Qt DOCUMENTATION
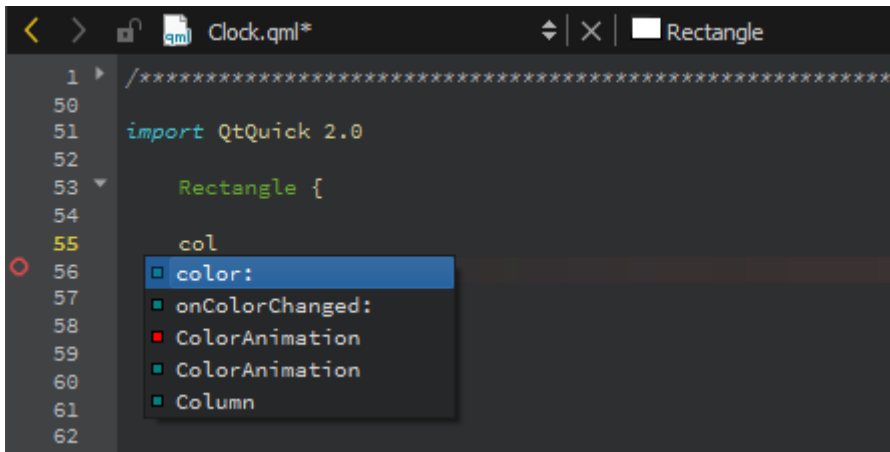
🔍 搜索

Qt设计工作室手册 3.7.0

Topics ＞

Qt设计工作室手册 ＞ 完成代码

完成代码

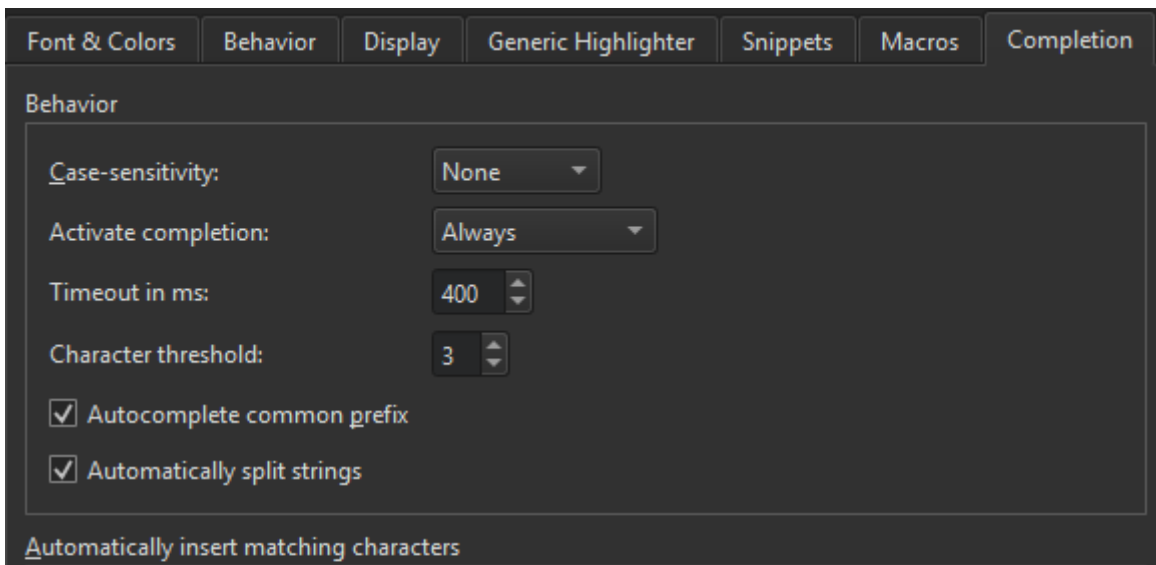在编写代码时，Qt 设计工作室会建议属性、ID 和代码片段来完成代码。它提供了对当前游标下的语句的建议列表。按 **Tab 或 Enter** 接受所选建议并完成代码。
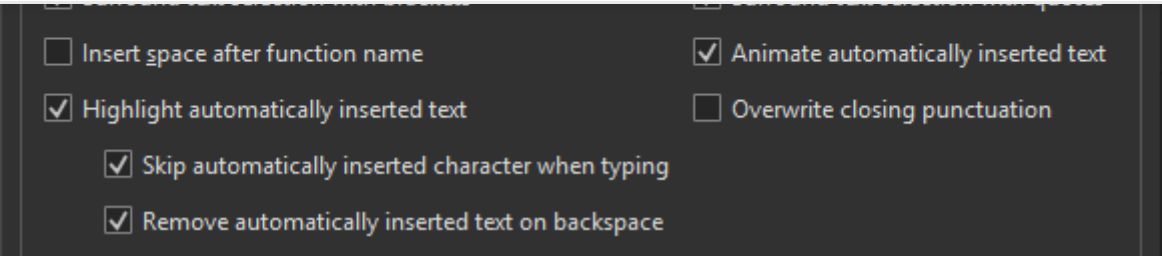
下图显示了完成 QML 代码的建议：

若要随时打开建议列表，请按 **Ctrl+空格键**。如果只有一个选项可用，Qt设计工作室会自动插入它。

指定完成设置

若要指定代码完成的设置，请选择"**编辑**>**首选项**">"**文本编辑器**">**完成**"。

默认情况下，代码完成不考虑大小写。若要应用完整字母或首字母区分大小写，请在"区分**大小写**"字段中选择"**完整**"或"**首字母**"。

默认情况下，始终调用代码完成，但您可以在"**激活完成**"字段中更改此行为，以**手动**或**触发时**调用它。

可以在"超时（毫秒）"字段中设置代码完成的超时（以毫秒**为单位**）。

在"**字符阈值**"字段中，指定在触发代码完成之前需要输入的字符数。

手动调用完成时，Qt 设计工作室将完成建议列表的通用前缀。这对于具有多个名称相似的成员的类特别有用。要禁用此功能，请取消选中"**自动完成通用前缀**"复选框。

选中"**自动拆分字符串**"复选框，通过在光标位置添加结束引号（按 Enter 键时）将字符串拆分为两行，并在下一行的开头（在字符串的其余部分之前）添加一个开始引号。此外，按 Shift+Enter **会在**光标位置插入一个转义字符，并将字符串的其余部分移动到下一行。
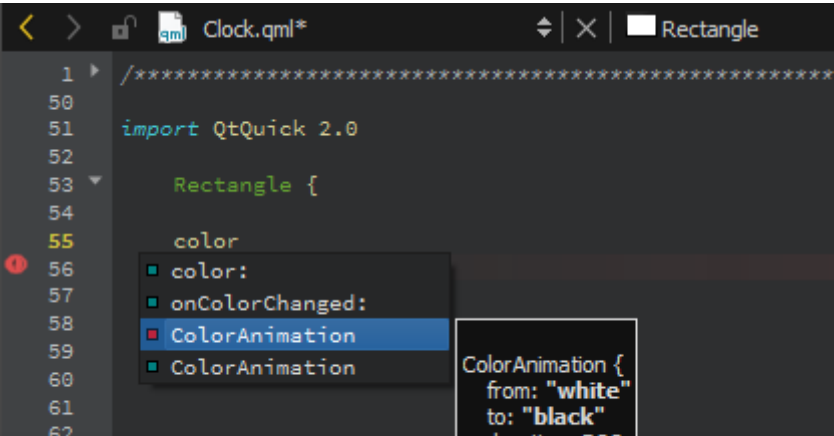
可用类型摘要

下表列出了代码完成的可用类型和用于每种类型的图标。

图标	描述:_____
■	QML 类型
■	一个量子点代码片段

完成代码片段

Code snippets can consist of multiple variables that you specify values for. Select an item in the list and press **Tab** or **Enter** to complete the code. Press **Tab** to move between the variables and specify values for them. When you specify a value for a variable, all instances of the variable within the snippet are renamed.
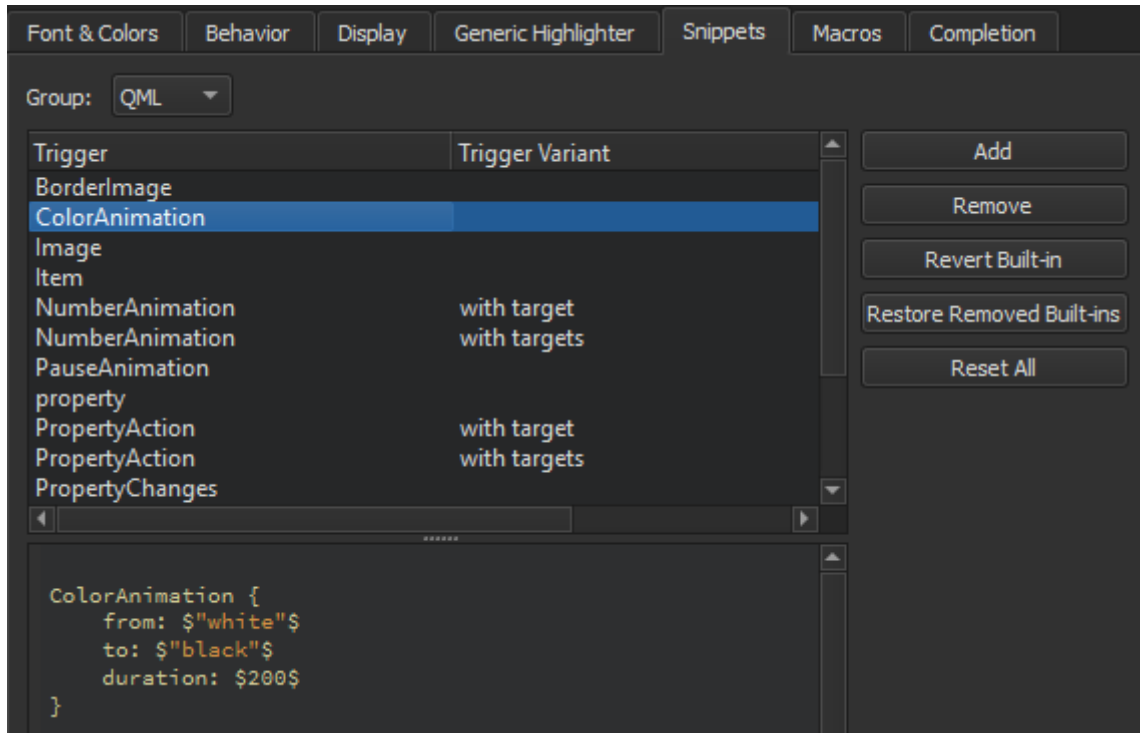
The following image shows a QML code snippet:

Editing Code Snippets

Code snippets specify code constructs. You can add, modify, and remove snippets in the snippet editor. To open the editor, select **Edit** > **Preferences** > **Text Editor** > **Snippets**.

The following image shows built-in QML code snippets:

Qt Design Studio provides you with built-in snippets in the following categories:

› Text snippets, which can contain any text string. For example, code comments

› QML code snippets, which specify QML code constructs

Adding and Editing Snippets

Select a snippet in the list to edit it in the snippet editor. To add a new snippet, select **Add**. Specify a trigger and, if the trigger is already in use, an optional variant, which appear in the list of suggestions when you write code. Also specify a text string or QML code in the snippet editor, depending on the snippet category. You can use predefined variables in snippets.

The snippet editor provides you with:

› Highlighting

› Indentation

› Parentheses matching

› Basic code completion

Specify the variables for the snippets in the following format:

```
$variable$
```

To discard the changes you made to a built-in snippet, select **Revert Built-in**.

Removing Snippets

Several similar built-in snippets might be provided for different use cases. To make the list of suggestions shorter when you write code, remove the built-in snippets that you do not need. If you need them later, you can restore them.

To remove snippets, select a snippet in the list, and then select **Remove**. To restore the removed snippets, select **Restore Removed Built-ins**.
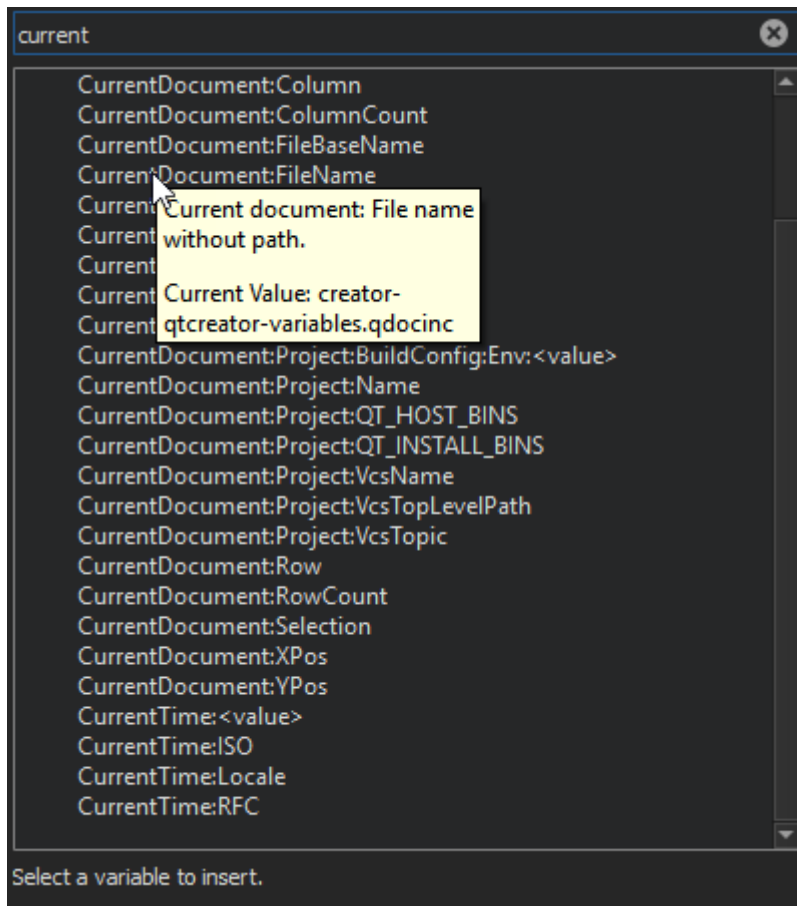
Resetting Snippets

To remove all added snippets and to restore all removed snippets, select **Reset All**.

> **Note:** If you now select **OK** or **Apply**, you permanently lose all your own snippets.

Using Qt Creator Variables

You can use Qt Design Studio variables in arguments, executable paths, and working directories. The variables take care of quoting their expansions, so you do not need to put them in quotes.

Select the (**Variables**) button in a field to select from a list of variables that are available in a particular context. For more information about each variable, move the cursor over it in the list.

within *variable* with *replacement*, use:

```
%{variable/pattern/replacement}
```

To replace all matches of *pattern* within *variable* with *replacement*, use:

```
%{variable//pattern/replacement}
```

The pattern can be a regular expression and the replacement can contain backreferences. For example, if is , then is expanded to .`%{variable}my123var%{variable/(..)(\d+)/\2\1}123myvar`

Instead of the forward slash, you can also use the pound sign () as the substitution character. This can be helpful if the value is supposed to be a file path, in which case forward slashes might get translated to backslashes on Windows hosts.#

To use the default value if the variable is not set, use:

```
%{variable:-default}
```

< Checking Code Syntax   Indenting Text or Code >

Company

About Us

Investors

Newsroom

Careers

Office Locations

Licensing

Terms & Conditions

Open Source

FAQ

Qt DOCUMENTATION

Support Services

Professional Services

Partners

Training

Support Center

Downloads

Qt Login

Contact Us

Customer Success

Community

Contribute to Qt

Forum

Wiki

Downloads

Marketplace

© 2022 The Qt Company

Feedback   Sign In