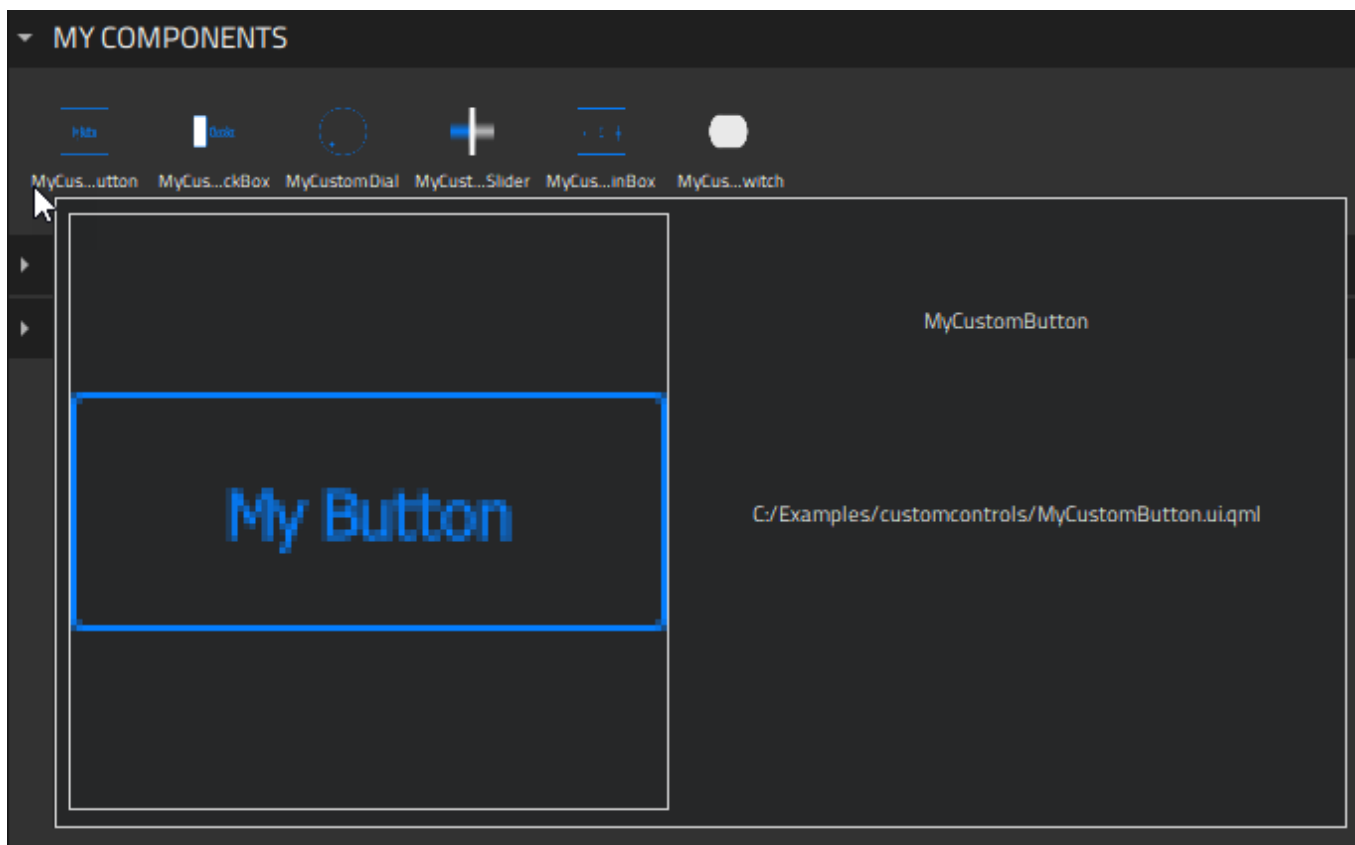


Qt设计工作室手册 > [创建自定义组件](#)

# 创建自定义组件

可以使用项目向导模板创建自定义组件和控件，也可以将组件实例移动到单独的文件中，以将它们转换为可以创建实例的新组件。然后，您可以在其他组件中使用新组件的实例。



自定义组件存储在**组件 > 我的组件**中。您可以通过将自定义组件从**组件**拖放到**2D**、**3D**或**导航器**视图来创建自定义组件的实例。

## 从头开始创建组件

要使用向导模板创建自定义组件，请执行以下操作：

1. 选择**文件 > 新建文件 > Qt 快速文件 > Qt 快速 UI 文件 > 选择**创建新文件。 .ui.qml

注意：仅当文件名以大写字母开头时，组件才会列在“组件，我的组件”中。

2. 单击 **设计** 以在 2D 视图中打开文件。
3. 将组件从**组件**拖放到**导航器**或2D视图。
4. 在“**属性**”视图中编辑组件属性。可用属性取决于组件类型。您可以在“**连接**”视图的“**属性**”选项卡上**添加组件的属性**。
5. 要以Qt Design Studio默认不支持的方式更改组件实例的外观和行为，您可以在“连接”视图的“属性”选项卡上定义自定义**属性**。

以下各节包含有关如何使用 2D 视图编辑**2D**内容和 3D 视图编辑**3D**场景的详细信息，以及如何使用基本组件的实例创建 UI 控件的示例：

- › 二维视图
- › 3D视图
- › 创建按钮
- › 创建可缩放的按钮和边框

## 命名约定

建立命名约定以保持 UI 中的组件井井有条。准确命名组件并为其提供合适的 ID。特别是，检查并修改从设计工具导出的零部件的默认名称，以创建遵循您建立的命名约定的可靠且不言自明的名称。

例如，按钮符号可能具有 ID`myButton_symbol`，而不同的图层可能将其作为前缀并附加状态，从而生成 ID`myButton_symbol_default`。若要为项目后面的其他鼠标区域做好准备，可以对它们使用类似的 ID，例如 `myButton_symbol_hotspot`。当您将该按钮用作屏幕中的实例时，应为每个实例提供一个与该屏幕相关的唯一 ID。例如，`myButton_myMenu_home`、`myButton_myMenu_profile` 和 `myButton_myMenu_settings`

## 创建自定义控件

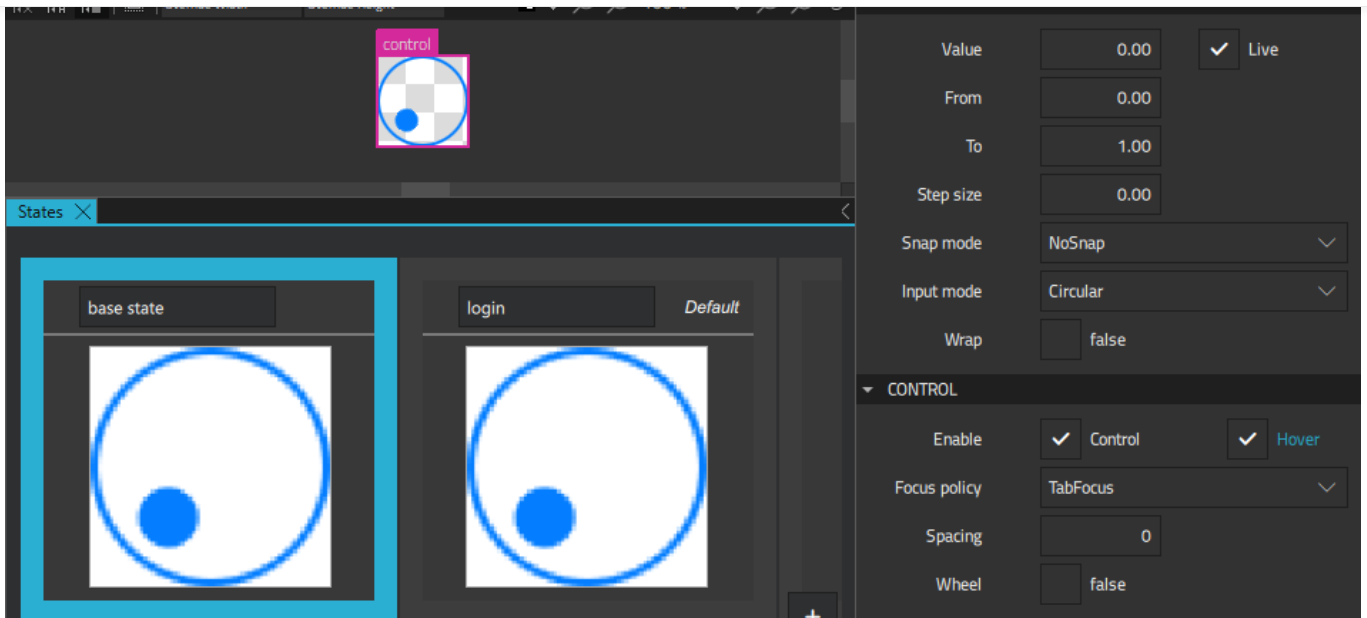
您可以使用项目向导模板根据Qt快速控件模块中的组件创建可样式化的UI控件：

<b>按钮</b> 	<b>复选框</b> 	<b>拨号</b> 
<b>滑块</b> 	<b>旋转盒</b> 	<b>开关</b> 

您可以在所有预设**状态**下编辑控件的属性，以将自己的样式应用于它们。

**注意：**对于按钮和复选框，可以通过选择“**属性**>**控件**”，然后禁用“**悬停**”复选框来禁用行为异常的悬停效

Qt设计工作室手册3.8.0  
Topics >



创建可设置样式的 UI 控件：

1. 选择“**文件>新建文件>文件和类>Qt 快速控件**”。
2. 选择要创建的控件，然后选择“**选择**”。

**注意：**仅当文件名以大写字母开头时，组件才会列在“**组件>我的组件**”中。

3. 在“**属性**”视图中编辑组件属性。

可用属性取决于组件类型。您可以在 {连接} 视图的“**属性**”选项卡上**添加组件的属性**。

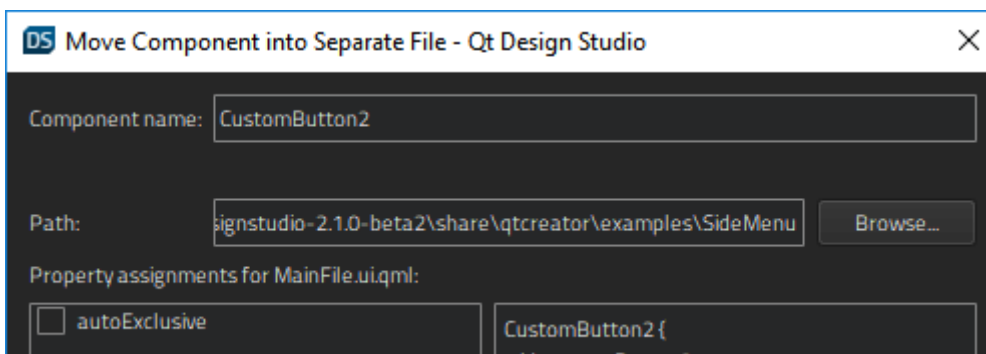
有关使用Button模板创建按钮并设置其样式的示例，请参阅[登录 UI - 组件教程中的创建按钮](#)。

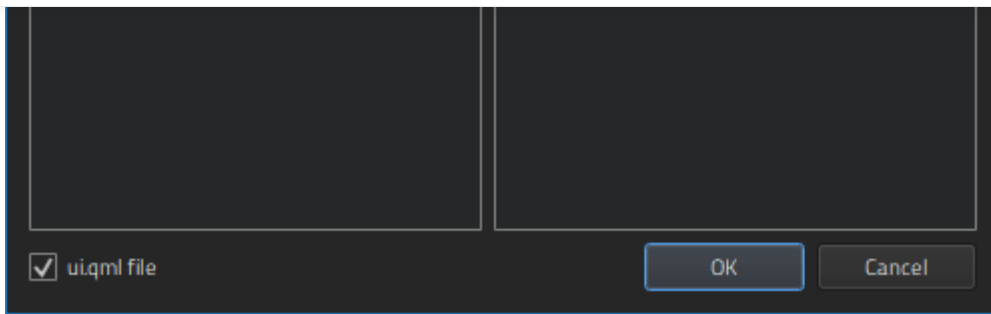
此外，您还可以为不同类型的屏幕创建起点：

- › [窗格](#)
- › [堆栈布局](#)
- › [滑动视图](#)

## 将组件实例转换为自定义组件

创建可重用组件的另一种方法是通过将组件实例移动到单独的组件文件（.ui.qml）中，将组件实例转换为自定义组件。右键单击“**导航**”或2D视图中的组件实例，然后在上下文菜单中选择“**将组件移动到单独的文件中**”。





为新组件命名，并选择是为新组件还是为原始组件设置属性。

选择“确定”时，将创建一个新的组件文件，并将组件的实例添加到当前组件文件的代码中。UI 在 2D 视图中的外观不会更改。

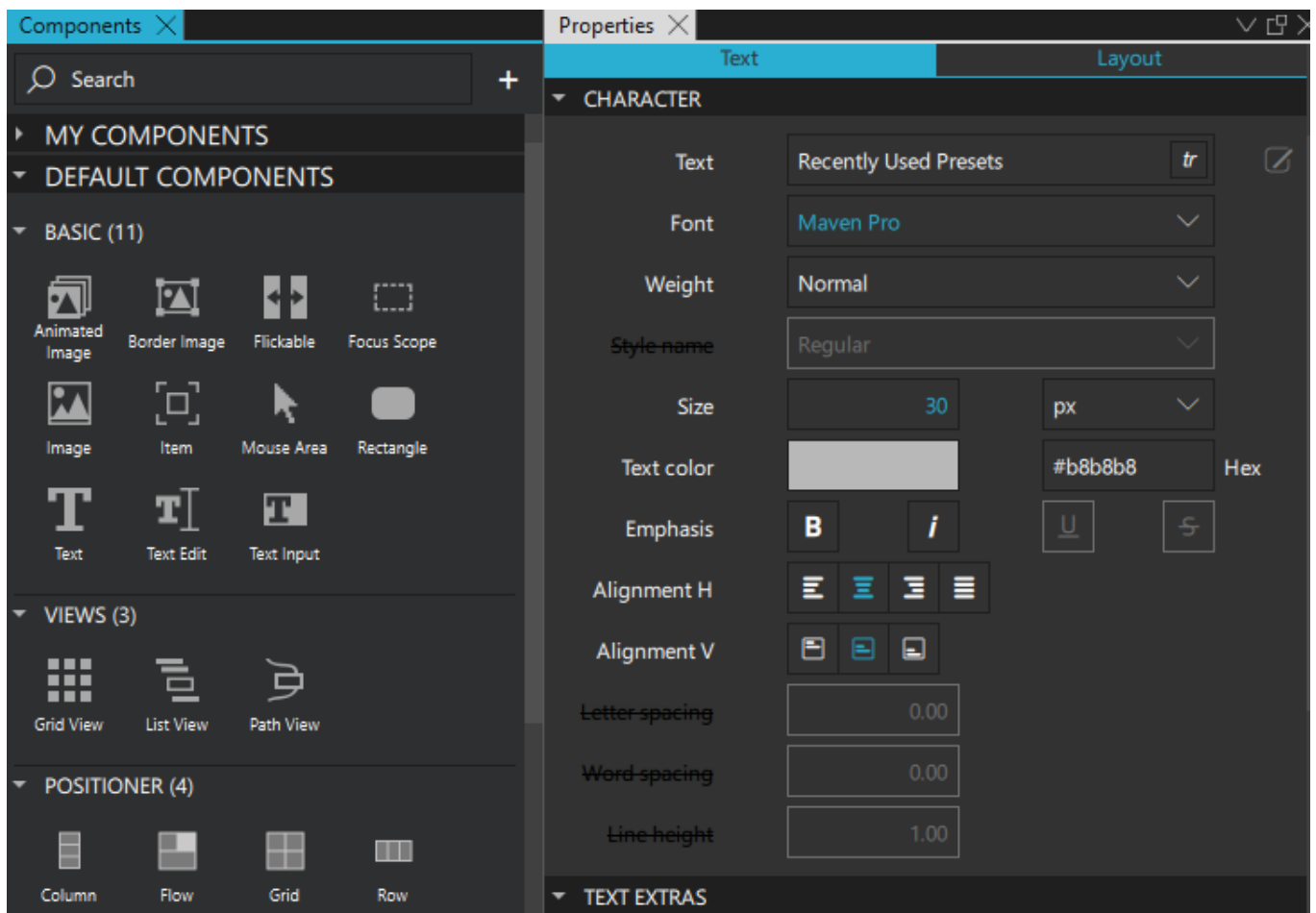
若要打开新的组件文件以编辑要为组件的所有实例更改的属性，请右键单击该组件，然后在上下文菜单中选择“进入组件”。有关打开基本组件的其他方法，请参见在[组件内移动](#)。

有关创建可重用自定义组件的示例，请参阅[进度栏](#)。自定义组件列在[组件 > 我的组件](#)中，您可以使用它们的实例来构建更多组件。

## 为 MCU 创建用户界面

Qt for MCU 使您能够使用组件子集为由微控制器单元（MCU）供电的设备创建 UI。支持的组件子集取决于您用于开发的 Qt for MCU 版本。在本手册中，我们指出了在撰写本文时支持哪些组件。

要针对 MCU 进行开发，请[创建一个 MCU 项目](#)。只有 MCU 上可用的组件才会显示在[组件](#)中。支持的组件仅支持属性的子集。MCU 上不可用的属性在“属性”视图中用删除线文本标记。



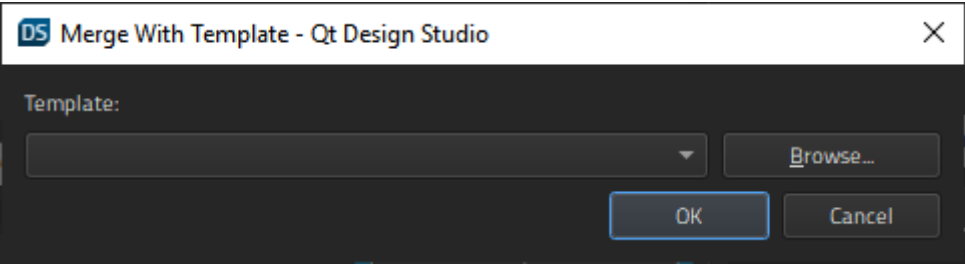


有关支持的组件及其属性的更多信息，请参阅[Qt for MCU - 所有 QML 类型](#)。

## 将文件与模板合并

您可以使用第二个文件将当前组件文件与现有的第二个组件文件合并，其方式类似于使用 CSS 样式表。

要使用此实验性功能，请右键单击“**导航器**”或“**二维**”视图中的组件，然后在上下文菜单中选择“**将文件与模板合并**”。



在 模板 字段中，选择要用作**模板**的文件。

[◀ 创建组件实例](#)

[创建按钮 ▶](#)



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

专业服务  
合作 伙伴  
训练

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