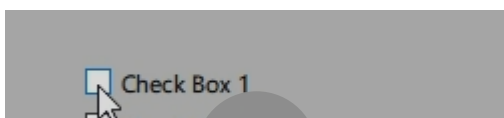
例如，我们可以指定如果选中一个复选框，则无法选中另一个复选框。首先，我们将**复选框**组件的两个实例和**Not Operator**组件的一个实例拖放到**导航器**。然后，我们选择**非运算符**组件实例并在**属性**中设置其**属性**。在**绑定编辑器**中，我们将 NOT 运算符的属性值绑定到一个复选框实例的属性值。inputchecked



然后，我们选择另一个复选框实例，并将其“**已检查**”字段的值绑定到“**非运算符**”组件的“**输出**”字段的值。



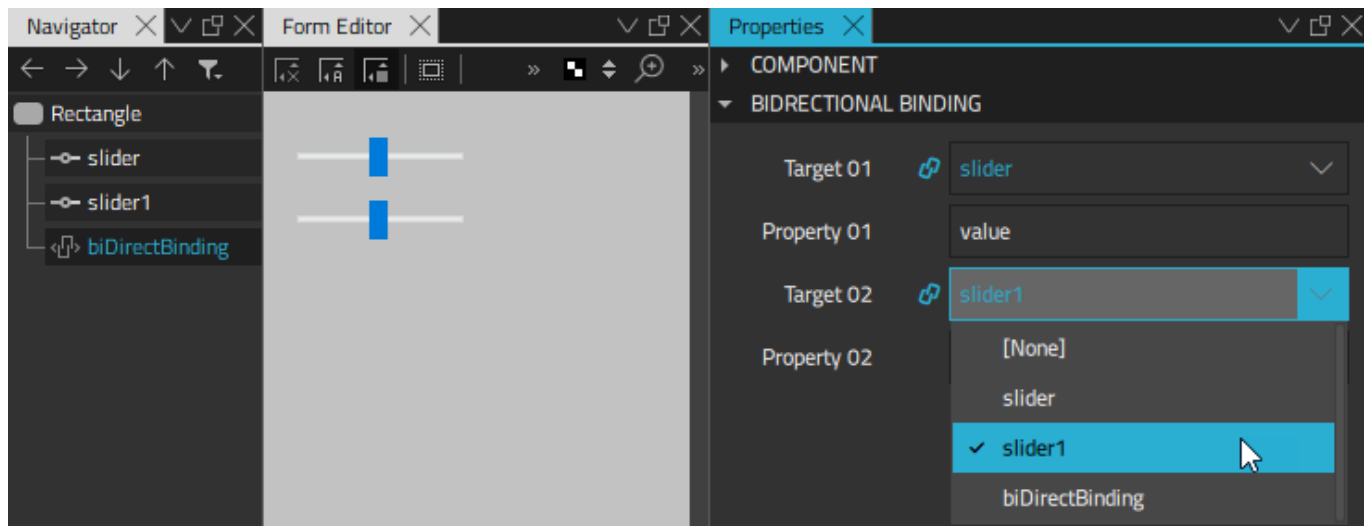
当我们预览 UI 时，最初会选中第二个复选框。但是，当我们选中第一个复选框时，第二个复选框会自动清除。



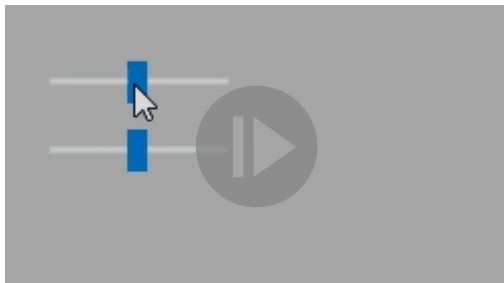
双向绑定

毕直通。绑定组件将两个控件的值绑定在一起，以便在更改一个值时，另一个值紧随其后。此组件可用于同步两个滑块或一个滑块和复选框。通常，它用于将后端值绑定到控件，例如滑块。

例如，要同步两个滑块的值，我们拖放了**滑块**组件的两个实例和一个Bi Direct **实例**。将组件绑定到**导航器**中的同一父组件。然后，我们选择双向绑定实例并在**属性**中设置其属性。



在**目标01**和**目标 02**字段中，输入要绑定在一起的组件的 ID。在**属性01**和**属性 02**字段中，输入要同步的属性的名称。在我们的示例中，我们将两个滑块组件的属性绑定在一起，以便当我们在预览中移动一个滑块手柄时，另一个滑块手柄也会随之移动。value



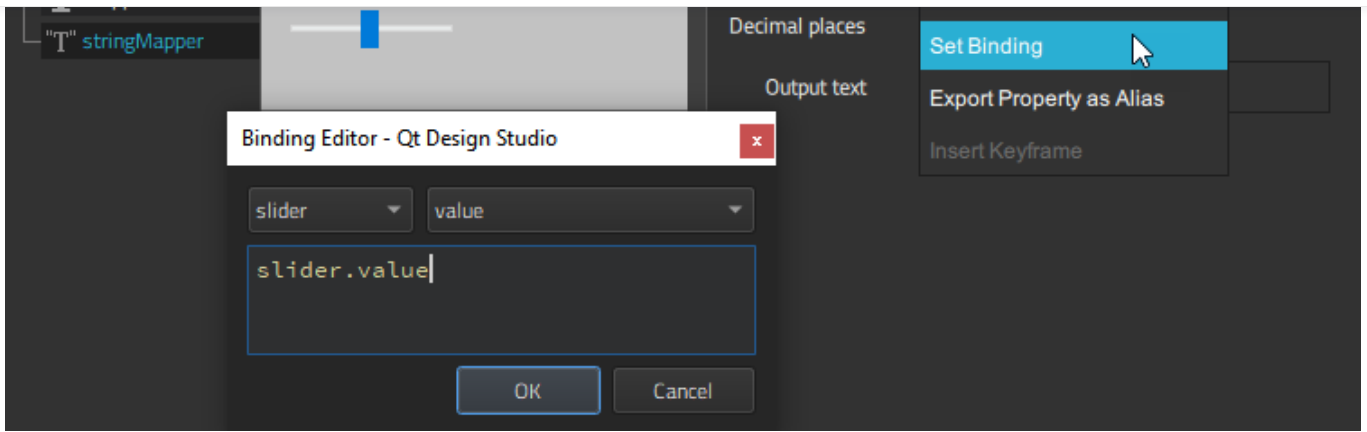
如果要添加显示滑块值的文本字段，可以使用**字符串映射器**组件。

字符串映射器

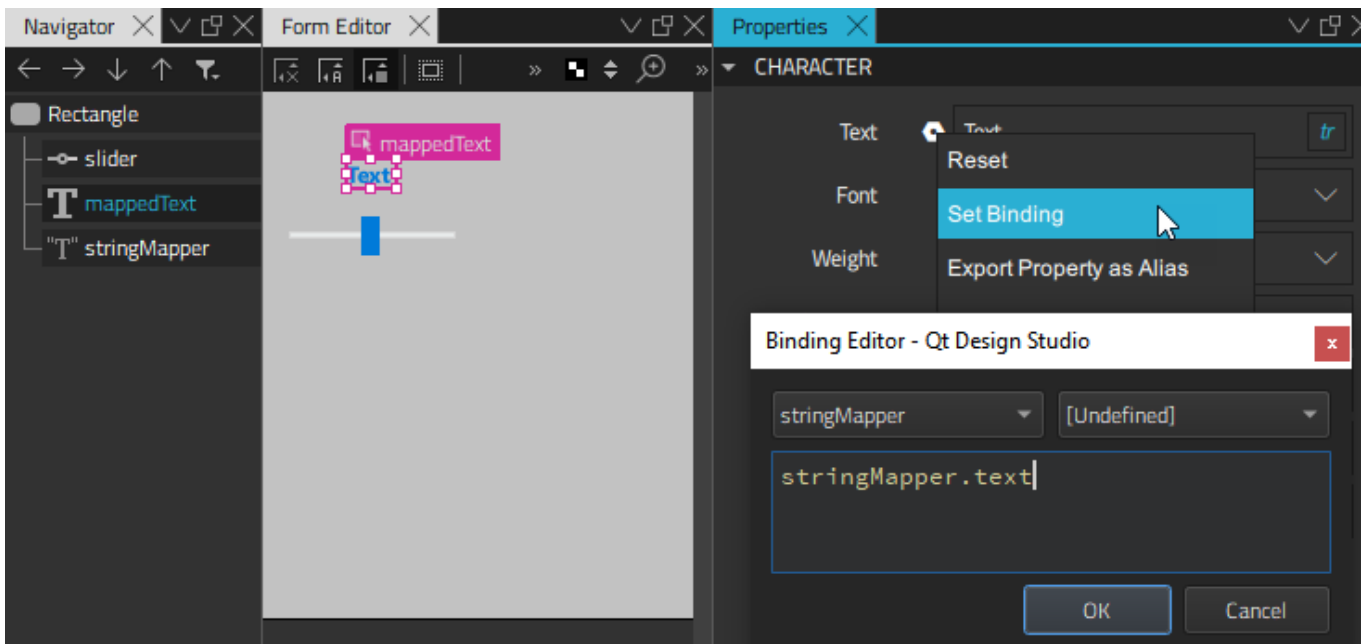
字符串映射器组件将数字映射到字符串。首先，在字符串映射器属性和要从中获取值的控件的属性之间**添加一个绑定**。然后，在字符串映射器的属性和将显示字符串的组件的属性之间添加绑定。inputvaluetext

例如，要使用文本组件显示滑块的值，我们将**文本**、**滑块**和**字符串映射器**组件拖放到同一父组件。然后，我们在**导航器**中选择**字符串映射器**实例以在属性中显示其**属性**。在那里，我们将**输入**字段的值绑定到Slider实例的属性值。value

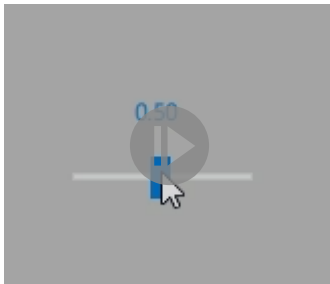




接下来，我们选择 Text实例，并将Text字段的值绑定到字符串映射器组件的输出文本字段（属性）的值。text



当我们在预览中移动滑块手柄时，文本组件中显示的值会相应更改。



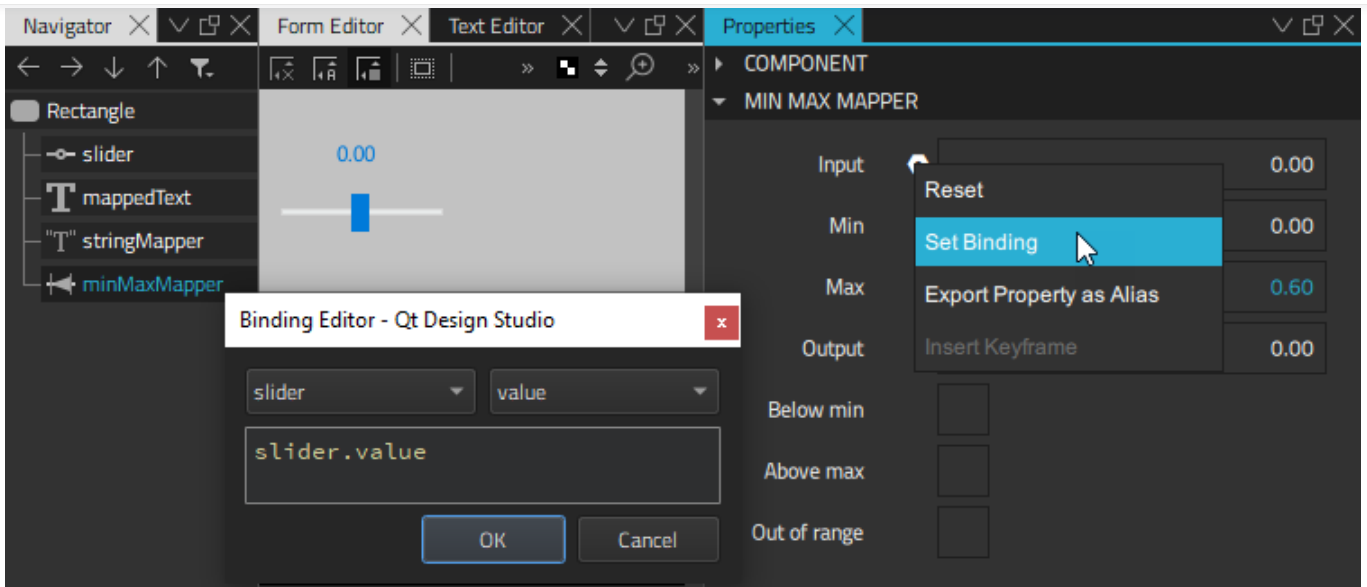
“小数位数”字段的值确定小数分隔符后的位数。

最小-最大映射器

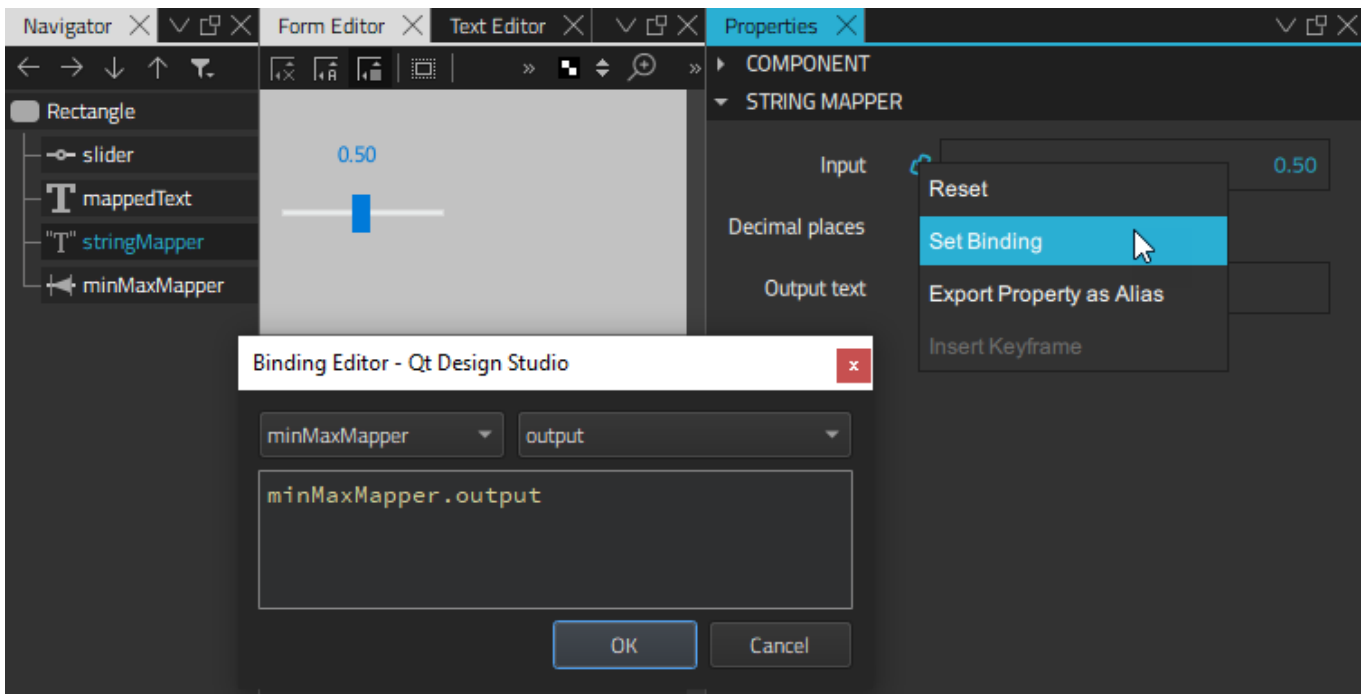
最小最大映射器组件具有输出值，即使输入值超出范围也是如此。这使您能够对值应用操作，例如更改状态中的颜色，即使它们低于最小值或高于最大值。

若要访问控件的值，请将最小-最大映射器的Input属性绑定到控件属性的属性。value

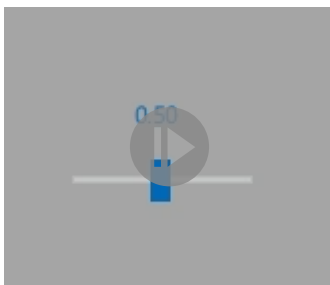
例如，要将滑块的最大值限制为 0.60，而不考虑滑块属性中设置的最大值，我们将**最小最大映射器**拖放到上面



要使文本组件显示最大值，我们选择字符串映射器组件并更改我们设置为输入字段 fromto 值的绑定。
`slider.valueminMaxMapper.output`



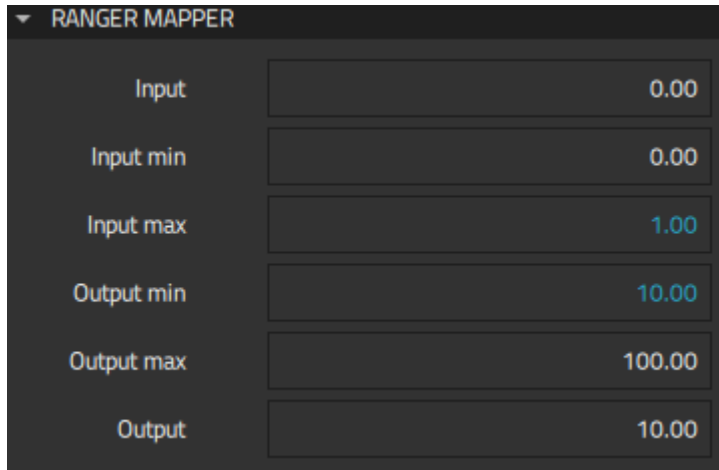
当我们在预览中移动滑块手柄时，它只会向上移动到值 0.60。



如果输入字段的值超出范围，则设置超出范围，高于最大值和低于最小值复选框。true

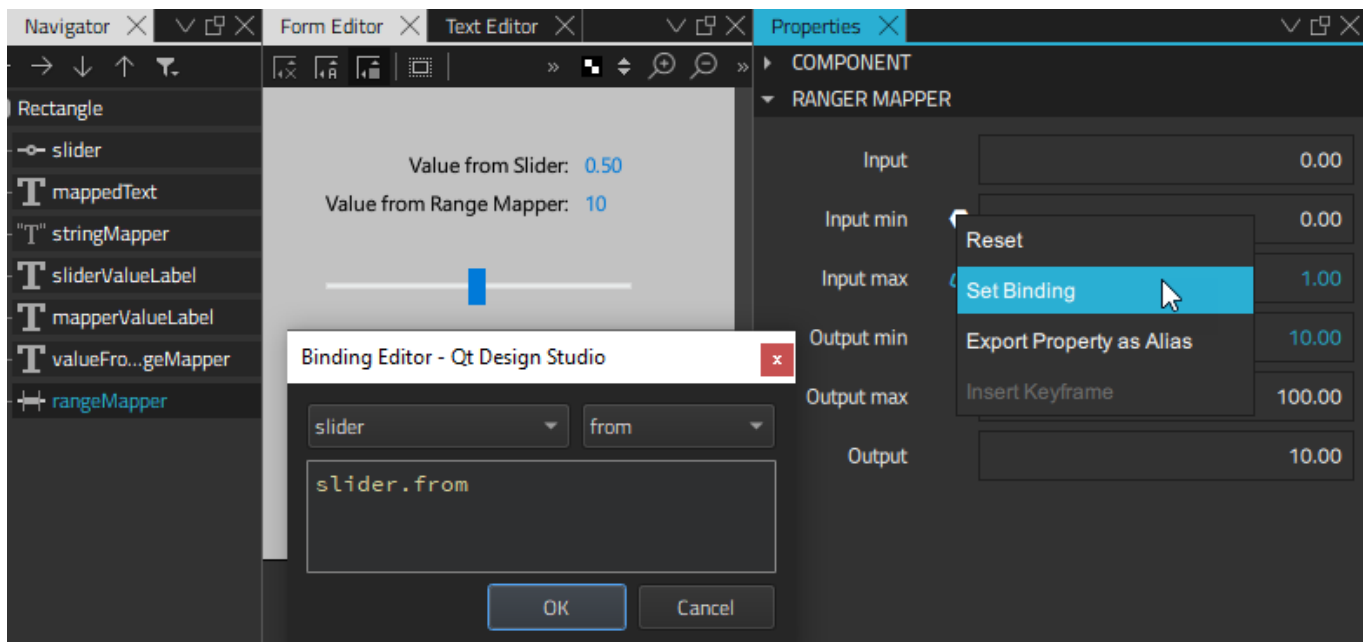
例如，在速度的上下文中，高于最大存在意味着太快，而低于最小值意味着太慢，超出范围意味着太快或太慢。true true true

“范围映射器”组件将一个数值范围映射到另一个范围，以便第二个范围的输出值跟随原始范围的输入值。例如，在重新映射时间轴上的当前帧时，这很有用。

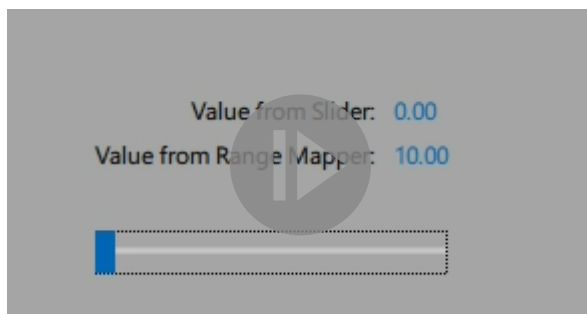


在输入最小值、输入最大值、输出最小值和输出最大值字段中指定最小和最大输入和输出值，并在输出字段中指定原始值。

例如，我们可以指定滑块的值范围从 0 到 1。如果我们想显示从 10 到 100 的值，我们可以将滑块组件的“从”和“到”字段的值绑定到范围映射器组件的“输入最小值”和“输入最大值”字段的值。然后，我们将“输出最小值”字段的值设置为 10，将“输出最大值”字段的值设置为 100。

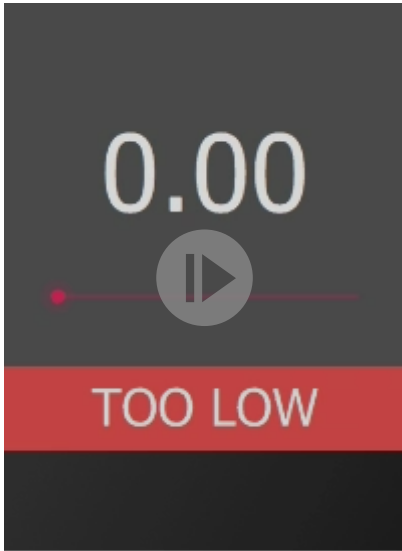


当我们在预览中移动滑块手柄时，滑块组件的输入值从 0 更改为 1，输出值从 10 更改为 100。



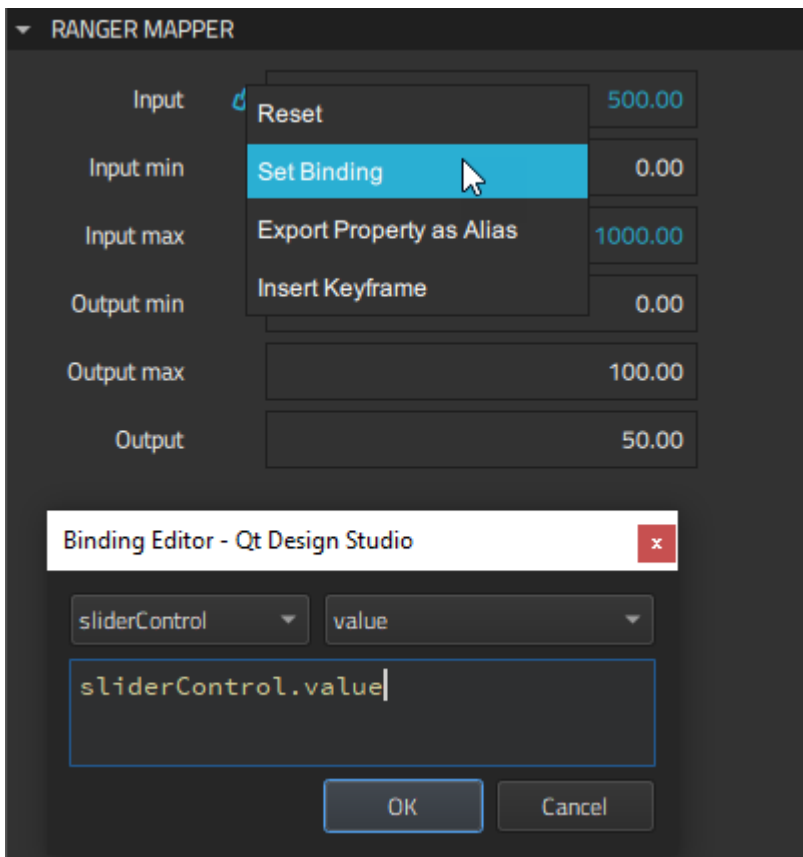
可以组合多个相同类型或不同类型的逻辑帮助程序来定义应用程序逻辑。

例如，我们创建了一个用于销售商品的小型应用程序。我们使用范围映射器组件来设置价格**范围**，并使用**最小最大映射器**组件来创建价格过低或过高的位置，以及价格低于或超过现行值的`badValueRange`。然后，我们使用**And 运算符**组件来确定该值是低于最小值还是高于最大值。`blockedRange`



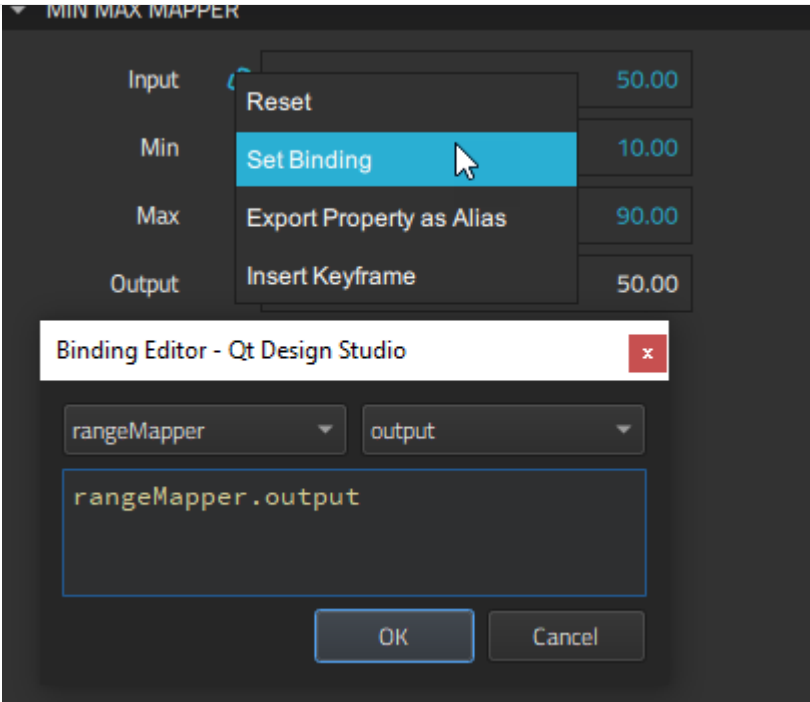
我们使用一个字符串映射器组件将滑块值映射到显示价格的**文本组件**，如**字符串映射器**中所述。

要定义从 0 到 1000 的价格范围，我们将范围**映射器**组件的Input属性绑定到滑块值，并将Input max字段中价格范围的最大输入值设置为 1000。默认情况下，最小输入值为 0，因此我们不需要更改它。

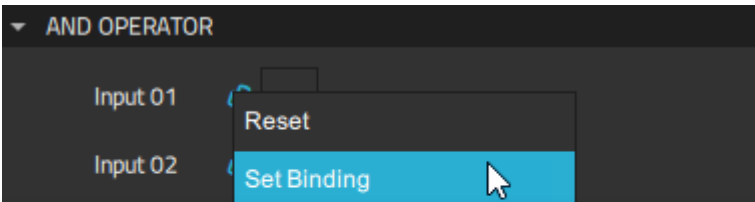
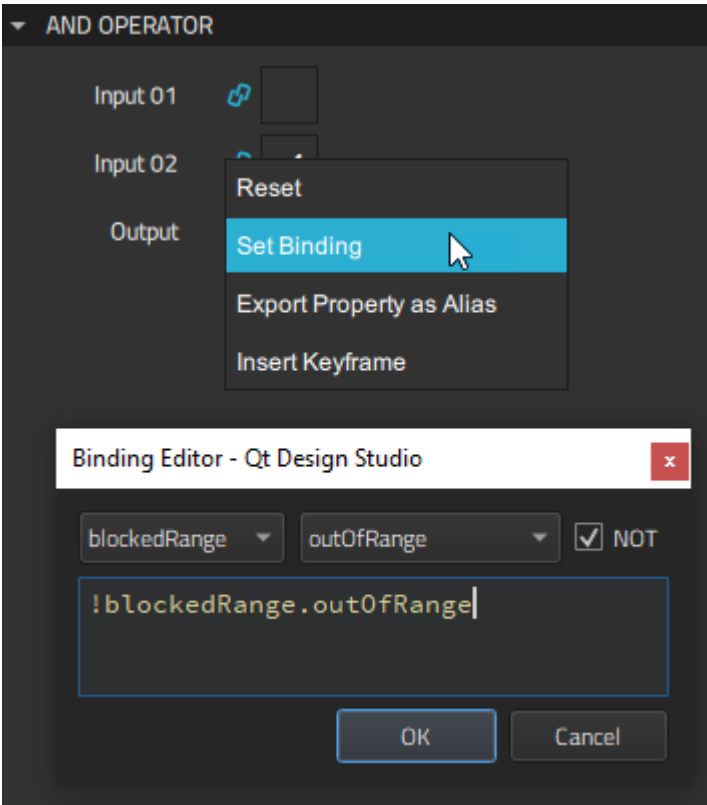


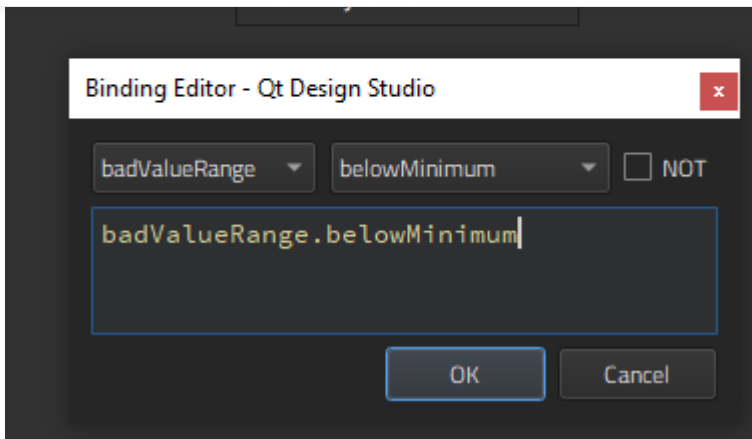
接下来，我们使用两个**最小最大映射器**组件创建一个阻止范围，其中销售按钮将被隐藏，以及一个**错误值范围**，通过更改销售提示的文本和颜色来**阻止**销售。

对于阻止的范围，我们将最小-最大映射器的Input属性绑定到**范围映射器**组件的**输出值**，并在Max字段中指定

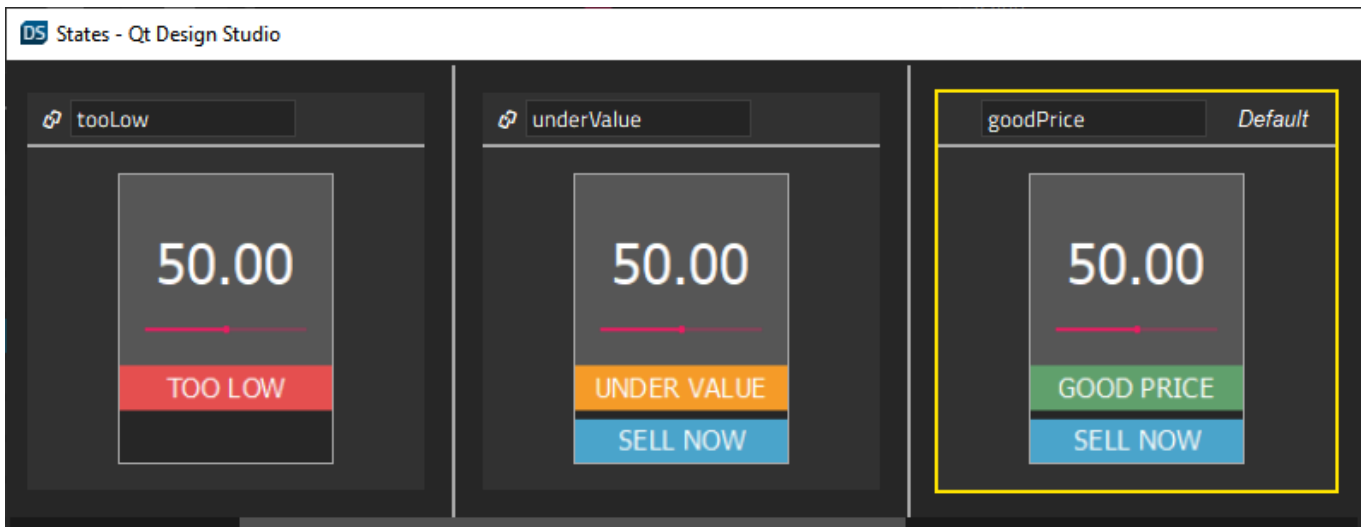


我们使用两个And 运算符组件来确定当值在阻止范围内时应隐藏卖出按钮。为此，我们将**输入 02**字段的值绑定到最小-最大映射器的**超出范围**字段的值评估。我们指定当值不超出范围时，它的计算结果为`true`。

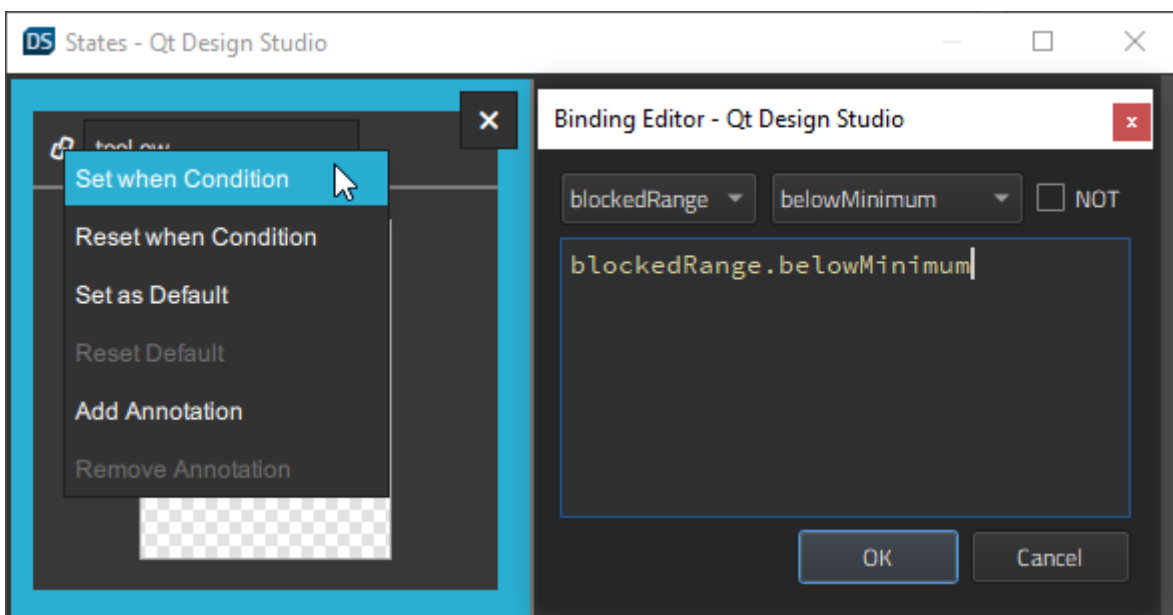




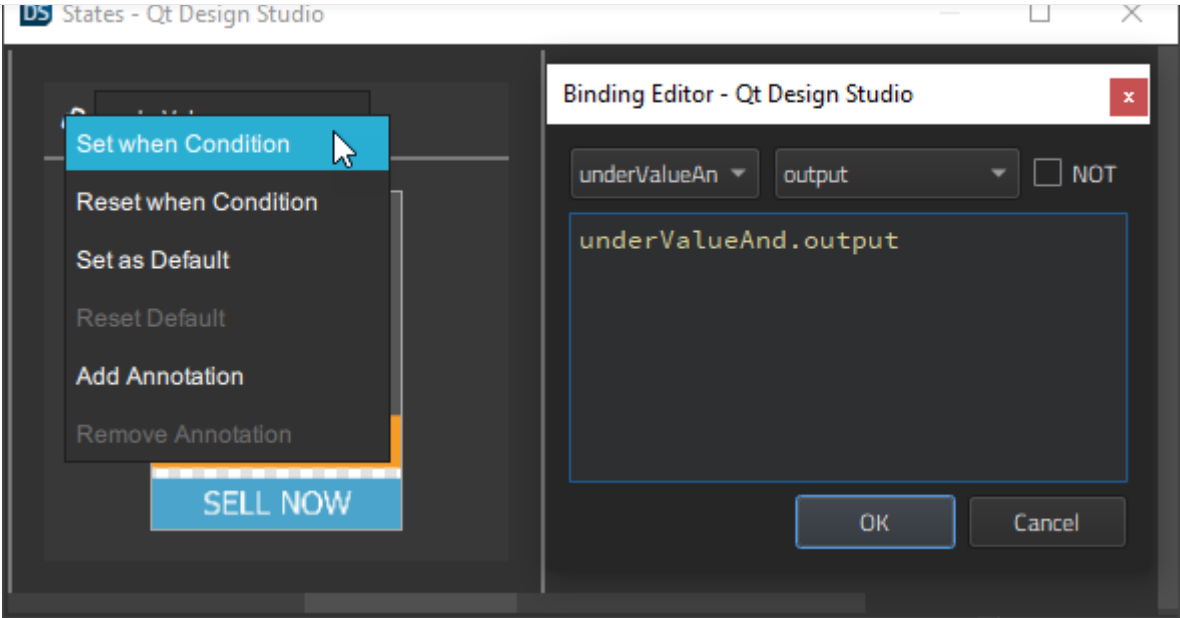
现在，我们可以评估**最小最大映射器和运算符**组件的值，以应用显示不同按钮文本和销售提示的不同**状态**。为此，我们创建状态并为它们设置条件。when



首先，我们创建一个`tooLow`状态，并设置一个条件，以便在阻止范围的最小映射器组件的**最小值**计算结果为时应用它。when true










对于刻度另一端的`tooHigh`状态，我们设置条件，以便在阻止范围的**最小最大映射器**组件的“**高于最大值**”字段的值计算结果时应用它。when true



对于 *overValue* 状态，我们设置条件以在 *overValueAnd* 组件的**输出**字段的值计算结果时应用它。when true
当我们预览应用程序时，我们可以看到滑块值更改时应用的状态。

逻辑帮助程序摘要

下表汇总了可用的逻辑帮助程序。

图标	逻辑助手	描述
	和运营商	布尔和。
	毕直接。捆绑	一种双向绑定，用于在两个方向上绑定两个值并使它们保持同步。
	最小最大映射器	将一个数字映射到具有最小值和最大值的另一个数字。
	不是操作员	布尔不是。
	或操作员	布尔或。
	范围映射器	将一个数值范围映射到另一个范围，以便第二个范围的输出值跟随原始范围的输入值。
	字符串映射器	将数字映射到具有定义精度的字符串。

[< 2D 效果](#)

[动画 >](#)



联系我们

公司

关于我们
投资者
编辑部
职业
办公地点

发牌

条款和条件
开源
常见问题

支持

支持服务
专业服务
合作 伙伴
训练

对于客户

支持中心
下载
Qt登录
联系我们
客户成功案例

社区

为Qt做贡献
论坛
维基
下载
市场

© 2022 Qt公司

反馈 登录