

# Contents

## Ferramentas XML no Visual Studio

### Editor de XML

- Visão geral

- Validação de documento XML

- Recursos do IntelliSense o editor XML

### Snippets XML

- Visão geral

- Usar snippets XML

- Gerar um snippet de XML de um esquema XML

- Criar snippets XML

### Editar folhas de estilos XSLT

### Cache de esquema

### Tarefas de editor XML

- Passo a passo: Usar recursos do editor XML

- Selecionar os esquemas XML que serão usados

- Criar um esquema XML de um documento XML

- Executar uma transformação XSLT do editor de XML

- Editar arquivos XML

- Explicações passo a passo sobre o XSLT

- XSLT IntelliSense

- Hierarquia XSLT

### Propriedades de arquivo XML

### Caixa de diálogo de esquemas XML

### Designer de Esquema XML

- Visão geral

### XML Schema Explorer

- XML Schema Explorer

- Pesquisar o conjunto de esquemas

- Classificar, filtrar e agrupar

Menus de contexto

Integração de literais do XML

Workspace

Visão geral das exibições

O modo de Início

O modo do modelo de conteúdo

Exibição de gráfico

Integração com editor XML

Tarefas do Designer de Esquema XML

Criar e editar um arquivo de esquema XSD

Criar um documento XML baseado em um esquema XSD

Adicionar nós de resultados da pesquisa de conjunto de esquema ao workspace

Alternar entre os modos de exibição e o editor de XML

Adicionar nós ao workspace do XML Schema Explorer

Adicionar nós ao workspace da exibição inicial

Obter uma visão geral de um conjunto de esquemas

Examinar o modelo de conteúdo de nós

Usar o designer de esquema XML com literais do XML

Imprimir diagramas

Arquivos de exemplo XSD

Esquema simples

Esquema de ordem de compra

Relações

Depuração de XSLT

Visão geral

Passo a passo: Depurar uma folha de estilos XSLT

Avaliar uma expressão XPath

Interface de usuário do depurador XSLT

Modelo padrão XSLT

Criador de perfil XSLT

Considerações de segurança para XML

# Visão geral das ferramentas XML no Visual Studio

14/05/2021 • 2 minutes to read

*Linguagem XML (XML)* é uma linguagem de marcação que fornece um formato para descrever os dados. O XML separa os dados e sua apresentação usando folhas de estilo associadas, como XSL (Extensible Stylesheet Language) e CSS (Cascading Style Sheets). O Visual Studio inclui ferramentas e recursos que facilitam trabalhar com os esquemas XML, XSLT e XML.

## Editor de XML

O [Editor de XML](#) é usado para editar documentos XML. Ele fornece verificação de sintaxe XML completa, validação de esquema enquanto você digita, codificação de cor e IntelliSense. Se um esquema ou uma definição de tipo de documento forem fornecidos, ele é usado pelo IntelliSense para listar os elementos e atributos permitidos.

Recursos adicionais incluem:

- Suporte a trecho XML, incluindo trechos de código gerados pelo esquema
- Documento de estrutura de tópicos para que os elementos possam ser expandidos e recolhidos
- A capacidade de executar transformações XSLT e exibir os resultados como texto, XML ou HTML
- A capacidade de gerar esquemas XSD (linguagem de definição de esquema XML) a partir do documento da instância XML
- Suporte para edição de folhas de estilo XSLT, incluindo suporte a IntelliSense
- XML Schema Explorer

## Designer de Esquema XML

O [Designer de esquema XML](#) é integrado ao Visual Studio e ao editor de XML para permitir que você trabalhe com esquemas XSD (linguagem de definição de esquema XML).

## Depuração de XSLT

O Visual Studio dá suporte à [depuração de folhas de estilo XSLT](#). Usando o depurador, você pode definir pontos de quebra em uma folha de estilos XSLT, entrar em uma folha de estilos XSLT a partir do código, e assim por diante.

### NOTE

O depurador XSLT só está disponível na edição Enterprise do Visual Studio.

## Confira também

- [System.Xml](#)
- [Transformações XSLT](#)
- [Processar dados XML usando o modelo de dados XPath](#)
- [XML Document Object Model \(DOM\)](#)

- [SOM \(Schema Object Model\) XML](#)

# Editor de XML

14/05/2021 • 2 minutes to read

O editor de XML no Visual Studio é baseado no editor de texto e inclui suporte adicional para as linguagens XML. Quando você abre um arquivo XML no Visual Studio, ele é aberto no editor de XML.

O editor de XML inclui os seguintes recursos:

- Verificação de sintaxe XML 1.0.
- Validação de esquema quando você digita.
- Suporte de snippets de XML, incluindo snippets gerados por esquema.
- Suporte para definição de tipo de documento (DTD).
- Suporte para o esquema da linguagem XSD.
- Criando um esquema XML de um documento de instância XML.
- Convertendo um DTD ou um esquema XDR em um esquema XML.
- Verificação de sintaxe XSLT.
- Estrutura de tópicos de documento de modo que os elementos possam ser expandidos e recolhidos.
- Integração com o [XML Schema Explorer](#). Isso fornece uma exibição hierárquica de esquemas XML.

O editor de XML é invocado para extensões de arquivo conhecidas, como `.xml`, `.xsd`, `.xsl` e `.config`. Ele também será invocado em qualquer extensão de arquivo desconhecida se o arquivo parecer conter XML.

## XSLT IntelliSense

O [XSLT IntelliSense](#) permite que você preencha automaticamente nomes de conjunto de atributos, modos de modelo e nomes e nomes de parâmetro para um modo especificado ou um modelo nomeado especificado.

## Criador de perfil XSLT

O [criador de perfil XSLT](#) cria relatórios de desempenho XSLT detalhados que ajudam a medir, avaliar e direcionar problemas relacionados ao desempenho no código XSLT. O profiler XSLT também inclui dicas úteis para XSL e otimizações de folha de estilos XSLT.

## Hierarquia XSLT

A [ferramenta de hierarquia XSLT](#) permite que você adicione pontos de interrupção em folhas de estilo incluídas e/ou regras de modelo internas.

## Confira também

- [Opções do editor de XML – formatação](#)
- [Opções do editor de XML – diversos](#)
- [Recursos do editor de código](#)
- [Referência de padrões de XML](#)
- [Ferramentas XML no Visual Studio](#)

# Validação de documento XML

14/05/2021 • 2 minutes to read

O editor de XML verifica a sintaxe XML 1,0 e também executa a validação de dados conforme você digita. O editor pode validar usando uma DTD (definição de tipo de documento) ou um esquema. Os sublinhados ondulados vermelhos realçam todos os erros de XML 1.0 bem-formado. Os sublinhados ondulados azuis mostram os erros semânticos com base na validação de DTD ou de esquema. Cada erro tem uma entrada associada na lista de erros. Você também pode exibir a mensagem de erro pausando o mouse sobre o sublinhado ondulado.

Os esquemas usados na validação são encontrados correspondendo o `targetNamespace` de um esquema compilado com a declaração `xmlns` do elemento. Os esquemas compilados são carregados de um dos seguintes locais, listados por ordem de prioridade:

- Do nome de arquivo especificado no campo **esquemas** da janela **Propriedades** do documento.
- Um esquema ou DTD embutido.
- Uma DTD externa ou os atributos `xsd:schemaLocation` e `xsd:noNamespaceSchemaLocation`
- Um URI de um namespace do esquema XDR "x-schema".

Os esquemas também podem ser encontrados nos seguintes locais adicionais quando o esquema tiver um namespace de destino não vazio:

- Outra janela do editor que contenha o esquema.
- Um esquema na solução atual.
- Um esquema do diretório de cache de esquema.

## Arquivos XSLT

Ao editar um arquivo XSLT, o arquivo *XSLT.xsd* localizado no cache de esquema é usado para validação. Os erros de validação são mostrados como sublinhados ondulados azuis. Os erros do compilador XSLT são mostrados como sublinhados ondulados vermelhos.

## Arquivos de esquema XML (XSD)

Ao editar um arquivo de esquema XML, o arquivo *xsdSchema.xsd* localizado no cache de esquema é usado para validação. Os erros de validação são mostrados como sublinhados ondulados azuis. Todos os erros de compilação também são mostrados com traços ondulados vermelhos.

## Confira também

- [Editor de XML](#)

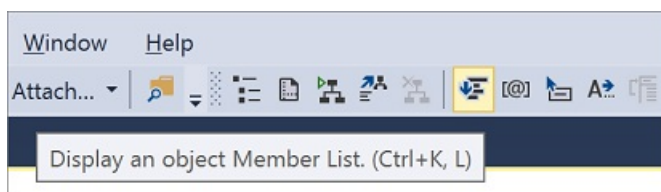
# Recursos do IntelliSense o editor XML

14/05/2021 • 3 minutes to read

O editor de XML fornece recursos do IntelliSense completos comparáveis a outros editores de linguagem fornecidos no Visual Studio. Esta seção explica como você pode usar o IntelliSense com a linguagem de definição de esquema XML (XSD) e documentos XSLT.

## IntelliSense em um documento XSD

Depois que um esquema é associado ao documento, você obtém uma lista suspensa de elementos esperados sempre que digitar "<" ou clicar no botão **exibir uma lista de membros de objetos** na barra de ferramentas do editor de XML.



Para obter informações sobre como associar esquemas a documentos XML, consulte validação de [documento XML](#).

Quando você digita o ESPAÇO de dentro de uma marca inicial, você também obtém uma lista suspensa que mostra todos os atributos que podem ser adicionados ao elemento atual.

Quando você digita "=" para um valor de atributo, ou as aspas de abertura para o valor, você também obtém a lista de valores possíveis para esse atributo. Os valores são fornecidos apenas se o esquema fornece valores enumerados através de facetas de `xsd:enumeration`, ou se o atributo é um tipo de `Boolean`. Uma lista do IntelliSense de códigos de idioma conhecido também é fornecida para `xml:lang` ou qualquer `simpleType` que deriva de `xsd:language`. Uma lista do IntelliSense de valores conhecidos de `targetNamespace` é fornecida para declarações de namespace.

Uma lista do IntelliSense de valores possíveis é fornecida também quando você digita ">" para fechar uma tag de início se o elemento é `simpleType`. O comportamento de elementos é semelhante ao comportamento dos atributos descritos no parágrafo anterior.

Dicas de ferramenta também aparece nessas IntelliSense listas com base em `xsd:annotation` e informações de `xsd:documentation` encontrado no esquema associado.

## IntelliSense em um documento XSLT

Após adicionar um modelo nomeado ou um atributo para o documento de fonte, você pode usar o IntelliSense para inserir o seguinte:

- Nomes definidos de atributo.
- Modos do modelo.
- Nomes de modelo.
- Nomes de parâmetro para um modo determinado.
- Nomes de parâmetro para um modelo chamado determinado.

Para obter mais informações, consulte [Walkthrough: using XSLT IntelliSense](#) tópico.

## Preenchimento automático

O editor XML também facilita editando XML preenchendo na sintaxe XML necessário para você. Por exemplo, se você digitar a seguinte marcação inicial:

```
<book>
```

O editor XML preenche a marca de fim e posicionar o cursor após a marca inicial. Veja a seguir um exemplo disso (o "|" anota a posição do cursor):

```
<book> | </book>
```

Porque valores de atributo devem sempre ter aspas, o editor XML preenche as aspas para você. Por exemplo, se você digitar o seguinte:

```
<book title=
```

O editor XML adiciona as aspas e posicionar o cursor entre aspas:

```
<book title=" | "
```

Da mesma forma, o editor XML também insere a seguinte sintaxe XML automaticamente para você:

- Termina uma instrução de processamento: `?>`
- Finalizar um bloco CDATA: `]]>`
- Termine um comentário: `-->`
- Termina uma declaração de DTD: `>`

O editor de XML também tem a capacidade de inserir uma declaração de namespace se você selecionar um atributo ou elemento qualificado de namespace de uma lista do IntelliSense e o namespace para esse elemento ou atributo ainda não estiver no escopo.

Por exemplo, se você selecionar o elemento de `e:Book` de lista do IntelliSense onde o prefixo é associado ao namespace de `http://books` que não foi declarada no documento, o editor XML insere a declaração de namespace necessário para você. O seguinte é o texto resultante XML:

```
<e:Book xmlns:e="http://books"
```

## Correspondência de chaves

O editor XML fornece a chave realçando para fornecer feedback imediato em elementos que apenas se você tiver fechado. Você também pode usar o atalho de teclado (**Ctrl** + **J**) para saltar de uma chave para a chave correspondente.

O editor XML faz isso para os seguintes itens:

- Correspondência de marcas inicial e de fim.
- Qualquer par de colchetes angulares "<" or ">".
- Início e fim de comentários.
- Início e fim de instruções de processamento.
- Início e fim de blocos CDATA.



- Início e fim de declarações DTD.
- Aspas de abertura e fechamento em atributos.

## Modificar as opções do IntelliSense

Os recursos do IntelliSense e de preenchimento automático são ativados por padrão. No entanto, você pode alterar isso modificando as configurações de opções de **ferramentas** > .

A seção **inserção automática** da página **diversos** controla o seguinte comportamento:

NOME	DESCRIÇÃO
Fechar marcas	Insere fechar marcas para novos elementos.
Citações de atributo	O valor do atributo das inserções que quando você digite um novo nome de atributo.
Outra marcação	Comentários, CDATA termina, DOCTYPE, instruções de processamento, e outras declarações de marcação.

### Para alterar o comportamento de preenchimento automático

1. Selecione **Opções** no menu **Ferramentas**.
2. Expanda **Editor de texto**, expanda **XML** e selecione **diversos**.
3. Faça as alterações na seção **inserção automática** e clique em **OK**.

## Confira também

- [Editor de XML](#)
- [Usando o IntelliSense](#)
- [Passo a passo: usando XSLT IntelliSense](#)

# Snippets XML

14/05/2021 • 2 minutes to read

O editor de XML oferece um recurso, chamado *trechos XML*, que permite criar arquivos XML mais rapidamente. Você pode reutilizar XML inserindo snippets nos seus arquivos. Você também pode gerar dados XML com base em um esquema XSD (linguagem de definição de esquema XML).

## Trechos de código XML reutilizáveis

O editor de XML inclui muitos trechos de código que abrangem algumas tarefas comuns. Isso permite que você crie arquivos XML mais facilmente. Por exemplo, se você estivesse criando um esquema XML, usar os trechos de código "elemento de sequência de tipo complexo" e "elemento de tipo simples" insere o seguinte texto XML em seu arquivo. Você alteraria o valor de `name` para atender às suas necessidades.

```
<xs:element name="name">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="name">
        <xs:simpleType>
          <xs:restriction base="xs:string"></xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Você pode inserir snippets de duas maneiras. O comando **Insert Snippet** insere o trecho XML na posição do cursor. O comando **surround with** encapsula o trecho XML em torno do texto selecionado. Ambos os comandos estão disponíveis no submenu **IntelliSense**, no menu **Editar**, ou no menu de atalho dentro do editor.

Para obter mais informações, consulte [como: usar trechos XML](#).

## Trechos XML gerados pelo esquema

O editor de XML também tem a capacidade de gerar um trecho XML a partir de um esquema XML. Esse recurso permite que você popule um elemento com elementos XML gerados de informações de esquema para esse elemento. Para obter mais informações, consulte [como gerar um trecho XML a partir de um esquema XML](#).

## Criar novos trechos XML

Além dos trechos de código incluídos no Visual Studio por padrão, você também pode criar e usar seus próprios trechos de código XML. Para obter mais informações, consulte [como: criar trechos de código XML](#).

## Confira também

- [Snippets de código no Visual Studio](#)
- [Editor de XML](#)

# Como: usar trechos XML

14/05/2021 • 4 minutes to read

Você pode invocar trechos XML usando os dois comandos a seguir no menu de atalho do editor de XML. O comando **Insert Snippet** insere o trecho XML na posição do cursor. O comando **surround with** encapsula o trecho XML em torno do texto selecionado. Cada snippets XML designou tipos de snippets. Os tipos de trecho determinam se o trecho de código está disponível com o comando **Insert Snippet**, o comando **surround with** ou ambos.

Depois que o snippet XML foi adicionado ao editor, todos os campos editáveis no snippet estão realçados em amarelo, e o cursor está localizado no primeiro campo editável.

## Inserir Snippet

Os procedimentos a seguir descrevem como acessar o comando **Insert Snippet**.

### NOTE

O comando **Inserir trecho de código** também está disponível por meio do atalho de teclado (**Ctrl + K** e **Ctrl + X**).

### Para inserir snippets do menu de atalho

1. Posicionar o cursor onde você deseja inserir o snippet XML.
2. Clique com o botão direito do mouse e selecione **Inserir trecho**.

Uma lista de snippets disponíveis XML é exibida.

3. Selecione um trecho da lista usando o mouse ou digitando o nome do trecho de código e pressionando **Tab** ou **Enter**.

### Para inserir snippets usando o menu do IntelliSense

1. Posicionar o cursor onde você deseja inserir o snippet XML.
2. No menu **Editar**, aponte para **IntelliSense** e, em seguida, selecione **Inserir trecho de código**.

Uma lista de snippets disponíveis XML é exibida.

3. Selecione um trecho da lista usando o mouse ou digitando o nome do trecho de código e pressionando **Tab** ou **Enter**.

### Para inserir trechos por meio da lista de palavras do IntelliSense Complete

1. Posicionar o cursor onde você deseja inserir o snippet XML.
2. Comece a digitar o snippet XML que você deseja adicionar ao seu arquivo. Se o preenchimento automático é ativada, a lista de palavras completo do IntelliSense é exibida. Se não aparecer, pressione **Ctrl + espaço** para ativá-lo.
3. Selecione o snippet XML da lista de palavras completo.
4. Pressione **Tab** e **Tab** para invocar o trecho XML.

#### NOTE

Pode haver casos quando o snippet XML não é chamado. Por exemplo, se você tentar inserir um elemento de `xs:complexType` dentro de um nó de `xs:element`, o editor não gerencia um snippet XML. Quando um elemento de `xs:complexType` é usado dentro de um nó de `xs:element`, não houver nenhum atributo ou subelements necessário, o editor não tem nenhum dados para inserir.

#### Para inserir snippets usando o nome do atalho

1. Posicionar o cursor onde você deseja inserir o snippet XML.
2. Tipo `<` no painel do editor.
3. Pressione **ESC** para fechar a lista de palavras concluídas do IntelliSense.
4. Digite o nome do atalho do trecho e pressione **Tab** para invocar o trecho XML.

## Envolver com

Os procedimentos a seguir descrevem como acessar o comando **surround with**.

#### NOTE

O comando **surround with** também está disponível por meio do atalho de teclado (**Ctrl + K**, depois **Ctrl + S**).

#### Para usar surround with no menu de contexto

1. Selecione o texto a ser circundado no editor de XML.
2. Clique com o botão direito do mouse e selecione **Circundar com**.  
  
Uma lista de bordadura disponíveis com snippets XML é exibida.
3. Selecione um trecho da lista usando o mouse ou digitando o nome do trecho de código e pressionando **Tab** ou **Enter**.

#### Para usar o surround com no menu do IntelliSense

1. Selecione o texto a ser circundado no editor de XML.
2. No menu **Editar**, aponte para **IntelliSense** e, em seguida, selecione **surround com**.  
  
Uma lista de bordadura disponíveis com snippets XML é exibida.
3. Selecione um trecho da lista usando o mouse ou digitando o nome do trecho de código e pressionando **Tab** ou **Enter**.

## Usar snippets XML

Uma vez que você escolher um snippet XML, o texto de snippet de código é inserido automaticamente a posição do cursor. Todos os campos editáveis no snippet são realçadas, e o primeiro campo editável é automaticamente selecionado. O campo selecionado é convertido.

Quando um campo é selecionado, você pode digitar um novo valor para o campo. Pressionar **Tab** percorre os campos editáveis do trecho de código; pressionar **Shift + Tab** percorre os ciclos na ordem inversa. Clicando em um campo colocar o cursor no campo, e clique duas vezes em um campo selecioná-lo. Quando um campo é realçado, uma dica de ferramenta pode ser exibido, oferecendo uma descrição do campo.

Somente a primeira instância de um campo dado é editável. Quando esse campo é realçado, as outras instâncias do campo são descritas. Quando você altera o valor de um campo editável, o campo ele alterado everywhere é

usado no snippet.

Pressionar **Enter** ou **ESC** cancela a edição de campo e retorna o editor para normal.

As cores padrão para campos de trecho de código editável podem ser alteradas modificando a configuração do **campo de trecho de código** no painel **fontes e cores** da caixa de diálogo **Opções** . Para obter mais informações, consulte [como alterar fontes e cores no editor](#).

## Confira também

- [Snippets XML](#)
- [Como gerar um trecho XML a partir de um esquema XML](#)
- [Como: criar trechos XML](#)

# Como gerar um trecho XML a partir de um esquema XML

14/05/2021 • 3 minutes to read

O editor de XML tem a capacidade de gerar trechos de código XML a partir de um esquema XSD (linguagem de definição de esquema XML). Por exemplo, como você está criando um arquivo XML, enquanto posicionado ao lado do nome do elemento, você pode pressionar **Tab** para preencher o elemento com os dados XML gerados com base nas informações de esquema para esse elemento.

Este recurso está disponível somente nos elementos. As seguintes regras também se aplicam:

- O elemento deve ter um tipo associado de esquema; isto é, o elemento deve ser válido de acordo com qualquer esquema associado. O tipo de esquema não pode ser abstract e o tipo deve conter os atributos necessários e/ou os elementos filho necessários.
- O elemento atual no editor deve ser deixado sem atributos. Por exemplo, todos os seguintes são válidos

- `<Account`
- `<Account>`
- `<Account></Account>`

- O cursor deve ser localizado imediatamente à direita do nome do elemento.

O snippet gerado contém todos os atributos e elementos necessários. Se `minOccurs` é maior de um, o número mínimo necessário de instâncias desse elemento é incluído no snippet, até um máximo de 100 instâncias. Todos os valores fixos encontrados no esquema levam a valores fixos no snippet. `xsd:any` os `xsd:anyAttribute` elementos e são ignorados e não resultam em construções de trecho adicionais.

Os valores padrão são gerados e observados como valores editáveis. Se o esquema especifica um valor padrão, esse valor padrão é usado. Entretanto, se o valor padrão de esquema é uma cadeia de caracteres vazia, o editor gerencia os valores padrão da seguinte maneira:

- Se o tipo de esquema contém quaisquer facetas de enumeração, direta ou indiretamente por meio de alguns dos membros de um tipo de união, o primeiro valor enumerado encontrado no modelo de objeto de esquema é usado como o padrão.
- Se o tipo de esquema é um tipo atômico, o editor obtém o tipo atômico e insere o nome atômico de tipo. Para um tipo derivado simples usa o tipo simples base. Para um tipo de lista o tipo atômico é `itemType`. Para uma união atômico, o tipo é o tipo atômico de primeiro `memberType`.

## Exemplo

As etapas nesta seção mostram como usar o recurso de trecho XML gerado pelo esquema do editor de XML.

### NOTE

Antes de iniciar estes procedimentos, salve o arquivo de esquema para o seu computador local.

### Para criar um novo arquivo XML e associá-lo a um esquema XML

1. No menu **arquivo**, aponte para **novo** e clique em **arquivo**.

2. Selecione **arquivo XML** no painel **modelos** e clique em **abrir**.

Um novo arquivo é aberto no editor. O arquivo contém uma declaração XML padrão,

```
<?xml version="1.0" encoding="utf-8"> .
```

3. Na janela Propriedades do documento, clique no botão procurar (...) no campo **esquemas** .

A caixa de diálogo **esquemas XSD** é exibida.

4. Clique em **Adicionar**.

A caixa de diálogo **abrir esquema XSD** é exibida.

5. Selecione o arquivo de esquema e clique em **abrir**.

6. Clique em **OK**.

O esquema XML agora está associado ao documento XML.

### Para gerar um snippet XML

1. Tipo **<** no painel do editor.

2. A lista de membros exibe os itens possíveis:

**!--** adicionar um comentário.

**! DOCTYPE** para adicionar um tipo de documento.

**?** para adicionar uma instrução de processamento.

**Contate** para adicionar o elemento raiz.

3. Selecione **contato** na lista de membros e pressione **Enter**.

O editor adiciona a tag de início **<Contact** e posicionar o cursor após o nome do elemento.

4. Pressione **Tab** para gerar dados XML para o **Contact** elemento com base em suas informações de esquema.

## Entrada

O seguinte arquivo de esquema é usado por passo a passo.

```

<?xml version="1.0" encoding="utf-8" ?>
<xs:schema elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="phoneType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="Voice"/>
      <xs:enumeration value="Fax"/>
      <xs:enumeration value="Pager"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="Contact">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Name">
          <xs:simpleType>
            <xs:restriction base="xs:string"></xs:restriction>
          </xs:simpleType>
        </xs:element>
        <xs:element name="Title"
          type="xs:string" />
        <xs:element name="Phone"
          minOccurs="1"
          maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Number"
                minOccurs="1">
                <xs:simpleType>
                  <xs:restriction base="xs:string"></xs:restriction>
                </xs:simpleType>
              </xs:element>
              <xs:element name="Type"
                default="Voice"
                minOccurs="1"
                type="phoneType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

## Saída

A seguir estão os dados XML que são gerados com base nas informações de esquema associada com o elemento de `Contact` . Itens marcados como **bold** designar campos editáveis no trecho XML.

```

<Contact>
  <Name>name</Name>
  <Title>title</Title>
  <Phone>
    <Number>number</Number>
    <Type>Voice</Type>
  </Phone>
</Contact>

```

## Confira também

- [Snippets XML](#)
- [Como: usar trechos XML](#)



# Como: criar trechos XML

14/05/2021 • 3 minutes to read

O editor de XML pode ser usado para criar novos trechos de código XML. O editor inclui um snippet XML, chamado "Snippets", que é um snippet de texto constante para criar novos snippets XML.

## Para criar um novo snippet XML

Para criar um novo trecho de código XML, crie um novo arquivo XML e use o recurso **Inserir trecho**.

1. No menu **arquivo**, clique em **novo** e em **arquivo**.
2. Clique em **arquivo XML** e em **abrir**.
3. Clique com o botão direito do mouse no painel do editor e selecione **Inserir trecho de código**.
4. Selecione **trecho** na lista e pressione **Enter**.
5. Faça as alterações para o novo snippet.
6. No menu **arquivo**, selecione **salvar XMLFile.xml**.

A caixa de diálogo **salvar arquivo como** é exibida.

7. Insira o nome para o novo trecho e selecione **arquivos de trecho** na janela suspensa **salvar como tipo**.
8. Use a lista suspensa **salvar na** para alterar o local do arquivo para a pasta *Meus Documentos\Visual Studio 2005 \ código Snippets\XML\My XML Snippets* e, em seguida, pressione **salvar**.

## Descrição do trecho

Esta seção descreve alguns dos elementos-chave no snippet de texto constante. Para obter mais informações sobre elementos de esquema usados pelos trechos XML, consulte [referência de esquema de trechos de código](#).

### Elemento SnippetType

Suporte do editor dois tipos de snippet:

```
<SnippetTypes>
  <SnippetType>SurroundsWith</SnippetType>
  <SnippetType>Expansion</SnippetType>
</SnippetTypes>
```

- O **Expansion** tipo determina se o trecho de código aparece quando você invoca o comando **Insert Snippet**. O **SurroundsWith** tipo determina se o trecho de código é exibido quando você invoca o comando **Circundar com**.

### Elemento de código

O elemento de **Code** define o texto XML que será inserido quando o snippet é chamado.

#### NOTE

O texto de snippet XML deve ser incluído em uma seção de `<![CDATA[...]]>`.

O seguinte é o elemento de `Code` que é criado pelo snippet de texto constante.

```
<Code Language="XML">
  <![CDATA[<test>
    <name>$name$</name>
    $selected$ $end$</test>]]>
</Code>
```

O elemento de `Code` inclui três variáveis.

- `$name$` variável é definido pelo usuário. Cria um elemento de `name`, que tem um valor editável que usa padrão "para nomear". As variáveis definidas pelo usuário são definidos usando o elemento de `Literal`.
- `$selected$` é uma variável predefinido. Ele representa o texto que foi selecionado no editor de XML antes de invocar o trecho de código. O posicionamento dessa variável determina onde o texto selecionado aparece no snippet de código que circunda a seleção.
- `$end$` é uma variável predefinido. Quando o usuário pressiona **Enter** para concluir a edição dos campos de trecho de código, essa variável determina para onde o cursor (^) é movido.

O elemento acima de `Code` insira o seguinte texto XML:

```
<test>
  <name>name</name>
</test>
```

O valor do elemento de nome é marcado como uma região editável.

### Elemento Literal

O elemento de `Literal` é usado para identificar o texto de substituição que pode ser personalizada depois que é inserido no arquivo. Por exemplo, cadeias de caracteres literais, os valores numéricos, e alguns nomes de variável podem ser declarados como literais. Você pode definir qualquer número do snippet em literais XML e você pode referir-lhes várias vezes dentro de snippet. O código a seguir é um exemplo de um elemento de `Literal` que define um variável de `$name$` cujo valor padrão é "nome".

```
<Literal>
  <ID>name</ID>
  <Default>name</Default>
</Literal>
```

Literais também podem se referir funções. O editor de XML inclui uma função chamada **LookupPrefix**. A função **LookupPrefix** pesquisa o URI de namespace fornecido do local no documento XML para o qual esse trecho é invocado e retorna o prefixo de namespace definido para esse namespace, se houver, e inclui os dois-pontos (:) nesse nome. Veja a seguir um exemplo de um `Literal` elemento que usa a função **LookupPrefix**.

```
<Literal Editable="false">
  <ID>prefix</ID>
  <Function>LookupPrefix("namespaceURI")</Function>
</Literal>
```

A variável de `$prefix$` pode então ser usado em qualquer lugar no seu snippet XML.

## Consulte também

- [Snippets XML](#)

- [Como: usar trechos XML](#)
- [Como gerar um trecho XML a partir de um esquema XML](#)

# Editar folhas de estilo XSLT

14/05/2021 • 2 minutes to read

O editor de XML também pode ser usado para editar folhas de estilo XSLT. Você pode tirar proveito dos recursos do editor padrão como o IntelliSense, estruturação, snippets XML, e assim por diante. Além disso, há também os novos recursos que tornam ficar em XSLT.

## Recursos de fonte

A tabela a seguir descreve os recursos específicos para trabalhar com folhas de estilos XSLT.

### Cor da sintaxe

As palavras-chave XSLT, como `template` e `match`, são exibidas na cor da palavra-chave XSLT especificada pelas configurações de **fontes e cores**.

### Traços ondulados

O editor de XML usa o arquivo *XSLT.xsd* instalado para validar as folhas de estilo XSLT. Os erros de validação são mostrados como sublinhados ondulados azuis. O editor de XML também compila a folha de estilos em segundo plano e relata erros ou avisos do compilador com sublinhados ondulados apropriados.

### Suporte para blocos de script

O código nos blocos de script é suportado pelo depurador XSLT para que você pode definir pontos de interrupção e percorrer o código de bloco de script.

### Saída XSLT de exibição

Você pode executar uma transformação XSL e exibir a saída do editor de XML. Para obter mais informações, consulte [como executar uma transformação XSLT no editor de XML](#).

### Depuração XSLT

Você pode iniciar o depurador XSLT de um arquivo XSLT no editor de XML. O depurador oferece suporte pontos de interrupção no arquivo XSLT, estado de configuração de execução XSLT de exibição, e assim por diante. Passa sobre uma variável XSLT traz anterior um ToolTip com o valor da variável. O depurador pode ser usado para depurar uma folha de estilos, ou depurar uma transformação XSL compilado chamada de outro aplicativo. Para obter mais informações, consulte [DEBUGGING XSLT](#).

## Consulte também

- [Editor de XML](#)

# Cache de esquema

14/05/2021 • 4 minutes to read

O editor de XML fornece um cache de esquema localizado no diretório `%VSINSTALLDIR%\xml\Schemas`. O cache de esquema é global para todos os usuários em seu computador e inclui esquemas XML padrão que são usados para a validação de documento XML e IntelliSense.

O editor de XML também pode encontrar esquemas localizados na solução, esquemas especificados no campo **esquemas** da janela **Propriedades** do documento e esquemas identificados pelos atributos `xsi:schemaLocation` e `xsi:noNamespaceSchemaLocation`.

A tabela a seguir descreve os esquemas que são instalados com o editor de XML.

NOME DE ARQUIVO	DESCRIÇÃO
<i>catalog.xsd</i>	Esquema para arquivos de catálogo do editor XML. Para obter informações sobre catálogos do esquema, consulte abaixo.
<i>DotNetConfig.xsd</i>	Esquema para arquivos de Web.Config, <code>http://schemas.microsoft.com/.NETConfiguration/v2.0</code> .
<i>msbuild.xsd</i>	Esquema para o MSBuild Make files, <code>http://schemas.microsoft.com/developer/msbuild/2003</code> .
<i>msdata.xsd</i>	Para anotações esquema XSD adicionados pela classe de <a href="#">DataSet</a> , "urna: esquema-Microsoft-COM: XML-msdata".
<i>msxsl.xsd</i>	Esquema para extensões do bloco de script do Microsoft XSLT, urna: esquema-Microsoft-COM: XSLT.
<i>SnippetFormat.xsd</i>	Esquema para os arquivos XML de snippet de código. Para obter exemplos, consulte <code>%VSINSTALLDIR%\VC#\Expansions</code> .
<i>Soap1.1.xsd</i>	Esquema para SOAP (Simple Object Access Protocol) 1,1, <code>http://schemas.xmlsoap.org/soap/envelope/</code> .
<i>Soap1.2.xsd</i>	Esquema para o protocolo de acesso simples 1,2 do objeto.
<i>SiteMapSchema.xsd</i>	Esquema para o arquivo XML Sitemap ASP.NET, <code>http://schemas.microsoft.com/AspNet/SiteMap-File-1.0</code> .
<i>wsdl.xsd</i>	Esquema para linguagem de descrição de serviço Web, <code>http://schemas.xmlsoap.org/wsdl/</code> .
<i>xenc.xsd</i>	Esquema para criptografia XML, <code>http://www.w3.org/2000/09/xmldsig#</code> .

NOME DE ARQUIVO	DESCRIÇÃO
<i>xhtml.xsd</i>	Esquema para XHTML <a href="http://www.w3.org/1999/xhtml">http://www.w3.org/1999/xhtml</a> .
<i>xlink.xsd</i>	Esquema para XLink 1.0, <a href="http://www.w3.org/1999/xlink">http://www.w3.org/1999/xlink</a> .
<i>xml.xsd</i>	Esquema que descreve os atributos XML: Space e XML: lang, <a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a> .
<i>xmlsig.xsd</i>	Esquema para assinaturas digitais XML, <a href="http://www.w3.org/2000/09/xmldsig#">http://www.w3.org/2000/09/xmldsig#</a> .
<i>xsdschema.xsd</i>	Esquema que descreve o XSD em si, <a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a> .
<i>xslt.xsd</i>	Esquema para transformações XML, <a href="http://www.w3.org/1999/XSL/Transform">http://www.w3.org/1999/XSL/Transform</a> .

## Atualizar esquemas no cache

O editor carrega o diretório de cache do esquema quando o pacote de editor XML é carregado e observações para todas as alterações ao executar. Se um esquema foi adicionado, é carregado automaticamente em um índice de memória conhecidos de esquemas. Se um esquema foi removido, ele é removido automaticamente de índice de memória. Se um esquema foi atualizado, invalida automaticamente o cache de memória deste esquema.

### NOTE

Porque o diretório de cache de esquema é global para seu computador, você só deve adicionar os esquemas aqui e padrões que são úteis para todos os projetos do Visual Studio que podem ser criados no seu computador.

O editor XML também suporta qualquer número de arquivos de catálogo de esquema no diretório de cache de esquema. Catálogo de esquema podem apontar para outros locais para esquemas que você deseja sempre o editor para saber. O arquivo *Catalog.xsd* define o formato para o arquivo de catálogo e é incluído no diretório de cache do esquema. O arquivo de *catalog.xml* é o catálogo padrão e contém links para outros esquemas no % *VSINSTALLDIR*%. Veja a seguir uma amostragem do arquivo de *catalog.xml*:

```
<SchemaCatalog xmlns="http://schemas.microsoft.com/xsd/catalog">
  <Schema href="%VSInstallDir%/help/schemas/Favorites.xsd" targetNamespace="urn:Favorites-Schema"/>
  <Schema href="%VSInstallDir%/help/schemas/Links.xsd" targetNamespace="urn:Links-Schema"/>
  <Schema href="%VSInstallDir%/help/schemas/MyHelp.xsd" targetNamespace="urn:VSHelp-Schema"/>
</SchemaCatalog>
```

O atributo de `href` pode ser qualquer URL do caminho de arquivo ou de HTTP que aponta para o esquema. O caminho de arquivo pode ser relativo ao documento de catálogo. As variáveis a seguir, delimitadas por%%, são reconhecidas pelo editor e expandidas no caminho:

- VSInstallDir
- Sistema
- ProgramFiles
- Programas

- CommonProgramFiles
- ApplicationData
- CommonApplicationData
- LCID

O documento de catálogo pode incluir um elemento de `Catalog`, que aponta para outros catálogos. Você pode usar o elemento de `Catalog` para apontar para um catálogo central compartilhado por sua equipe ou empresa, ou um catálogo online compartilhado com seus sócios de negócios. O atributo de `href` é a URL do caminho de arquivo ou de HTTP para os outros catálogos. A seguir está um exemplo de elemento de `Catalog`:

```
<Catalog href="file://c:/xbl/xblCatalog.xml"/>
```

O catálogo também pode controlar como os esquemas são associados com os documentos XML usando o elemento especial de `Association`. Esse elemento associa esquemas que não têm nenhum namespace de destino com uma extensão de arquivo específica, o que pode ser útil porque o editor de XML não faz nenhuma associação automática de esquemas que não têm um `targetNamespace` atributo. No exemplo a seguir o elemento de `Association` associa o esquema de dotNetConfig com todos os arquivos que possuem a extensão do arquivo de configuração: "

```
<Association extension="config" schema="%VSInstallDir%/xml/schemas/dotNetConfig.xsd"/>
```

## Esquemas localizados

Em muitos casos, o arquivo de *catalog.xml* não contém entradas para esquemas localizados. Você pode adicionar entradas adicionais ao arquivo de *catalog.xml* que aponta para o diretório de esquema localizado.

No exemplo a seguir um novo elemento de `Schema` foi criado que usa a variável de %LCID% para apontar para o esquema encontrado.

```
<Schema href="%InstallRoot%/Common7/IDE/Policy/Schemas/%LCID%/TDLSchema.xsd"
  targetNamespace="http://www.microsoft.com/schema/EnterpriseTemplates/TDLSchema"/>
```

## Alterar o local do cache de esquema

Você pode personalizar o local do cache de esquema usando a página opções **diversas**. Se você tiver um diretório de esquemas favoritos, o editor pode ser configurado para usar esses esquemas.

### NOTE

Essa alteração afeta somente o usuário atual Visual Studio.

### Para alterar o local de cache do esquema

1. No menu **Ferramentas**, selecione **Opções**.
2. Expanda **Editor de texto**, expanda **XML** e clique em **diversos**.
3. Clique no botão **procurar** no campo **esquemas**.
4. Selecione a pasta para o cache de esquema e clique em **OK**.

### Para adicionar um diretório diferente de esquemas comuns

1. Edite o arquivo de *catalog.xml* no diretório de cache do esquema do editor de XML.
2. Adicionar um novo elemento de `<Catalog href="..." />` que aponta para o diretório de esquemas adicionais.
3. Salve suas alterações.

O catálogo é recarregado automaticamente.

## Confira também

- [Editor de XML](#)



# Passo a passo: Usar recursos do editor XML

14/05/2021 • 3 minutes to read

As etapas nessa explicação passo a passo mostram como criar um novo documento XML. O passo a passo também usa alguns dos recursos do editor de XML que o tornam valioso para a criação de XML.

## NOTE

Antes de iniciar o passo a passo, salve o arquivo *HireDate.xsd* (incluído abaixo neste tópico) em seu computador local.

## Para criar um novo arquivo XML e associá-lo a um esquema XML

1. No menu **arquivo**, aponte para **novo** e clique em **arquivo**.
2. Selecione **arquivo XML** no painel **modelos** e clique em **abrir**.

Um novo arquivo é aberto no editor. O arquivo contém uma declaração XML padrão,

```
<?xml version="1.0" encoding="utf-8"> .
```

3. Na janela Propriedades do documento, clique no botão procurar (...) no campo **esquemas**.

A caixa de diálogo **esquemas XSD** é exibida.


4. Clique em **Adicionar**.

A caixa de diálogo **abrir esquema XSD** é exibida.

5. Selecione o arquivo *HireDate.xsd* e clique em **abrir**.
6. Clique em **OK**.

O esquema XML agora está associado com o documento XML. O esquema XML é usado para validar o documento. Também é usado pelo IntelliSense para preencher a lista de membros de elementos válidos.

## Para adicionar dados

1. Tipo  no painel do editor.

A lista de membros exibe os itens possíveis:

- **!--** adicionar um comentário.
- **! DOCTYPE** para adicionar um tipo de documento.
- **?** para adicionar uma instrução de processamento.
- **funcionário** para adicionar o elemento raiz.

2. Selecione **< !--** para adicionar um nó de comentário e pressione **Enter**.

O editor insere uma marca de fim do comentário e colocar o cursor entre o início e marcas de comentário final.

3. Digite o **arquivo XML de teste**.

4. Em uma nova linha, digite  e selecione **funcionário** na lista de membros.

O editor adiciona o início de um elemento XML, `<employee>`. Neste momento você pode adicionar atributos para o elemento ou você pode fechar a tag de início digitando `>`.

5. Tipo `>` para a marca de fechamento.

6. O editor adiciona a marca de fim. A marca de fim é adicionada com um a linha subscrita ondulada que indica um erro de validação. A **dica de ferramenta** exibe a mensagem: **o elemento ' Employee ' tem conteúdo incompleto. ' ID ' esperado.**

7. Digite `<` e selecione ID na lista de membros. Digite `>`.

O editor adicione o elemento XML, `<ID></ID>`, e posicionar o cursor após a marca de início de identificação.

8. Digite **ABC**.

O texto **ABC** tem um sublinhado ondulado. A **dica de ferramenta** exibe a mensagem: **o elemento ' ID ' tem um valor inválido de acordo com seu tipo de dados.**

9. Clique com o botão direito do mouse no elemento ID e selecione **ir para definição**.

O editor abre o arquivo *HireDate.xsd* em uma nova janela de documento e posiciona o cursor sobre a definição do elemento de esquema de ID.

10. Retorne ao arquivo XML e substitua o texto **ABC** por **123**.

O sublinhado ondulado e a **dica de ferramenta** são apagados sob o valor do elemento ID. A **dica de ferramenta** para a marca de fim de funcionário agora exibe a mensagem: **o elemento ' Employee ' tem conteúdo incompleto. ' Data de contratação ' esperado.**

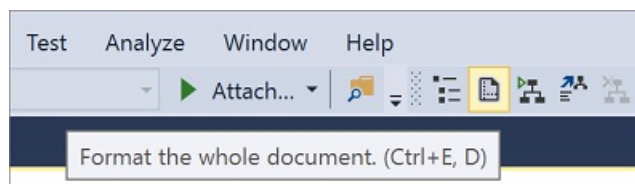
11. Coloque o cursor após a marca de fim de ID, digite em `<`, selecione **contratação-Data** na lista de membros e digite `>`.

O editor adicione o elemento XML, `<hire-date></hire-date>`, e posicionar o cursor após a marca de início da data de admissão.

12. Digite **2003-01-10** para o valor de data de contratação.

## Para formatar o documento XML

- Selecione o botão **Formatar documento** na barra de ferramentas do editor de XML ou pressione **Ctrl + E,D**.



O documento XML é reformatado.

## Para salvar o documento XML

1. No menu **arquivo**, selecione **Salvar como**.

A caixa de diálogo **salvar arquivo como** é exibida. O nome de arquivo padrão é *'xmlarquivo1'*.

2. Insira o nome do arquivo e o local para o documento XML e clique em **salvar**.

## arquivo. xsd contratado

O seguinte arquivo de esquema é usado neste passo a passos:

```
<?xml version="1.0"?>
<xs:schema attributeFormDefault="unqualified"
  elementFormDefault="qualified" targetNamespace="urn:empl-hire"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="employee">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="ID" type="xs:unsignedShort" />
        <xs:element name="hire-date" type="xs:date" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

## Confira também

- [Editor de XML](#)

# Como: selecionar os esquemas XML a serem usados

14/05/2021 • 2 minutes to read

O editor de XML fornece um cache de esquema localizado no diretório `%VSINSTALLDIR%\xml\Schemas`. O cache de esquema inclui esquemas XML conhecidos que são usados para validação do IntelliSense e de documento XML.

Use a propriedade de documento **esquemas** para selecionar um ou mais esquemas de linguagem de definição de esquema XML (XSD). Você pode selecionar esquemas do cache de esquema ou em outro lugar.

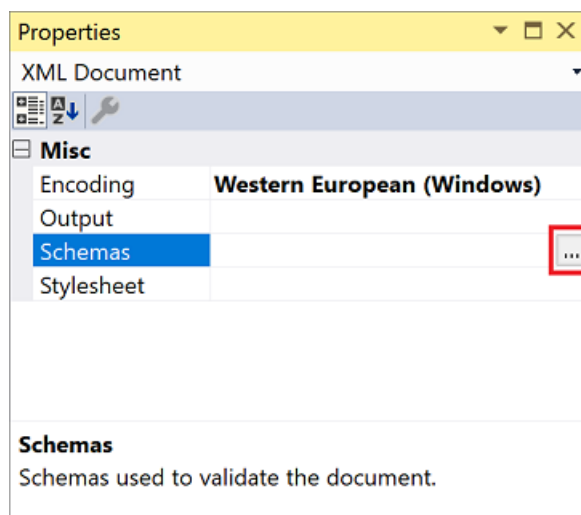
Os esquemas que você especificar são salvos em um arquivo de opções de usuário da solução (oculta) (`.Suo`), juntamente com todas as outras propriedades de documento XML. Como resultado, você não precisará reinserir esses valores na próxima vez que abrir a solução.

## NOTE

O editor pode validar usando um esquema embutido ou um esquema referenciado pelo `xsd:schemaLocation` atributo. Para obter mais informações, consulte [validação de documento XML](#).

## Para selecionar um esquema XML do cache de esquema

1. Abrir um arquivo no editor XML.
2. Na janela Propriedades do documento, clique no campo **esquemas**. Quando o botão procurar (...) for exibido, clique nele.



A [caixa de diálogo esquemas XML](#) é aberta. A caixa de diálogo lista todos os esquemas com um. extensão `XSD` no cache de esquema (incluindo esquemas referenciados no arquivo `catalog.xml`) e também qualquer esquema que esteja na solução atual, aberto no Visual Studio, referenciado em um `xsd:schemaLocation` atributo ou referenciado na propriedade **esquemas**.

3. Selecione os esquemas para usar a validação seguindo um destes procedimentos:
  - Selecione um esquema listado na caixa de diálogo **esquemas XML**, clique na coluna **usar** e, em seguida, selecione **usar este esquema**.

-ou-

- Selecione vários esquemas listados na caixa de diálogo **esquemas XML** e clique com o botão direito do mouse e selecione **usar este esquema**.

4. Selecione **OK**.

A lista de esquemas selecionados é copiada de volta para a propriedade de documento **esquemas** .

## Para adicionar um esquema XML ao cache de esquema

1. Na janela Propriedades do documento, clique no botão no campo **esquemas** .
2. Clique em **Adicionar**.

A caixa de diálogo **abrir esquema XSD** é aberta.

3. Procurar e selecione os esquemas para adicionar ao cache de esquema.
4. Clique em **Abrir**.

Os esquemas são adicionados ao cache de esquema e o valor de coluna de **uso** é definido para **usar esse esquema**.

## Para excluir um esquema XML do cache de esquema

1. Na janela Propriedades do documento, clique no botão no campo **esquemas** .
2. Selecione o esquema a ser removido e clique em **remover**.

O esquema é removido do cache de memória do esquema, mas não é removido do sistema de arquivos.

### NOTE

Se você ainda tiver uma referência ao esquema por meio de um `schemaLocation` atributo ou se uma correspondência for `targetNamespace` **removida** , isso não funcionará nessa situação devido à associação automática. Nesse caso, é recomendável que você marque o esquema como **não usar esquemas selecionados** na coluna **usar** .

## Confira também

- [Cache de esquema](#)
- [Caixa de diálogo esquemas XML](#)
- [Editor de XML](#)

# Como: criar um esquema XML a partir de um documento XML

14/05/2021 • 2 minutes to read

O editor de XML permite que você crie um esquema XSD (linguagem de definição de esquema XML) a partir de um documento XML. O arquivo XML determina como o esquema é gerado da seguinte maneira:

- Se o documento XML não tiver nenhum esquema ou DTD (definição de tipo de documento) associado a ele, os dados no documento XML serão usados para inferir um novo esquema XML.
- Se o documento XML contiver um DTD associado, o DTD externo e o subconjunto interno serão convertidos em um esquema XML correspondente.
- Se o documento XML contiver dados internos com um esquema XDR reduzido de dados XML, o esquema XDR será convertido em um esquema XML correspondente.

Os esquemas que são criados são usados para fornecer o IntelliSense para o arquivo XML.

Para obter mais informações sobre o mecanismo de inferência de esquema, consulte [inferir um esquema XML](#).

## Para criar um esquema XML

1. Abra um arquivo XML no Visual Studio.
2. Na barra de menus, escolha **XML > criar esquema**.

Um documento de esquema XML é criado e aberto para cada namespace encontrado no arquivo XML.

Cada esquema é aberto como um arquivo variado temporário. Os esquemas podem ser salvos no disco, adicionados ao seu projeto ou descartados.

## Consulte também

- [Editor de XML](#)

# Como executar uma transformação XSLT do editor de XML

14/05/2021 • 2 minutes to read

O editor de XML permite associar uma folha de estilos XSLT a um documento XML, executar a transformação e exibir a saída. A saída resultante de transformação XSLT são exibidas em uma nova janela do documento.

A propriedade **output** especifica o nome do arquivo para a saída. Se a propriedade de **saída** estiver em branco, um nome de arquivo será gerado no diretório temporário. A extensão do arquivo é baseada no `xsl:output` elemento em sua folha de estilos e pode ser *XML*, *txt* ou *htm*.

Se a propriedade de **saída** especificar um nome de arquivo com um *htm* ou extensão *HTML*, a saída XSLT é visualizada usando um navegador da Web. Todas as outras extensões de arquivo são abertas usando o editor padrão escolhido pelo Visual Studio. Por exemplo, se a extensão de arquivo for *XML*, o Visual Studio usa o editor de XML.

## Executar uma transformação XSLT de um arquivo XML

1. Abra um documento XML no editor de XML.
2. Associar uma folha de estilos XSLT com o documento XML.
  - Adicione uma instrução de processamento de `xml-stylesheet` para o documento XML. Por exemplo, adicione a seguinte linha ao prólogo do documento:

```
<?xml-stylesheet type='text/xsl' href='filename.xsl'?>
```

-OU-
  - Adicione a folha de estilos XSLT usando a janela **Propriedades**. Com o arquivo XML aberto no editor, clique com o botão direito do mouse em qualquer lugar no editor e escolha **Propriedades**. Na janela **Propriedades**, clique no campo **folha de estilos** e escolha o botão procurar (...). Selecione a folha de estilos XSLT e, em seguida, escolha **abrir**.
3. Na barra de menus, escolha **XML > Iniciar XSLT sem depuração**. Ou pressione **Ctrl + ALT + F5**.

A saída da transformação XSLT é exibida em uma nova janela de documento.

### NOTE

Se não houver nenhuma folha de estilos associada com o documento XML, avisos de uma caixa de diálogo você fornecer a folha de estilos ao uso.

## Executar uma transformação XSLT de uma folha de estilos XSLT

1. Abra uma folha de estilos XSLT no editor de XML.
2. Especifique um documento XML no campo de **entrada** da janela **Propriedades** do documento.

#### NOTE

O documento XML é o documento de entrada usado para a transformação. Se um documento não for especificado quando a transformação XSLT for iniciada, a caixa de diálogo **Abrir arquivo** será exibida e você poderá especificar um documento nesse momento.

3. Na barra de menus, escolha **XML > Iniciar XSLT sem depuração**. Ou pressione **Ctrl + ALT + F5**.

A saída da transformação XSLT é exibida em uma nova janela de documento.

## Especificar um nome de arquivo de saída

Você pode especificar um nome de arquivo de saída para arquivos XML e XSL. Abra a janela **Propriedades** e especifique um nome de arquivo no campo **saída**.

## Confira também

- [Editor de XML](#)



# Como: editar arquivos XML

14/05/2021 • 2 minutes to read

O editor de XML é o novo editor de arquivos XML. Ele pode ser usado em um arquivo XML independente, ou em um arquivo associado a um projeto o Visual Studio. O editor de XML está associado às seguintes extensões de arquivo: *.config*, *.DTD*, *.xml*, *.xsd*, *.XDR*, *.xsl*, *.XSLT* e *.vssettings*. O editor de XML também está associado a qualquer outro tipo de arquivo que não tenha um editor específico registrado e que contenha conteúdo XML ou DTD.

## NOTE

Os documentos XHTML são tratados pelo Editor de HTML.

Para editar um arquivo XML, abra o arquivo que você deseja editar.

## Adicionar um novo arquivo XML a um projeto

1. No menu **projeto**, selecione **Adicionar novo item**.
2. Selecione **arquivo XML** no painel **modelos**.
3. Insira o nome do arquivo no campo **nome** e pressione **Adicionar**.

O arquivo XML é adicionado ao projeto e é aberto no editor de XML. O arquivo contém a declaração XML padrão, `<?xml version="1.0" encoding="utf-8" ?>`.

## Adicionar um arquivo XML existente a um projeto

1. No menu **Projeto**, selecione **Adicionar Item Existente**.

A caixa de diálogo **Adicionar item existente** é exibida.

2. Selecione um arquivo XML e pressione **Adicionar**.

## Criar um novo arquivo XML ou XSLT

1. No menu **arquivo**, selecione **novo**.

A caixa de diálogo **Novo Arquivo** será exibida.

2. Selecione o **arquivo XML** para criar um novo arquivo XML; ou então, selecione **arquivo XSLT** para criar uma nova folha de estilo XSLT.
3. Selecione **Abrir**.

## Criar um projeto vazio para arquivos XML

1. No menu **arquivo**, selecione **novo > projeto**.

A caixa de diálogo **Novo Projeto** aparecerá.

2. Selecione o idioma de código de sua escolha e, em seguida, selecione o modelo de **projeto vazio (.NET Framework)**.

3. Selecione **OK**.

1. No menu **arquivo** , selecione **novo > projeto**.

2. Insira **projeto vazio** na caixa de pesquisa de modelo, selecione o modelo de **projeto vazio (.NET Framework)** e, em seguida, selecione **Avançar**.

3. Selecione **Criar**.

4. Adicionar arquivos XML ao projeto.

O editor de XML localiza os esquemas que você adiciona a este projeto e os usa para validação e IntelliSense em qualquer XML, esquema ou arquivos XSLT que você editar enquanto este projeto está aberto.

## Consulte também

- [Editor de XML](#)
- [Propriedades do documento XML, janela Propriedades](#)
- [Como: criar um esquema XML a partir de um documento XML](#)

# Passo a passo: usando XSLT IntelliSense

14/05/2021 • 2 minutes to read

Essa explicação passo a passo demonstra como usar IntelliSense do XSLT para completar automaticamente o valor de alguns atributos.

## Para usar o IntelliSense no atributo de nome de elementos `xsl:with-param` e `xsl:call-template`

1. Crie um novo arquivo XSLT e copie no seguinte código:

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
<!-- These 2 elements effectively assign
    $messages = resources/en.xml/<messages>,
    then $messages is used in the "localized-message" template. -->
<xsl:param name="lang">en</xsl:param>
<xsl:variable name="messages"
    select="document(concat('resources/', $lang, '.xml'))/messages"/>

<xsl:template name="msg23" match="msg23">
</xsl:template>

<xsl:template name="localized-message">
    <xsl:param name="msgcode"/>
    <!-- Show message string. -->
    <xsl:message terminate="yes">
        <xsl:value-of select="$messages/message[@name=$msgcode]"/>
    </xsl:message>
</xsl:template>
</xsl:stylesheet>
```

2. Insira o cursor depois `<xsl:template name="msg23" match="msg23">` e pressione **Enter**. Inicie digitando o elemento `xsl:call-template` a seguir:

```
<xsl:call-template name="localized-message">
</xsl:call-template>
```

A lista de nomes de modelo aparece no atributo `name=""` do elemento `xsl:call-template` à medida que você digita.

3. Insira o cursor depois `<xsl:call-template name="localized-message">` e pressione **Enter**. Inicie digitando o elemento `xsl:with-param` a seguir:

```
<xsl:with-param name="msgcode">msg23</xsl:with-param>
```

A lista de nomes de parâmetro aparece no atributo `name=""` do elemento `xsl:with-param`.

## Para usar o IntelliSense no atributo do modo de um elemento `xsl:apply-templates`

1. Crie um novo arquivo XSLT e copie no seguinte código:

```

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:template match="/">
    <HTML>
      <BODY>
        <TABLE>
          <xsl:apply-templates select="customers/customer">
            <xsl:sort select="state"/>
            <xsl:sort select="name"/>
          </xsl:apply-templates>
        </TABLE>
      </BODY>
    </HTML>
  </xsl:template>
  <xsl:template match="customer">
    <TR>
      <xsl:apply-templates select="name" />
      <xsl:apply-templates select="address" />
      <xsl:apply-templates select="phone" />
    </TR>
  </xsl:template>
  <xsl:template match="name">
    <TD STYLE="font-size:14pt font-family:serif">
      <xsl:apply-templates />
    </TD>
  </xsl:template>
  <xsl:template match="address">
    <TD>
      <xsl:apply-templates />
    </TD>
  </xsl:template>
  <xsl:template match="phone">
    <TD>
      <xsl:apply-templates />
    </TD>
  </xsl:template>
  <xsl:template match="phone" mode="accountNumber">
    <xsl:param name="Area_Code"/>
    <TD STYLE="font-style:italic">
      1-<xsl:value-of select="."/>-001
    </TD>
  </xsl:template>
</xsl:stylesheet>

```

2. Insira o cursor depois `<xsl:apply-templates select="phone" />` e pressione **Enter**. Inicie digitando o elemento `xsl: apply-templates` a seguir:

```

<xsl:apply-templates select="phone" mode="accountNumber">

```

A lista de modos de modelo aparece no atributo `mode=""` do elemento `xsl:apply-templates`.

## Para usar o IntelliSense nos atributos stylesheet-prefix e result-prefix de um elemento xsl:namespace-alias

1. Crie um novo arquivo XSLT e copie no seguinte código:

```

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:alt="http://www.w3.org/1999/XSL/Transform-alternate"
version="1.0">
  <xsl:param name="browser" select="'InternetExplorer'"/>
  <xsl:template match="/">
    <alt:stylesheet>
      <xsl:choose>
        <xsl:when test="$browser='InternetExplorer'">
          <alt:import href="IERoutines.xsl"/>
          <alt:template match="/">
            <div>
              <alt:call-template name="showTable"/>
            </div>
          </alt:template>
        </xsl:when>
        <xsl:otherwise>
          <alt:import href="OtherBrowserRoutines.xsl"/>
          <alt:template match="/">
            <div>
              <alt:call-template name="showTable"/>
            </div>
          </alt:template>
        </xsl:otherwise>
      </xsl:choose>
    </alt:stylesheet>
  </xsl:template>
</xsl:stylesheet>

```

## 2. Insira o cursor depois

```

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:alt="http://www.w3.org/1999/XSL/Transform-alternate" version="1.0">

```

e pressione **Enter**. Inicie digitando o elemento `xsl:namespace-alias` a seguir:

```

<xsl:namespace-alias stylesheet-prefix="alt" result-prefix="xsl"/>

```

Observe como a lista de prefixos apareceu nos atributos `stylesheet-prefix` e `result-prefix` do elemento de `xsl:namespace-alias`.

## Confira também

- [Recursos do IntelliSense o editor XML](#)

# Walkthrough: usar a hierarquia XSLT

14/05/2021 • 2 minutes to read

A ferramenta da hierarquia XSLT simplifica muitas tarefas de desenvolvimento XML. Uma folha de estilos XSLT frequentemente usa `includes` e instruções de `imports`. A compilação parte da folha de estilos principal, mas quando você verá um erro no resultado de criar uma folha de estilos XSLT, o erro pode vir de uma fonte diferente da folha de estilos principal. Corrigir o erro ou editar a folha de estilos podem exigir acesso incluiu ou importaram folhas de estilos. Percorrer de folha de estilo no depurador pode abrir folhas de estilo embutidas e importados, e você pode querer adicionar um ponto de interrupção em algum ponto de uma ou mais das folhas de estilo embutidas.

Outro cenário onde a ferramenta da hierarquia XSLT pode ser útil é colocando pontos de interrupção nas regras de modelo interno. As regras de modelo são modelos especiais gerados para cada modo de folha de estilos e chamados por `xsl:apply-templates` quando nenhum outro modelo corresponde ao nó. Para implementar a depuração em regras de modelos internos, o depurador XSLT gerencia o arquivo com as regras na pasta temporária e compilar-las juntamente com a folha de estilos principal. Sem entrar no código de qualquer `xsl:apply-template`, pode ser difícil localizar as folhas de estilos que foram incluídas na folha de estilos principal ou localizar e abrir a folha de estilos com as regras de modelo interno.

O exemplo neste tópico demonstra a depuração em uma folha de estilos referenciada.

## Para depurar em uma folha de estilos referenciada

1. Abrir um documento XML no Visual Studio. Este exemplo usa o seguinte documento:

```
<?xml version="1.0" encoding="utf-8"?>
<?xml-stylesheet type="text/xsl" href="xslinclude.xsl"?>
<COLLECTION>
  <BOOK>
    <TITLE>Lover Birds</TITLE>
    <AUTHOR>Cynthia Randall</AUTHOR>
    <PUBLISHER>Lucerne Publishing</PUBLISHER>
  </BOOK>
  <BOOK>
    <TITLE>The Sundered Grail</TITLE>
    <AUTHOR>Eva Corets</AUTHOR>
    <PUBLISHER>Lucerne Publishing</PUBLISHER>
  </BOOK>
  <BOOK>
    <TITLE>Splish Splash</TITLE>
    <AUTHOR>Paula Thurman</AUTHOR>
    <PUBLISHER>Scootney</PUBLISHER>
  </BOOK>
</COLLECTION>
```

2. Adicione o seguinte *xslincludefile.xsl*:

```
<?xml version='1.0'?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xml:space="preserve">

  <xsl:template match="TITLE">
    Title - <xsl:value-of select="."/><BR/>
  </xsl:template>

  <xsl:template match="AUTHOR">
    Author - <xsl:value-of select="."/><BR/>
  </xsl:template>

  <xsl:template match="PUBLISHER">
    Publisher - <xsl:value-of select="."/><BR/><!-- removed second <BR/> -->
  </xsl:template>

</xsl:stylesheet>
```

3. Adicione o seguinte arquivo *xslinclude.xsl*:

```
<?xml version='1.0'?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

  <xsl:output method="xml" omit-xml-declaration="yes"/>

  <xsl:template match="/">
    <xsl:for-each select="COLLECTION/BOOK">
      <xsl:apply-templates select="TITLE"/>
      <xsl:apply-templates select="AUTHOR"/>
      <xsl:apply-templates select="PUBLISHER"/>
      <BR/>
      <!-- add this -->
    </xsl:for-each>
  </xsl:template>

  <!-- The following template rule will not be called,
  because the related template in the including stylesheet
  is called. If we move this template so that
  it follows the xsl:include instruction, this one
  will be called instead.-->
  <xsl:template match="TITLE">
    <DIV STYLE="color:blue">
      Title: <xsl:value-of select="."/>
    </DIV>
  </xsl:template>

  <xsl:include href="xslincludefile.xsl" />
</xsl:stylesheet>
```

4. Adicione um ponto de interrupção na instrução `<xsl:include href="xslincludefile.xsl" />` .

5. Inicie a depuração.

6. Quando o depurador parar na instrução `<xsl:include href="xslincludefile.xsl" />` , pressione o botão **Step Into** . A depuração pode continuar na folha de estilos referenciada. A hierarquia é visível e o designer o caminho correto.

## Confira também

- [Criador de perfil XSLT](#)

# Propriedades de documento XML, janela Propriedades

14/05/2021 • 2 minutes to read

A janela **Propriedades** fornece informações básicas sobre o documento que está ativo no editor de XML. As propriedades que estão disponíveis varia dependendo do tipo de documento XML que está atualmente ativa.

## NOTE

Todas as propriedades de documento XML são salvas na solução. Como resultado, você não tem que digitar novamente esses valores na próxima vez que você abrir a solução.

## Codificação

A codificação de caractere para o arquivo. Alterar essa propriedade também altera o atributo de codificação na declaração XML, e vice-versa. A nova codificação é usada para codificar o arquivo quando você salva o arquivo.

## Entrada

O documento de entrada associado com a folha de estilos XSLT. Ele é usado pelos comandos **Start XSLT**, por exemplo, **XML > Start XSLT sem depuração**. Um documento pode ser selecionado usando o botão procurar (...).

Essa propriedade só é visível quando um arquivo XSLT é aberto no editor.

## Saída

O arquivo que é gerado para transformar um documento XML.

Se um arquivo não for especificado, um nome de arquivo padrão será gerado com base no `method` atributo no `xsl:output` elemento, que determina a extensão do arquivo. O arquivo padrão é localizado no diretório temporário do usuário atual.

## Esquemas

Os esquemas a ser usado para validação. O botão abre a caixa de diálogo **esquemas XSD**, que pode ser usada para selecionar os esquemas a serem usados.

Você também pode ir para o caminho para esquemas. Se vários esquemas são especificados, cada caminho de esquema deve ser colocado entre aspas duplas.

## Xls

O arquivo XSLT usado para transformar o documento quando os comandos **Iniciar Depuração XSLT** e **Iniciar XSLT sem depuração** são usados. Se esse campo estiver em branco, o editor usará o valor fornecido na `xml:stylesheet` instrução de processamento do documento ou solicitará um nome de arquivo.

Ao editar um arquivo XSLT, essa propriedade pode ser usada para especificar que uma folha de estilos diferente deve ser usada quando o comando **Iniciar Depuração XSLT** ou **Iniciar XSLT sem depuração** está selecionado. Por exemplo, talvez você queira fazer isso quando estiver editando uma folha de estilos que está incluída em uma folha de estilos pai.

## Confira também



- [Editor de XML](#)

# Caixa de diálogo de esquemas XML

14/05/2021 • 2 minutes to read

A caixa de diálogo **esquemas XML** é usada para selecionar quais esquemas XSD (linguagem de definição de esquema XML) associar a um documento XML. Você pode selecionar um esquema de cache do esquema, ou especificar um esquema que não está localizado no cache. Os esquemas selecionados são considerados parte de um conjunto de esquema. O esquema é usado para o IntelliSense e também validação de documento XML.

Você pode acessar a caixa de diálogo **esquemas XML** clicando no botão **esquemas** na janela Propriedades do documento ou selecionando **esquemas** no menu **XML**.

## Lista de elementos de interface do usuário

### Uso

Selecione como o esquema XML deve ser usada.

- **Automático.** Este esquema não está em uso pelo documento atual mas está disponível para a associação automática. Se o documento XML declarar um namespace que corresponde `targetNamespace` deste esquema, o esquema será associado e é automaticamente encapsulado no conjunto de esquema.
- **Use este esquema.** Este esquema está sendo usado pelo documento atual. Qualquer o usuário que solicitou explicitamente este esquema está usado clicando nessa coluna, ou o esquema foi associado automaticamente com base em `targetNamespace` correspondente.
- **Não use esquemas selecionados.** Este esquema não é usado pelo documento atual, mesmo se o esquema tem `targetNamespace` correspondente. Essa configuração pode ser útil para resolver conflitos quando há mais de uma versão do mesmo esquema no cache ou na solução de esquema.

### Namespace de destino

Exibe o namespace de destino associada com o esquema XML.

### Nome do Arquivo

Exibe o nome do arquivo de esquema XML.

### Adicionar

Abre a caixa de diálogo **abrir esquema XSD**, que permite que você selecione esquemas adicionais para adicionar ao conjunto de esquema. Quando você adiciona um esquema ao conjunto de esquema, o valor de coluna de **uso** é definido para **usar esse esquema**.

### Remover

Remove o esquema do dataset selecionado de esquema. Remove o esquema de cache de memória do esquema, mas não no sistema de arquivos.

## Confira também

- [Como: selecionar os esquemas XML a serem usados](#)
- [Cache de esquema](#)

# Visão geral do designer de esquema XML

14/05/2021 • 2 minutes to read

O Designer de Esquema XML (Designer XSD) é uma ferramenta gráfica por meio da qual é possível visualizar um esquema definido em níveis distintos de abstração. Os principais componentes do designer XSD são:

- [XML Schema Explorer](#), que permite procurar e navegar na árvore de esquema XML e realizar pesquisas.
- O [modo de exibição inicial](#), que é o ponto de partida para o designer XSD. Na exibição inicial, você pode navegar para outras exibições do designer XSD e ver os detalhes do conjunto de esquema.
- A [exibição de gráfico](#), que permite que você veja uma visão geral de um conjunto de esquema e as relações entre os nós de esquema.
- A [exibição do modelo de conteúdo](#), que fornece uma representação gráfica dos detalhes de nós de esquema locais e globais, incluindo tipos simples e complexos, elementos, grupos, atributos e grupos de atributos.

# XML Schema Explorer

14/05/2021 • 3 minutes to read

O **XML Schema Explorer** é integrado com Microsoft Visual Studio e o editor de XML para permitir que você trabalhe com esquemas XSD (linguagem de definição de esquema XML). Quando você abre um arquivo de esquema XML, o nó de **conjunto de esquema** aparece no **XML Schema Explorer**. Todos os esquemas incluídos, importados ou redefinidos para seu arquivo de destino, bem como todos os arquivos que são referenciados por meio de uma `include` `import` instrução ou, também aparecem no **XML Schema Explorer**.

O **XML Schema Explorer** permite que você faça o seguinte:

- Obter uma visão geral rápido do conjunto de esquema.
- Procurar e navegar na árvore.
- Realizar pesquisas de palavra-chave e específicas do esquema. Para obter mais informações, consulte [pesquisando o conjunto de esquema](#).
- Adicionar os resultados da pesquisa ao modo de exibição de gráfico ou de modelo de conteúdo
- Classificar a árvore pela ordem de documento, tipo ou nome. Para obter mais informações, consulte [classificar, filtrar e agrupar](#).
- Abra o editor de XML e salte para os locais de código no arquivo XSD. Para obter mais informações, consulte [integração com o editor de XML](#).
- Gere o exemplo de XML para elementos globais.

O **XML Schema Explorer** fornece uma exibição hierárquica do conjunto de esquema por meio de uma exibição de árvore. O **XML Schema Explorer** também fornece pesquisa, filtragem, navegação e classificação. Para acessar o **XML Schema Explorer**, siga um destes procedimentos:

- Se você estiver na [exibição inicial](#), clique no link do **Gerenciador de esquema XML**.
- Se você estiver na [exibição de gráfico](#) ou no [modo de exibição de modelo de conteúdo](#) e tiver nós em seu espaço de trabalho, use o menu de contexto (clique com o botão direito do mouse) para selecionar o Gerenciador de **esquema XML**.
- Você também pode selecionar o **Gerenciador de esquema XML** no menu **Exibir**.
- Você pode acessar o **XML Schema Explorer** de um arquivo `.vb` que tem um literal XML Visual Basic associado a um arquivo `.xsd`. Para ver o conjunto de esquema no **Gerenciador de esquema XML**, clique com o botão direito do mouse em um nó XML em um literal XML ou em uma importação de namespace XML e selecione o comando **Mostrar no Gerenciador de esquema**. Para obter mais informações, consulte [integração de literais XML com o XML Schema Explorer](#).

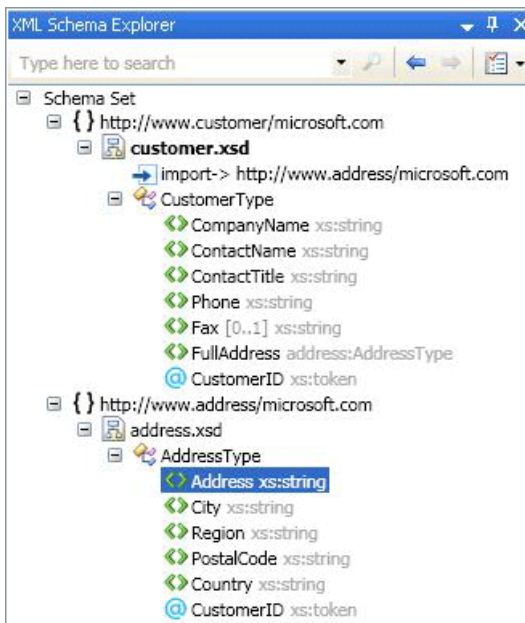
## Modo de exibição de árvore

O **XML Schema Explorer** exibe informações de conjunto de esquema pré-compilado em uma estrutura de árvore. A estrutura de árvore é organizada da seguinte maneira:

- No nível superior está o nó do conjunto de esquema.
- O segundo nível contém os namespaces.
- O terceiro nível contém os arquivos.

- O quarto nível contém os nós globais. Isso pode incluir elementos, grupos, tipos complexos, tipos simples, atributos, grupos de atributo e instruções `include`, `import` e `redefine`.

A seguir veja um exemplo de uma estrutura de árvore:



## Seleção e ativação

Para realçar e selecionar um nó, clique uma vez no Schema Explorer.

Para ativar um nó, clique duas vezes nele ou pressione **Enter** quando o nó for selecionado.

- Ativar um nó abre o arquivo no qual o nó está definido (se o arquivo já não estiver aberto) e seleciona o nó no arquivo.
- Ativar um nó de arquivo abre o arquivo selecionado (se ele já não estiver aberto) e destaca o nó `<schema>`.
- Ativar um SchemaSet ou um nó de namespace não fará nada.

## Arrastar e soltar nós

Você pode arrastar e soltar nós globais, nós de arquivo e nós de namespace em uma exibição do Designer XSD. Se o modo de exibição atual for o [modo de exibição de início](#), arrastar um nó para a exibição abrirá o modo de exibição de [gráfico](#). Se a exibição atual for o modo de [exibição de modelo de conteúdo](#) ou de gráfico, a exibição não será alterada quando você soltar um nó nele.

O descarte de arquivos na exibição adicionará todos os nós globais no arquivo ao [espaço de trabalho do designer XSD](#). Soltar namespaces na exibição adicionará todos os nós globais no namespace para o workspace. O workspace é compartilhado entre todas as visualizações.

Você não pode arrastar e soltar nós locais ou importações.

## Confira também

- [Como adicionar nós ao espaço de trabalho do XML Schema Explorer](#)

# Pesquisar o conjunto de esquemas

14/05/2021 • 2 minutes to read

O **XML Schema Explorer** permite que você pesquise o conjunto de esquema das seguintes maneiras:

- Pesquisa de palavras-chave.
- Pesquisa Esquema- específica.

## Pesquisa de palavra-chave

Você executa pesquisas de palavra-chave inserindo uma subcadeia de caracteres na caixa de texto **Pesquisar schemaSet** da barra de ferramentas do **Gerenciador de esquema XML**.



O **XML Schema Explorer** pesquisa o conjunto de esquema para os seguintes atributos:

- Alguns atributos de `name` ou de `ref` que corresponderem a palavra-chave especificada. Você pode encontrar elementos, atributos, tipos e assim por diante, por nome.
- Atributos de `schemaLocation` de incluem instruções.
- Atributos de `namespace` de instruções de importação.

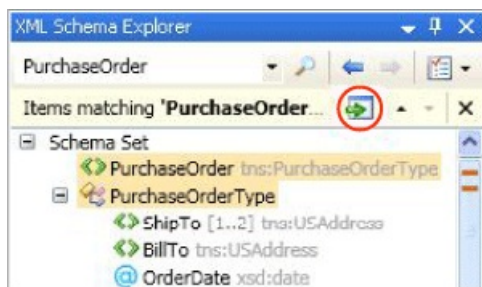
## Pesquisa específica do esquema

O **XML Schema Explorer** também inclui pesquisas internas que você pode acessar usando o menu de contexto (clique com o botão direito do mouse) do **XML Schema Explorer**. Para obter mais informações sobre menus de contexto disponíveis, consulte [menus de contexto](#). Você também pode executar uma pesquisa específica de esquema do modo de exibição inicial; para obter mais informações, consulte a seção "detalhes do conjunto de esquema" no tópico [Start View](#).

## Exibir e navegar pelos resultados da pesquisa

Depois que a pesquisa é concluída, o painel de resultados de resumo é adicionado à barra de ferramentas com os resultados da pesquisa. Os resultados da pesquisa também são realçados no **XML Schema Explorer** e marcados por tiques na barra de rolagem vertical. Você pode navegar pelos resultados da pesquisa usando o botão **para o próximo resultado da pesquisa** e ir para os botões de **resultado da pesquisa anterior** no painel resultados do resumo da barra de ferramentas do **Gerenciador de esquema XML**; usando as teclas de teclado **F3** e **Shift + F3**; ou clicando nas marcas de escala na barra de rolagem.

Você pode adicionar os resultados da pesquisa ao espaço de trabalho clicando no botão **adicionar nós realçados ao espaço de trabalho** no painel resultados do resumo.



## Limpar resultados da pesquisa

Para limpar os resultados da pesquisa, clique no botão x no painel resultados do resumo da barra de ferramentas de pesquisa do **Gerenciador de esquema XML**.

## Confira também

- [XML Schema Explorer](#)

# Classificação, filtragem e agrupamento (XML Schema Explorer)

14/05/2021 • 2 minutes to read

Este tópico descreve as opções disponíveis por meio do menu **Opções de classificação, filtragem e agrupamento** na barra de ferramentas do **Gerenciador de esquema XML**.

## Opções de filtro

As seguintes opções de filtro estão disponíveis. Por padrão, as opções **Mostrar namespaces** e **Mostrar arquivos de esquema** estão selecionadas.

- **Mostrar namespaces.**
- **Mostrar arquivos de esquema.**
- **Mostrar compositores (sequência/escolha/tudo).**

## Opções de classificação

As seguintes opções de classificação estão disponíveis. O padrão é **classificar por tipo**. As opções **classificar por** não se aplicam a arquivos e namespaces.

- **Classificar por tipo.**
- **Classificar por nome.**
- **Ordem do documento.**

### Classificar por Tipo

Quando a opção **classificar por tipo** é selecionada, os nós globais são classificados na seguinte ordem. Nós são classificados em ordem alfabética dentro de cada grupo.

1. `import` Nós.
2. `include` Nós.
3. `redefine` Nós.
4. `attribute` Nós.
5. `attributeGroup` Nós.
6. `complexType` Nós.
7. `simpleType` Nós.
8. `element` Nós.
9. `group` Nós.

### Classificar por Nome

Quando a opção **classificar por nome** for selecionada, os nós globais serão classificados na seguinte ordem:

1. `import` Nós (em ordem alfabética de namespaces).



2. `include` Nós (em ordem alfabética de `schemaLocation` atributos).
3. `redefine` Nós (em ordem alfabética de `schemaLocation` atributos).
4. Outros nós globais em ordem alfabética.

### Ordem de documento

A opção **ordem de documento** está disponível quando a opção **Mostrar arquivos de esquema** está selecionada. Quando a **ordem do documento** é selecionada, os nós globais são exibidos na ordem em que aparecem no arquivo de esquema.

## Persistência das opções de classificação/filtro

A classificação, filtragem, e opções de agrupamento são salvas no Registro para cada usuário, não importa qual a solução ou arquivos estavam aberta quando as configurações foram alteradas.

# Menus de contexto (XML Schema Explorer)

14/05/2021 • 5 minutes to read

Um menu de contexto é o menu que aparece quando você clica com o botão direito do mouse em algo. Os seguintes itens de menu de contexto são usados para executar pesquisas esquema- específicas e outras operações.

## Tipo de nó: conjunto de esquema

A tabela a seguir descreve as opções que estão disponíveis para um nó do esquema.

OPÇÃO	DESCRIÇÃO
Elementos de raiz mais provável de apresentação	Localiza e realça todos os elementos globais não são referenciados de elementos globais diferentes de se.
Tipos globais de apresentação	Os localiza e realça todos globais no conjunto de esquema.
Elementos globais de apresentação	Localiza e realces todos elementos globais no conjunto de esquema.
Janela Propriedades	Abre a janela <b>Propriedades</b> (se ainda não estiver aberta). Esta janela exibe informações sobre o nó.

## Tipo de nó: namespace

A tabela a seguir descreve as opções que estão disponíveis para um nó de namespace.

OPÇÃO	DESCRIÇÃO
Mostrar todas as referências de entrada	Localiza e resalta os arquivos que importar o namespace selecionada.
Mostrar todas as referências de saída	Para cada arquivo no namespace selecionado, localiza e resalta o seguinte:  -Todos os namespaces referenciados em instruções de importação sem um <code>schemaLocation</code> atributo. -Todos os arquivos em namespaces diferentes do selecionado que são especificados no <code>schemaLocation</code> atributo nas instruções import e include.
Tipos globais de apresentação	Os localiza e realça todos globais no namespace selecionada.
Elementos globais de apresentação	Localiza e realces todos elementos globais no namespace selecionado.
Janela Propriedades	Abre a janela <b>Propriedades</b> (se ainda não estiver aberta). Esta janela exibe informações sobre o nó.

## Tipo de nó: arquivo

A tabela a seguir descreve as opções que estão disponíveis para um nó de arquivo.

OPÇÃO	DESCRIÇÃO
Mostrar todas as referências de entrada	Localiza e realça todos os arquivos que especificam o arquivo selecionado em atributos de <code>schemaLocation</code> do incluem e importar instruções.
Mostrar todas as referências de saída	Localiza e realces o seguinte:  -Todos os namespaces especificados nos atributos de namespace de todas as instruções de importação que não têm o <code>schemaLocation</code> atributo. -Todos os arquivos especificados nos <code>schemaLocation</code> atributos de todas as instruções import e include.
Tipos globais de apresentação	Os localiza e realça todos globais neste arquivo.
Elementos globais de apresentação	Os localiza e realça todos os elementos globais neste arquivo.
Exibir Código	Abre o arquivo que contém o nó selecionado no editor de XML. O item selecionado no XML Schema Explorer também será selecionado no editor de XML.
Janela Propriedades	Abre a janela <b>Propriedades</b> (se ainda não estiver aberta). Esta janela exibe informações sobre o nó.

## Todos os tipos de nó global

A tabela a seguir descreve as opções que estão disponíveis para todos os nós globais.

OPÇÃO	DESCRIÇÃO
Apresentação no modo de gráfico	Abre a exibição do gráfico. Se o nó selecionado não está no workspace, adicione-o ao workspace e selecione o nó.
Apresentação no modo do modelo de conteúdo	Abre a exibição do modelo de conteúdo. Se o nó selecionado não está no workspace, adicione-o ao workspace e selecione o nó.
Exibir Código	Abre o arquivo que contém o nó selecionado no editor de XML. O item selecionado no XML Schema Explorer também será selecionado no editor de XML.
Janela Propriedades	Abre a janela <b>Propriedades</b> (se ainda não estiver aberta). Esta janela exibe informações sobre o nó.

## Tipo de nó: elemento

Além das opções do nó globais descritos acima, o menu de contexto para nós do elemento tem as seguintes opções:

OPÇÃO	DESCRIÇÃO
Ir para definição de tipo	Navega para a definição de tipo do elemento selecionado. Isso é aplicável quando o tipo que é usado para o elemento é um tipo global.
Vá para o elemento original	Para referências de elemento, navega para a definição real do elemento.
Mostrar todas as referências	Para elementos globais, localiza e realça todas as referências (elementos que têm <code>ref="selectedElement"</code> ) ao elemento selecionado.
Membros de apresentação do grupo de substituição	Para os cabeçotes de um grupo de substituição, localiza e realça todos os elementos que são membros do grupo de substituição de que o elemento selecionado é um membro. Isso mostra participantes diretos e indiretos.
Chefes de grupo de substituição de apresentação	<p>Para elementos globais que são membros de um grupo de substituição, localiza e ressalta os cabeçotes qualquer diretos e indiretos para o elemento selecionado, como o seguinte:</p> <ul style="list-style-type: none"> <li>-Um cabeçalho do grupo de substituição especificado no elemento selecionado.</li> <li>-Um cabeçalho de grupo de substituição especificado em seu elemento de cabeçalho.</li> </ul>
Gerencia o exemplo XML	Disponível somente para os elementos globais. Gerencia um arquivo XML de exemplo para o elemento global.

## Tipo de nó: tipos globais

Além das opções do nó globais descritos acima, o menu de contexto para nós globais do tipo tem as seguintes opções:

OPÇÃO	DESCRIÇÃO
Tipo base de apresentação	Se o tipo selecionado é derivado de um tipo global, navega para o tipo de base do tipo selecionado.
Mostrar todas as referências	Localiza e realces todas as referências para o tipo selecionado. Isso inclui os elementos e atributos de tipo e tipos derivados selecionados do tipo selecionado.
Mostrar todos os tipos derivados	Localiza e realça todos os tipos que direta e indiretamente são derivados do tipo selecionado.
Mostrar todos os predecessores	Mostrar todos os tipos de base pai ().

## Tipo de nó: atributo

Além das opções do nó globais descritos acima, o menu de contexto para nós de atributo tem as seguintes opções:

OPÇÃO	DESCRIÇÃO
Ir para definição de tipo	Quando o tipo que é usado para o atributo é um tipo global, navega para a definição de tipo do atributo selecionado.
Vá para o atributo original	Para referências de atributo, navega para a definição real do atributo.
Mostrar todas as referências	Para atributos globais, localiza e realça todas as referências (outros atributos que têm <code>ref="selectedAttribute"</code> ) para o atributo selecionado.

## Tipo de nó: grupo de atributos

Além das opções do nó globais descritos acima, o menu de contexto para nós do grupo de atributo tem as seguintes opções:

OPÇÃO	DESCRIÇÃO
Ir para Definição	Para referências, navega para a definição real do atributo.
Mostrar todos os membros	Localiza e realces todos os membros do grupo de atributo.
Mostrar todas as referências	Localiza e realça todas as referências (grupos de atributo que têm <code>ref="selectedAttributeGroup"</code> ) para o grupo selecionado de atributo.

## Tipo de nó: grupo nomeado

Além das opções do nó globais descritos acima, o menu de contexto para nós nome de grupo tem as seguintes opções:

OPÇÃO	DESCRIÇÃO
Ir para Definição	Para referências, navega para a definição real do atributo.
Mostrar todos os membros	Localiza e realces todos os membros do grupo chamado.
Mostrar todas as referências	Localiza e realça todas as referências (grupos que têm <code>ref="selectedGroup"</code> ) para o grupo selecionado.

## Confira também

- [XML Schema Explorer](#)
- [Pesquisando o conjunto de esquema](#)

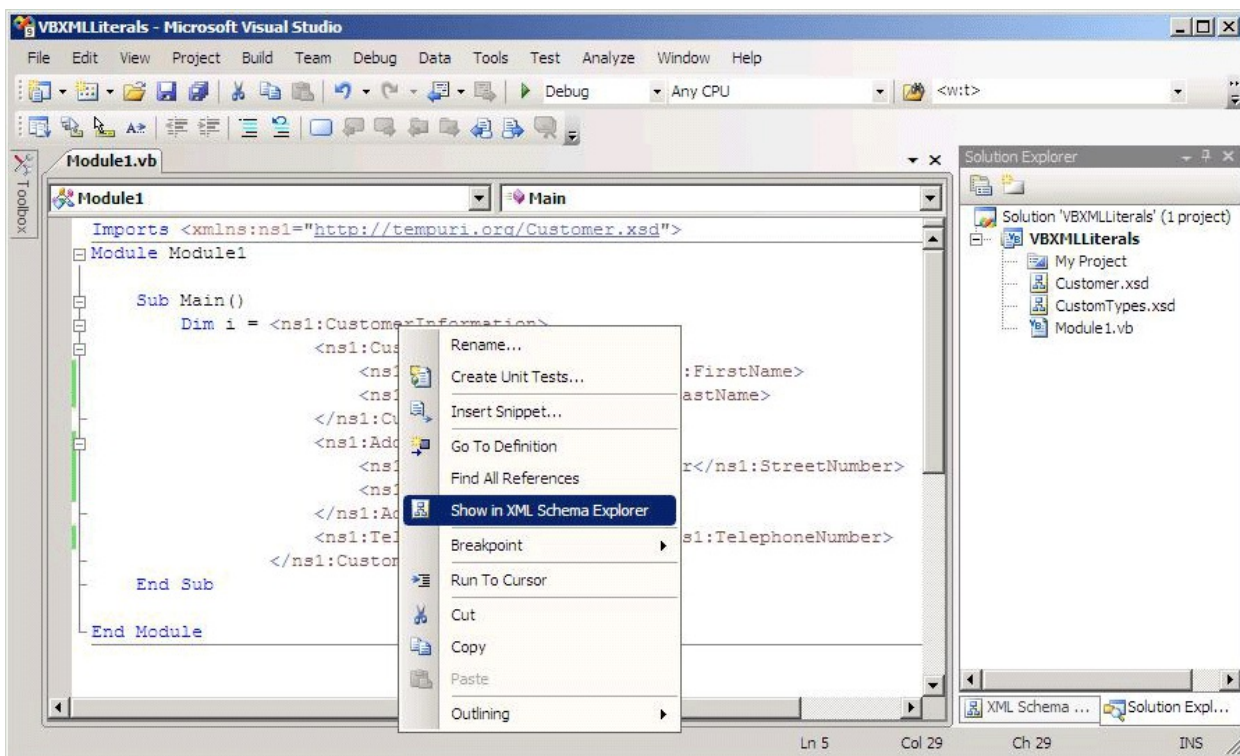
# Integração de literais XML com o XML Schema Explorer

14/05/2021 • 2 minutes to read

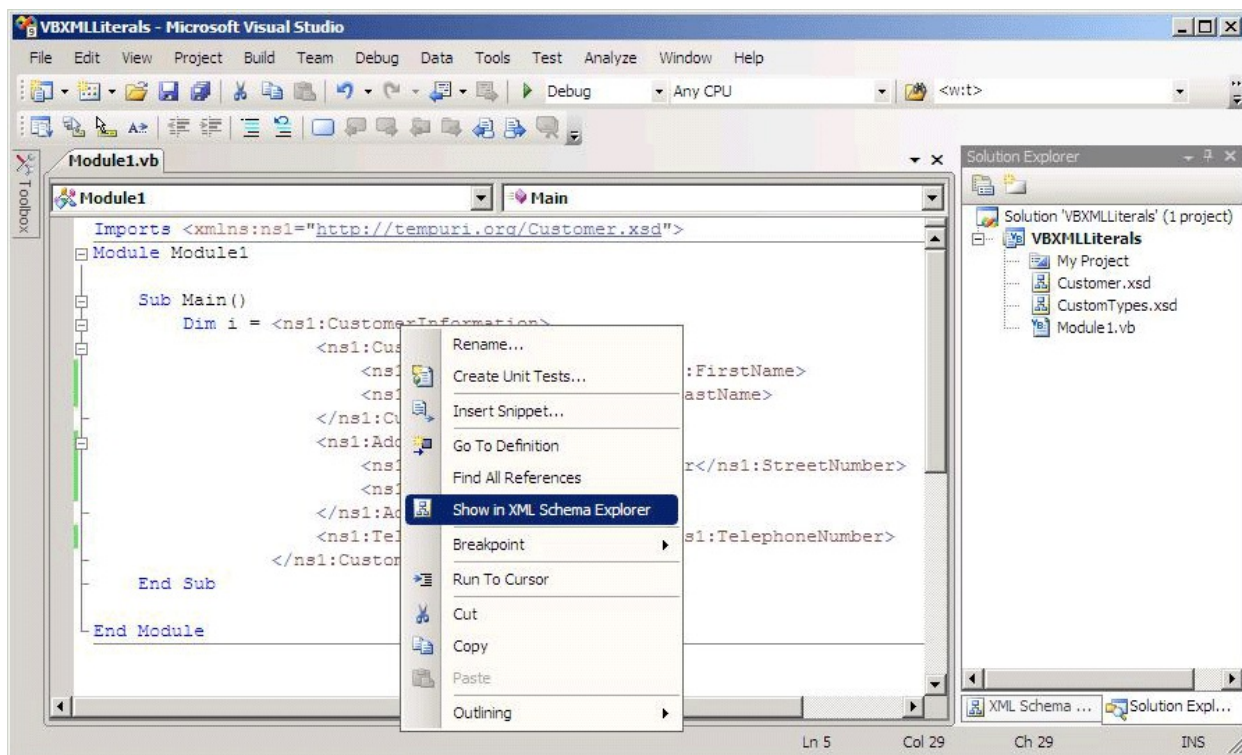
Visual Basic suporta literal XML, que significa que você pode inserir fragmentos XML diretamente no seu código Visual Basic. Para obter mais informações, consulte [visão geral de literais XML](#).

## Como

Se um arquivo XSD em seu projeto de Visual Basic incluir um literal XML, você poderá exibir o conjunto de esquema XML no **XML Schema Explorer**. Para exibir o conjunto de esquema associado a um literal XML, clique com o botão direito do mouse em um nó XML em um literal XML ou em uma importação de namespace XML e selecione **Mostrar no Gerenciador de esquema**.



Isso abre o **Gerenciador de esquema XML** lado a lado com o arquivo Visual Basic.



## Consulte também

- [Como: usar o designer de esquema XML com literais XML](#)

# Exibições de espaço de trabalho do designer de esquema XML

14/05/2021 • 2 minutes to read

O designer de esquema XML (XSD) designer é uma ferramenta gráfica que ajuda você a explorar seus esquemas XML. Além do [XML Schema Explorer](#), que permite procurar e navegar na árvore de esquema XML e realizar pesquisas, o designer XSD fornece três exibições nas quais você pode explorar o esquema XSD em mais detalhes.

- A **exibição inicial** é o ponto de partida para o designer XSD; na exibição inicial, você pode navegar para outras exibições do designer XSD e ver os detalhes do conjunto de esquema.
- A **exibição de gráfico** permite que você veja uma visão geral de um conjunto de esquema e as relações entre os nós de esquema.
- A **exibição de modelo de conteúdo** fornece uma representação gráfica dos detalhes de nós de esquema locais e globais, incluindo tipos simples e complexos, elementos, grupos, atributos e grupos de atributos.

Para começar a explorar os nós nos quais você está interessado, você deve adicioná-los ao espaço de trabalho. O workspace é compartilhado entre todas as visualizações.

## Adicionar nós ao espaço de trabalho

Você pode adicionar nós ao workspace das seguintes maneiras:

- Na seção "detalhes do conjunto de esquema" do [modo de exibição inicial](#), clique no link **Adicionar** ao lado do tipo de nó global.
- Arraste e solte nós globais, nós de arquivo e nós de namespace do **XML Schema Explorer** em qualquer uma das três exibições. Para obter mais informações, consulte a seção "arrastando e descartando nós" no [XML Schema Explorer](#).
- Use o menu de contexto (clique com o botão direito do mouse) no **XML Schema Explorer**. Para obter mais informações, consulte [menus de contexto](#).
- Execute uma pesquisa no explorador do XSD e clique no botão **adicionar nós realçados ao espaço de trabalho** no painel resultados do resumo. Para obter mais informações, consulte [pesquisando o conjunto de esquema](#).

## Alternar modos de exibição

Para alternar modos de exibição, use um destes procedimentos:

- A barra de ferramentas do designer XSD.
- Os menus de contexto (clique com o botão direito do mouse) da exibição do modelo de conteúdo e do modo de exibição de gráfico.
- Os marca de água na página no modo de Início ou a marca de água no modo em branco do modelo de conteúdo ou no modo de gráfico.
- Teclas de pressionamento: **Ctrl + 1** para a exibição de início, **Ctrl + 2** para o modo de exibição de gráfico e **Ctrl + 3** para a exibição do modelo de conteúdo.

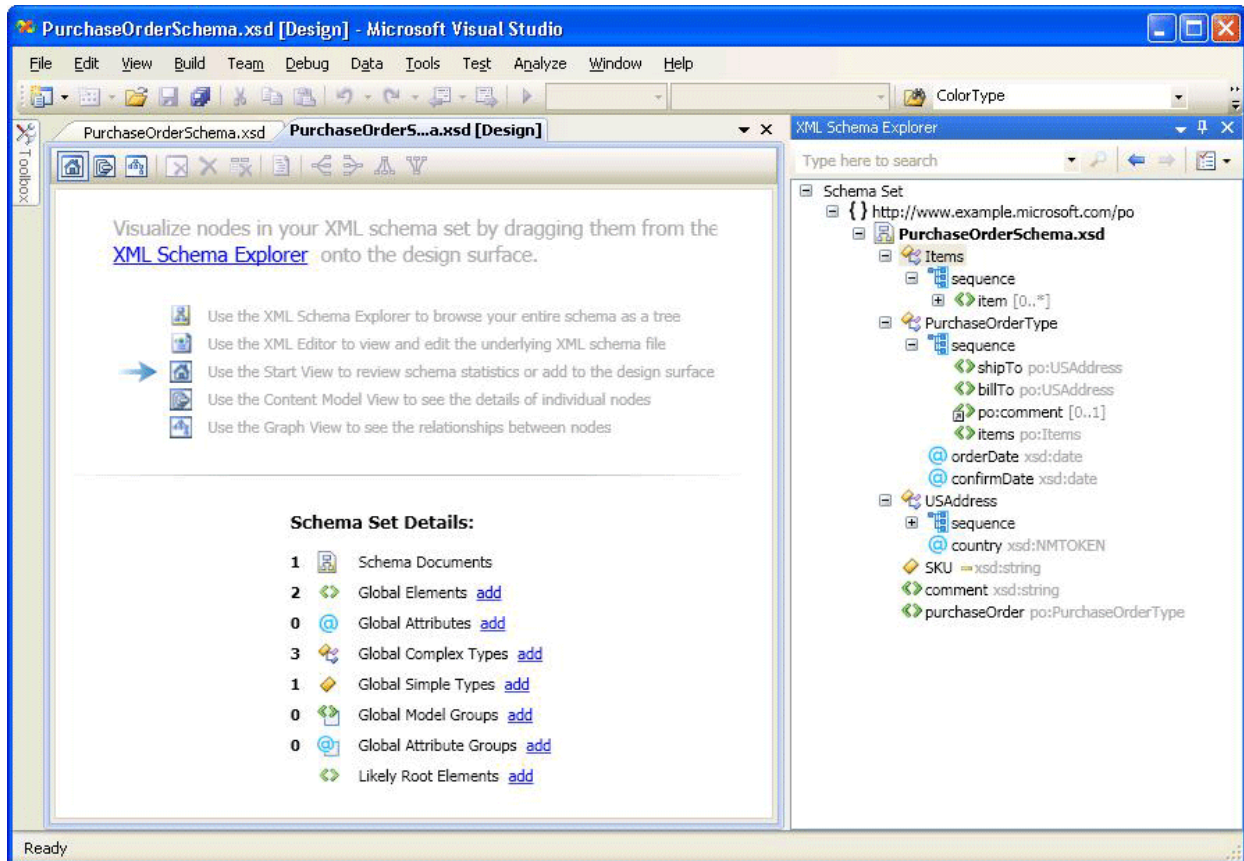


# O modo de Início

14/05/2021 • 2 minutes to read

O modo de Início é o ponto iniciando para o designer de (XSD) de esquema XML. Quando você cria um novo arquivo XSD, você verá inicialmente modo de Início.

O modo de exibição iniciar contém duas seções principais, a *marca-d' água* e o painel de **detalhes do conjunto de esquemas**. Também inclui a barra de ferramentas, que está disponível em todas as exibições de designer XSD.



## Marca de água

O painel de marca d' água contém uma lista de links para todas as exibições do designer XSD, editor de XML e Gerenciador de esquema XML. Se o esquema tem erros, o seguinte texto é exibido no fim da lista: "Use Lista de erros para exibir e corrigir erros no conjunto."

## Detalhes do conjunto de esquemas

O painel de **detalhes do conjunto de esquemas** lista os tipos de nó de esquema global e exibe quantas instâncias de cada tipo existem no esquema. Você pode usar os links **Adicionar** ao lado dos tipos de nó para adicionar novos nós ao espaço de trabalho.

## Barra de ferramentas

Você pode navegar entre a exibição de início, a [exibição de modelo de conteúdo](#) e a exibição de [gráfico](#) da barra de ferramentas do designer de esquema XML.



Os seguintes botões estão ativados na barra de ferramentas do designer XSD quando a exibição de Início é ativa:

OPÇÃO	DESCRIÇÃO
Mostrar Exibição Inicial	Alterna a O modo de Início. Este modo de exibição pode ser acessado usando o atalho de teclado: <b>Ctrl + 1</b> .
Mostrar Exibição de Modelo de Conteúdo	Alterna a O modo do modelo de conteúdo. Este modo de exibição pode ser acessado usando o atalho de teclado: <b>Ctrl + 2</b> .
Mostrar Exibição de Gráfico	Alterna a O modo de gráfico. Este modo de exibição pode ser acessado usando o atalho de teclado: <b>Ctrl + 3</b> .

## Confira também

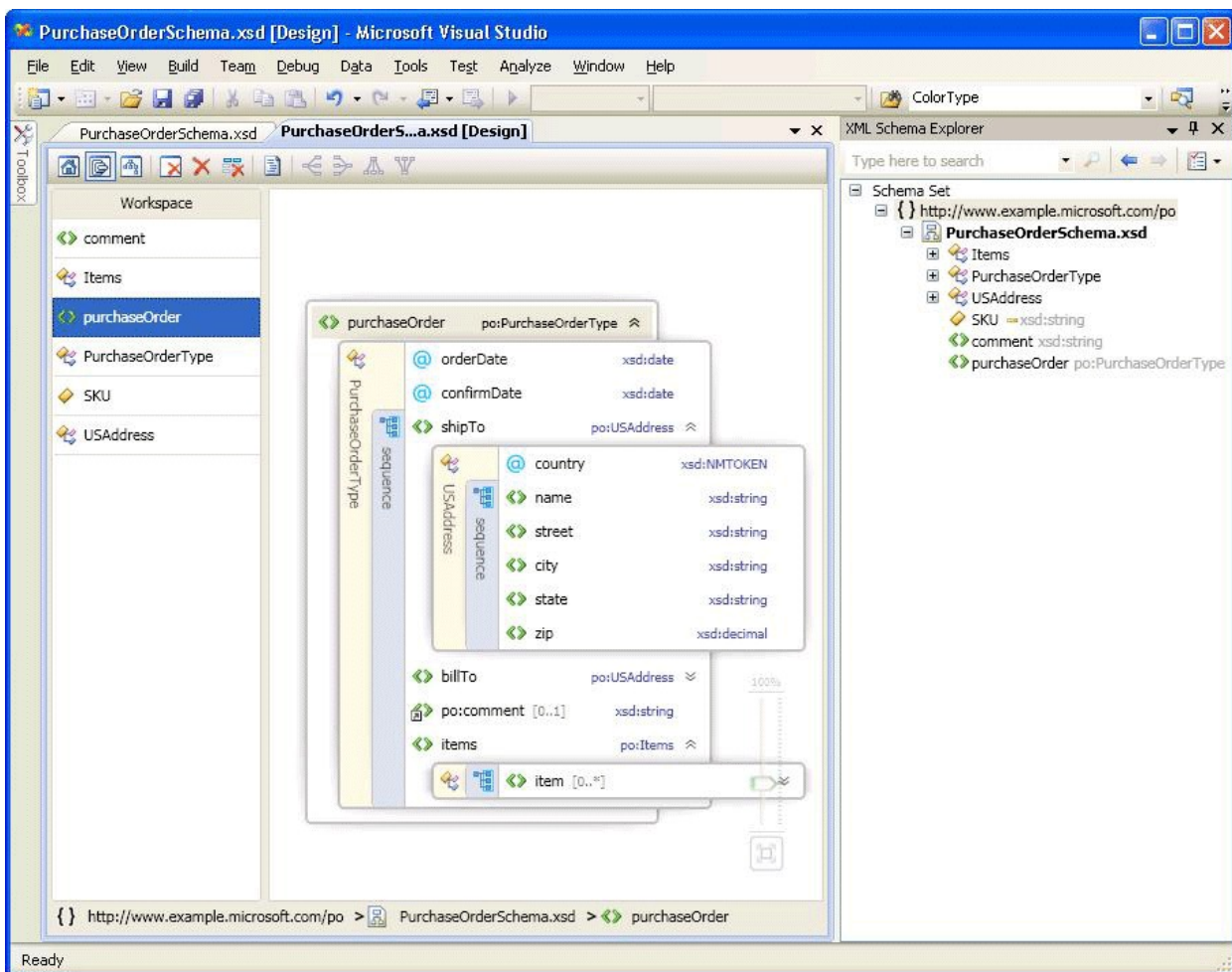
- [XML Schema Explorer](#)
- [Exibição de gráfico](#)
- [O modo do modelo de conteúdo](#)
- [Editor de XML](#)

# O modo do modelo de conteúdo

14/05/2021 • 6 minutes to read

A exibição do modelo de conteúdo fornece uma representação gráfica de nós locais e globais de esquema e seus componentes, de incluir simples e tipos complexos, de elementos, grupos de modelo, de atributos, e de grupos de atributo. Comentários e instruções de processamento XML não podem ser exibidos no modo do modelo de conteúdo. A exibição do modelo de conteúdo contém dois painéis: um painel do **espaço de trabalho** que contém uma lista dos nós no espaço de trabalho do [Designer de esquema XML](#) e a superfície de design onde você pode ver o modelo de conteúdo dos nós de esquema selecionados no painel do **espaço de trabalho**. A exibição do modelo de conteúdo também inclui a barra de ferramentas do designer de esquema XML e a barra de rastreamento.

Na imagem a seguir, o painel de **espaço de trabalho** contém seis nós de esquema. O `purchaseOrder` nó é selecionado no painel de **espaço de trabalho** e é exibido na superfície de design.



## Painel do espaço de trabalho

Depois de adicionar nós ao espaço de trabalho, a lista de nós aparecerá no painel do **espaço de trabalho** do modo de exibição de modelo de conteúdo. Quando você seleciona nós no painel do **espaço de trabalho**, eles aparecem na superfície de design da exibição do modelo de conteúdo. Para excluir nós do espaço de trabalho, use a barra de ferramentas do designer XSD, o menu de **espaço de trabalho**, clique com o botão direito do mouse ou a tecla **delete**.

Para obter informações sobre como adicionar nós, consulte a seção "adicionando nós ao espaço de trabalho" no

## Superfície de design

Quando um nó é selecionado no painel do **espaço de trabalho**, ele é adicionado à superfície de design da exibição do modelo de conteúdo, na qual você pode exibir os detalhes do nó.

O modelo de conteúdo de um nó é representado por uma árvore gráfico expansível com elementos e atributos que aparecem como nós de árvore. Por padrão, somente um nível é expandido. Outras informações, como compositores, nomes de tipo, grupos, e outros contêineres é colocada em uma barra vertical (quando expandido) ao longo de elementos e atributos que incluem. Quando você clica duas vezes em uma barra vertical, transformações horizontal e recolhe de árvore. Quando você clica duas vezes em uma barra horizontal e vertical, transformações a árvore expande. Selecionar a barra vertical seleciona todos os nós no contêiner. Os expansores aparecem à direita de um nó se um elemento pode ser expandido ou recolhido.

Se a superfície de design estiver em branco, o editor de XML, o **XML Schema Explorer** e a marca d' água serão mostrados. A *marca d' água* é uma lista de links para todas as exibições do designer XSD. Se o esquema tem erros, o seguinte texto é exibido no fim da lista: "Use Lista de erros para exibir e corrigir erros no conjunto."

## Barra de navegação estrutural

A barra de rastreamento na parte inferior do modo de modelo de conteúdo mostra onde o nó selecionado é localizado no conjunto de esquema.

## Menus de contexto

Quando você clica com o botão direito do mouse em um item na superfície de design ou no painel do **espaço de trabalho**, um menu de contexto é exibido. A tabela a seguir descreve as opções que estão disponíveis para a superfície de design do modo de modelo de conteúdo.

OPÇÃO	DESCRIÇÃO
Apresentação em XML Schema Explorer	Coloca o foco no esquema Explorer e ressalta o nó do esquema.
Apresentação no modo de gráfico	Alterna a O modo de gráfico.
Gerencia o exemplo XML	Disponível somente para os elementos globais. Gerencia um arquivo XML de exemplo para o elemento global.
Documentação de apresentação	Mostra ou de anotação/documentação de oculta conteúdo do nó.
Exportar diagrama como imagem	Salva a superfície de design para um arquivo XPS.
Exibir Código	Abre o arquivo que contém o nó selecionado no editor de XML. O item selecionado no <b>XML Schema Explorer</b> também é selecionado no editor de XML.
Janela Propriedades	Abre a janela <b>Propriedades</b> (se ainda não estiver aberta). Esta janela exibe informações sobre o nó.

A tabela a seguir descreve as opções que estão disponíveis para o painel do **espaço de trabalho**.

OPÇÃO	DESCRIÇÃO
<b>Apresentação em XML Schema Explorer</b>	Coloca o foco no esquema Explorer e resalta o nó do esquema.
<b>Apresentação no modo de gráfico</b>	Alterna para representar graficamente a exibição.
<b>O workspace claro</b>	Limpa o workspace e a superfície de design.
<b>Remova de workspace</b>	Remove selecionou nós de workspace e da superfície de design.
<b>Remova todos com exceção de seleção de workspace</b>	Remove os nós que não são selecionados de workspace e da superfície de design.
<b>Gerencia o exemplo XML</b>	Disponível somente para os elementos globais. Gerencia um arquivo XML de exemplo para o elemento global.
<b>Selecionar tudo</b>	Seleciona todos os nós no painel do <b>espaço de trabalho</b> .
<b>Exibir Código</b>	Abre o arquivo que contém o nó selecionado no editor de XML. O item selecionado no <b>XML Schema Explorer</b> também é selecionado no editor de XML.
<b>Janela Propriedades</b>	Abre a janela <b>Propriedades</b> (se ainda não estiver aberta). Esta janela exibe informações sobre o nó.

## Janela de Propriedades

Use o menu de clique com o botão direito do mouse (contexto) para abrir inicialmente a janela **Propriedades**. Por padrão, a janela **Propriedades** é exibida no canto inferior direito do Visual Studio. Quando você clica em um nó que é processado no modo de exibição de modelo de conteúdo, as propriedades desse nó são exibidas na janela **Propriedades**.

## Barra de ferramentas do designer XSD

Os seguintes botões da barra de ferramentas do designer XSD são ativados quando a exibição do modelo de conteúdo está ativo.



OPÇÃO	DESCRIÇÃO
<b>Mostrar Exibição Inicial</b>	Alterna para o <a href="#">modo de exibição de início</a> . Este modo de exibição pode ser acessado usando o atalho de teclado: <b>Ctrl + 1</b> .
<b>Mostrar Exibição de Modelo de Conteúdo</b>	Alterna para a <a href="#">exibição do modelo de conteúdo</a> . Este modo de exibição pode ser acessado usando o atalho de teclado: <b>Ctrl + 2</b> .
<b>Mostrar Exibição de Gráfico</b>	Alterna para o <a href="#">modo de exibição de gráfico</a> . Este modo de exibição pode ser acessado usando o atalho de teclado: <b>Ctrl + 3</b> .

OPÇÃO	DESCRIÇÃO
O workspace claro	Limpa o workspace e a superfície de design.
Remova de workspace	Remove selecionou nós de workspace e da superfície de design.
Remova todos com exceção de seleção de workspace	Remove os nós que não são selecionados de workspace e da superfície de design.
Documentação de apresentação	Mostra ou de anotação/documentação de oculta conteúdo do nó.

## Bandeja/rolagem

Você pode deslocar a superfície de design usando as barras de rolagem ou mantendo a tecla **Ctrl** pressionada enquanto clica e arrasta o mouse. Quando você panorâmica a superfície de design usando clicar e arrastar, o cursor muda para quatro setas cruzadas apontando para quatro direções.

## Desfazer/refazer

Desfazer/refaz o recurso é habilitado no modo de modelo de conteúdo para as seguintes ações:

- Adicionando um único nó arrastando e soltando-se.
- Adicionando mais nós da janela de resultados de pesquisa no esquema Explorer.
- Adicionando a exibição de nós do início.
- Excluindo única ou mais nós.

## Zoom

O zoom está disponível no canto inferior direito da exibição do modelo de conteúdo.

O zoom pode ser controlado das seguintes maneiras:

- Segurando a tecla **Ctrl** e girando a roda do mouse quando o mouse estiver passando pela superfície de exibição do modelo de conteúdo.
- Usando o controle deslizante. O controle deslizante mostra o nível atual de zoom.

O controle deslizante de zoom é opaco quando você o seleciona, passa o mouse sobre ele ou usa **Ctrl** com a roda do rato para aplicar zoom; em outras ocasiões, ele é transparente.

## Integração do editor de XML

Você pode alternar entre o **Designer XSD** e o editor de XML usando o menu de clique com o botão direito do mouse (contexto).

Se você fizer alterações no conjunto de esquema no editor de XML, as alterações serão sincronizadas na exibição do modelo de conteúdo. Para obter mais informações, consulte [integração com o editor de XML](#).

## Consulte também

- [Espaço de trabalho do designer de esquema XML](#)

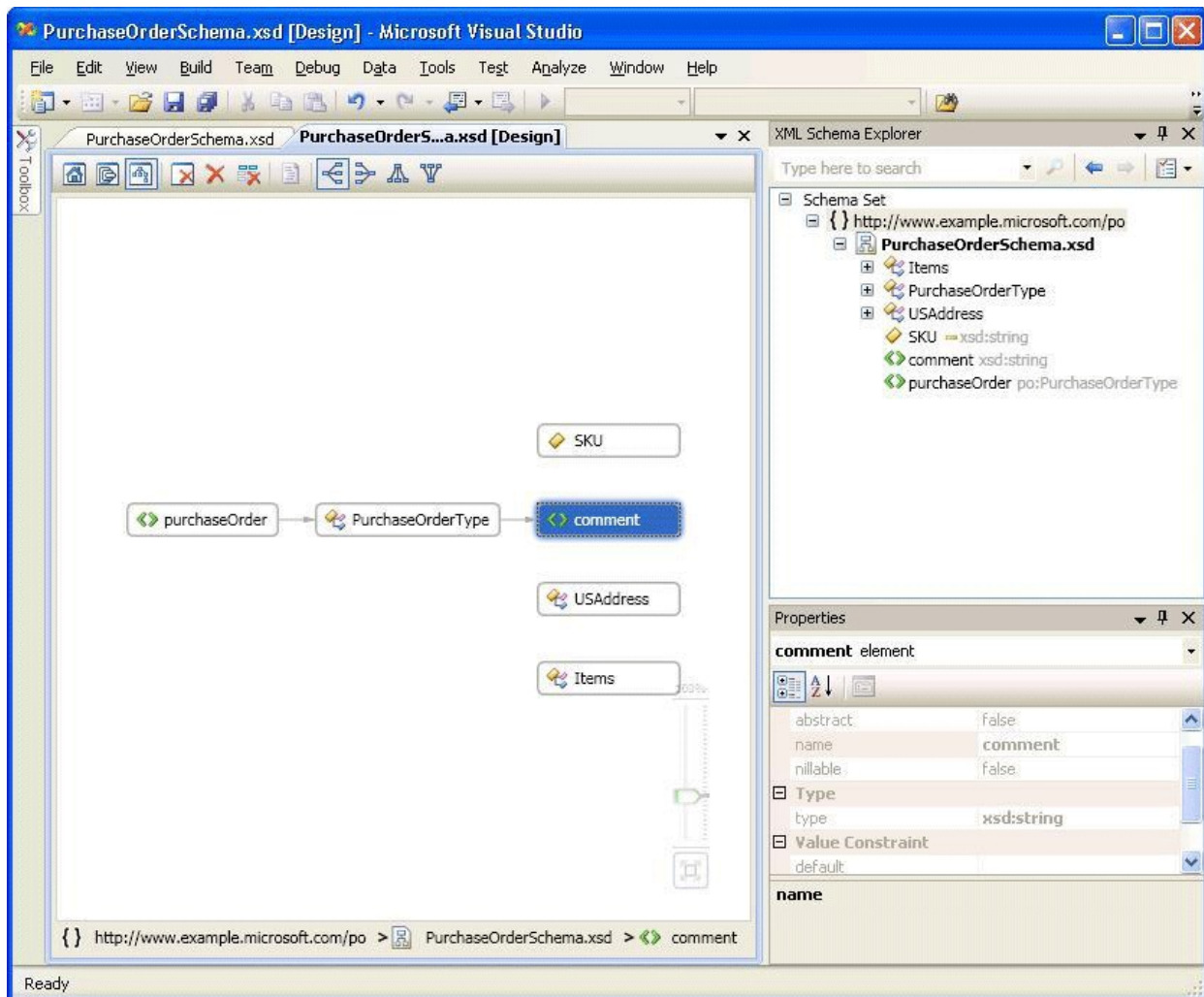


# Exibição de gráfico

14/05/2021 • 7 minutes to read

A exibição do gráfico fornece uma representação gráfica de nós globais do esquema e relações entre os nós. Observe que a exibição do gráfico não permite que você altere o layout do esquema definido na superfície de design. A exibição do gráfico também inclui a barra de ferramentas do designer de esquema XML e a barra de rastreamento.

A imagem a seguir mostra a visualização de gráfico com seis nós globais na superfície de design.



## Superfície de design






A superfície de design da exibição de gráfico exibe o conteúdo do [espaço de trabalho do designer de esquema XML](#). Se o workspace contém quaisquer nós globais do conjunto de esquema, os nós são mostrados na superfície de design do modo de gráfico e as setas são desenhadas entre os nós que possuem relações.

Clicar duas vezes em um nó no modo de exibição de gráfico abrirá o editor de XML.

Para excluir os nós selecionados do espaço de trabalho, use a barra de ferramentas do XSD designer ou a tecla **delete**.

Se a superfície de design estiver em branco, o editor de XML, o **XML Schema Explorer** e a *marca d'água* serão mostrados. A *marca d'água* é uma lista de links para todas as exibições do designer XSD.

Visualize nodes in your XML schema set by dragging them from the **XML Schema Explorer** onto the design surface.

-  Use the XML Schema Explorer to browse your entire schema as a tree
-  Use the XML Editor to view and edit the underlying XML schema file
-  Use the Start View to review schema statistics or add to the design surface
-  Use the Content Model View to see the details of individual nodes
-  Use the Graph View to see the relationships between nodes

Se o esquema tem erros, o seguinte texto é exibido no fim da lista: "Use Lista de erros para exibir e corrigir erros no conjunto."

## Barra de navegação estrutural

A barra de rastreamento na parte inferior do modo de figura a seguir mostra onde o nó selecionado é localizado no conjunto de esquema. Se vários itens são selecionados, a barra de rastreamento será em branco.

## Menu de contexto (clique com o botão direito do mouse)

A tabela a seguir descreve as opções que estão disponíveis para todos os nós na superfície de design do modo de gráfico.

OPÇÃO	DESCRIÇÃO
Apresentação em XML Schema Explorer	Coloca o foco no esquema Explorer e resalta o nó do esquema.
Apresentação no modo de gráfico	Alterna para o modo de exibição gráfico (desativada).
Gerencia o exemplo XML	Disponível somente para os elementos globais. Gerencia um arquivo XML de exemplo para o elemento global.
O workspace claro	Limpa o workspace e a superfície de design.
Remova de workspace	Remove selecionou nós de workspace e da superfície de design.
Remova todos com exceção de seleção de workspace	Remove os nós que não são selecionados de workspace e da superfície de design.
Exportar diagrama como imagem	Salva a superfície de design para um arquivo XPS.
Selecionar tudo	Selecionar todos os nós na superfície de design.
Exibir Código	Abre o arquivo que contém o nó selecionado no editor de XML. O item selecionado no <b>XML Schema Explorer</b> também é selecionado no editor de XML.
Janela Propriedades	Abre a janela <b>Propriedades</b> (se ainda não estiver aberta). Esta janela exibe informações sobre o nó.

Além das opções comuns descritas anterior, o menu de contexto para elementos globais também tem as seguintes opções:

OPÇÃO	DESCRIÇÃO
Adicionar Definição de Tipo	Adiciona o tipo base no diagrama.



OPÇÃO	DESCRIÇÃO
<b>Adicionar Todas as Referências</b>	Adiciona todos os nós que se referem ao elemento e desenha setas para indicar relações entre eles.
<b>Adicionar Membros do Grupo de Substituição</b>	Adiciona todos os membros do grupo de substituição. Essa opção aparece na exibição se o elemento é o início ou membro de um grupo de substituição.
<b>Gerencia o exemplo XML</b>	Gerencia um arquivo XML de exemplo para o elemento global.

Além das opções comuns descritas anterior, o menu de contexto para tipos complexos simples e globais globais também tem as seguintes opções:

OPÇÃO	DESCRIÇÃO
<b>Adicione o tipo base</b>	Se o tipo selecionado é derivado de um tipo global, adiciona o tipo base do tipo selecionado.
<b>Adicionar Todas as Referências</b>	Adiciona todas as referências de tipo selecionado. Isso inclui os elementos e atributos do tipo selecionado, e tipos derivados do tipo selecionado.
<b>Adicionar Todos os Tipos Derivados</b>	Adiciona todos os tipos que direta e indiretamente são derivados do tipo selecionado.
<b>Adicionar Todos os Ancestrais</b>	Adiciona todos os tipos de base pai ().

Além das opções comuns descritas anterior, o menu de contexto para grupos globais e grupos de atributo também tem as seguintes opções:

OPÇÃO	DESCRIÇÃO
<b>Adicionar Todas as Referências</b>	Adiciona todos os nós que se referem ao grupo e desenha setas para indicar relações entre eles.
<b>Adicionar Todos os Membros</b>	Adiciona todos os membros do grupo e desenha setas para indicar relações entre eles.

No addition em padrões comuns descritas anterior, o menu de contexto para atributos globais também tem as seguintes opções:

OPÇÃO	DESCRIÇÃO
<b>Adicionar Todas as Referências</b>	Adiciona todos os nós que se referem ao grupo e desenha setas para indicar relações entre eles.

## Janela de Propriedades

Use o menu de contexto (clique com o botão direito do mouse) para abrir inicialmente a janela **Propriedades** . Por padrão, a janela **Propriedades** é exibida no canto inferior direito do Visual Studio. Quando você clica em um nó que é processado no modo de exibição de modelo de conteúdo, as propriedades desse nó serão exibidas na janela **Propriedades** .

## Barra de ferramentas XSD

Os seguintes botões da barra de ferramentas XSD são ativados quando a exibição do gráfico está ativo.



OPÇÃO	DESCRIÇÃO
Mostrar Exibição Inicial	Alterna para o <a href="#">modo de exibição de início</a> . Este modo de exibição pode ser acessado usando o atalho de teclado: <b>Ctrl + 1</b> .
Mostrar Exibição de Modelo de Conteúdo	Alterna para a <a href="#">exibição do modelo de conteúdo</a> . Este modo de exibição pode ser acessado usando o atalho de teclado: <b>Ctrl + 2</b> .
Mostrar Exibição de Gráfico	Alterna para o <a href="#">modo de exibição de gráfico</a> . Este modo de exibição pode ser acessado usando o atalho de teclado: <b>Ctrl + 3</b> .
O workspace claro	Limpa o workspace e a superfície de design.
Remova de workspace	Remove selecionou nós de workspace e da superfície de design.
Remova todos com exceção de seleção de workspace	Remove os nós que não são selecionados de workspace e da superfície de design. Essa opção é ativada no modo do modelo de conteúdo e no modo de gráfico.
Da esquerda para a direita	Altera o layout no modo de gráfico a uma representação hierárquica esquerda para a direita de nós. Essa opção pode ser acessada usando o atalho de teclado: <b>+ seta para a direita Alt</b> .
Da direita para a esquerda	Altera o layout no modo de gráfico a uma representação hierárquica da direita para a esquerda de nós. Essa opção pode ser acessada usando o atalho de teclado: <b>ALT + seta para a esquerda</b> .
De Cima para Baixo	Altera o layout no modo de gráfico a uma representação hierárquica de cima para baixo de nós. Essa opção pode ser acessada usando o atalho de teclado: <b>ALT + seta para baixo</b> .
De baixo para cima	Altera o layout no modo de gráfico a uma representação hierárquica de parte inferior-à- parte superior dos nós. Essa opção pode ser acessada usando o atalho de teclado: <b>+ seta para cima Alt</b> .

## Bandeja/rolagem

Você pode deslocar a superfície de design usando as barras de rolagem ou mantendo a tecla **Ctrl** pressionada enquanto clica e arrasta o mouse. Quando você filtra a superfície de design usando o clique e o arrastar, o cursor será alterado a quatro setas cruzadas apontando em quatro direções.

## Desfazer/refazer

Desfazer/refaz o recurso é habilitado no modo de gráfico para as seguintes ações:

- Adicionando um único nó arrastando e soltando-se.
- Adicionando mais nós da janela de resultados de pesquisa no esquema Explorer ou em consultas de exibição de Início.
- Excluindo única ou mais nós.

## Zoom

O zoom está disponível no canto inferior direito do modo de gráfico.

O zoom pode ser controlado das seguintes maneiras:

- Segurando a tecla **Ctrl** e girando a roda do mouse quando o mouse estiver passando pela superfície de exibição do gráfico.
- Usando o controle deslizante. O controle deslizante mostra o nível atual de zoom.

O controle deslizante de zoom é opaco quando você o seleciona, passa o mouse sobre ele ou usa **Ctrl** com a roda do rato para aplicar zoom; em outras ocasiões, ele é transparente.

## Integração do editor de XML

Você pode alternar entre o modo de exibição de gráfico e o editor de XML clicando em um nó e usando o menu de contexto do código de exibição (clique com o botão direito do mouse).

Se você fizer alterações no conjunto de esquema no editor de XML, as alterações serão sincronizadas no modo de exibição de gráfico. Para obter mais informações, consulte [integração com o editor de XML](#).

## Confira também

- [Design Surface](#)

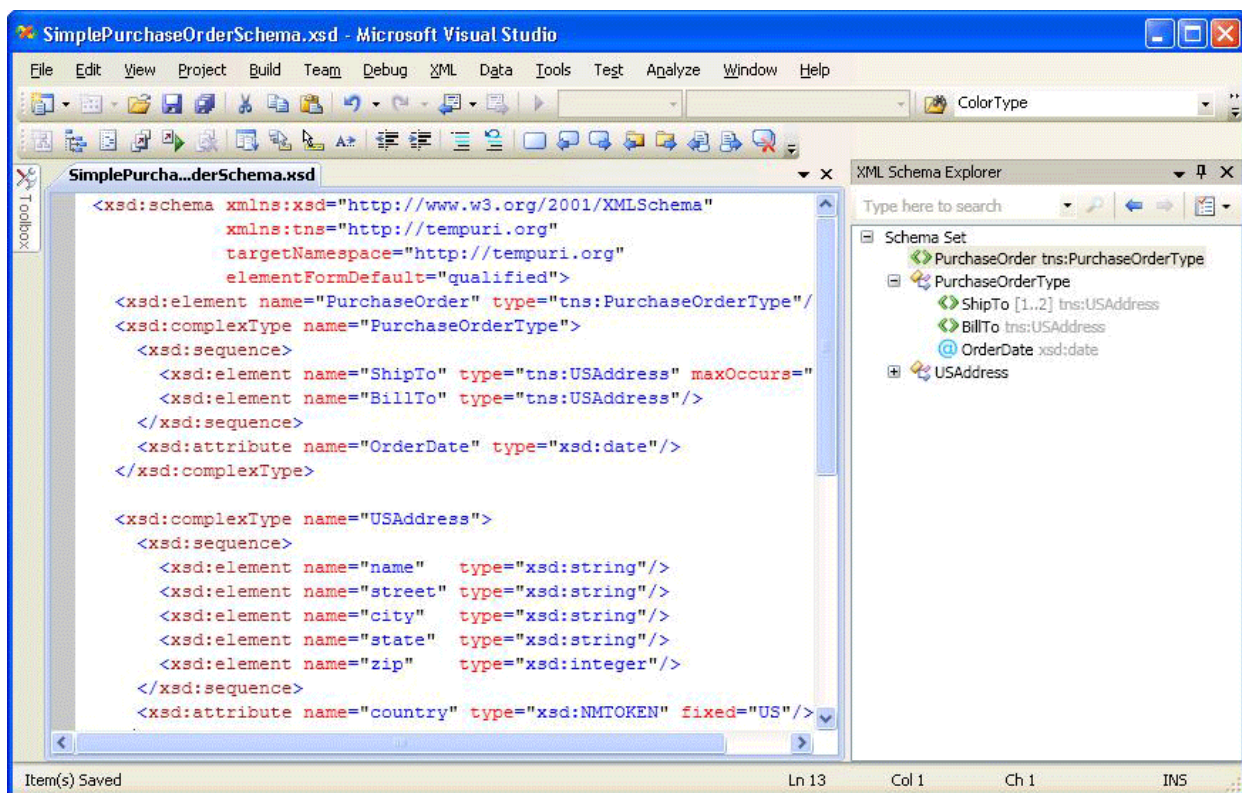
# Integração com editor XML

14/05/2021 • 2 minutes to read

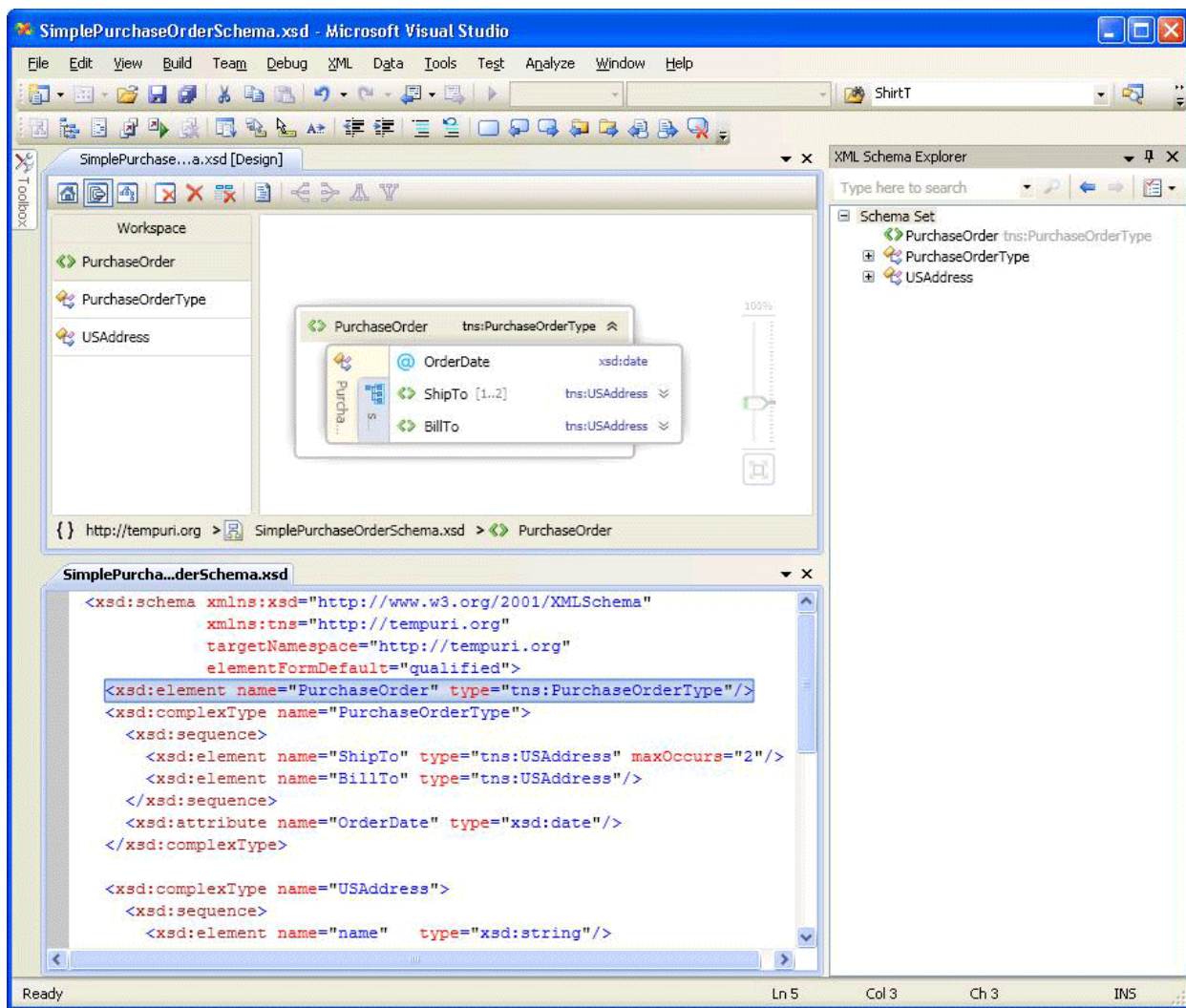
O designer de esquema XML é integrado ao editor de XML. Se você modificar um arquivo XSD no editor de XML, a alteração será refletida no [XML Schema Explorer](#). Se você tiver a [exibição de gráfico](#) ou a [exibição de modelo de conteúdo](#) aberta, a alteração também será refletida ali. Você pode navegar entre o designer de esquema XML e o editor de XML das seguintes maneiras:

- No editor de XML, clique com o botão direito do mouse em um nó e selecione **Mostrar no Gerenciador de esquema XML**.
- Na exibição de gráfico e no **Gerenciador de esquema XML**, clique duas vezes em um nó ou clique com o botão direito do mouse em um nó e selecione **Exibir código**. Na exibição modelo de conteúdo, clique com o botão direito do mouse em um nó e selecione **Exibir código**.

A captura de tela a seguir mostra um esquema XML aberto no **XML Schema Explorer**. O **XML Schema Explorer** exibe o conjunto de esquema em um modo de exibição de árvore. O editor de XML exibe a exibição de texto do nó que está atualmente ativo no **XML Schema Explorer**.



Às vezes, é útil ver o código no editor de XML e o designer gráfico lado a lado. Para exibir os dois arquivos ao mesmo tempo, clique com o botão direito do mouse em qualquer lugar no editor de XML e selecione **Exibir Designer**. No menu do Windows do Visual Studio, selecione **novo grupo de guias horizontal (ou vertical)**.



Confira também

- [XML Schema Explorer](#)

# Como: criar e editar um arquivo de esquema XSD

14/05/2021 • 2 minutes to read

Este tópico descreve como criar um novo arquivo de esquema XML (XSD) e, em seguida, adicionar conteúdo ao arquivo XSD.

## Para criar um novo arquivo de esquema XML

1. No Visual Studio, abra o menu **arquivo** e selecione **novo > arquivo**. Ou use o atalho de teclado **Ctrl + N**.
2. Na caixa de diálogo **novo arquivo**, selecione **esquema XML** e, em seguida, selecione **abrir**.

Um novo arquivo é criado. A exibição inicial e uma janela vazia do **Gerenciador de esquema XML** aparecem no Visual Studio.

## Para editar um arquivo

1. Selecione o **Editor de XML de uso para exibir e editar o link do arquivo de esquema XML subjacente** na exibição inicial.
2. Copie o código de exemplo do esquema XML do [esquema da ordem de compra](#) e cole-o para substituir o código que foi adicionado ao novo arquivo XSD por padrão.

O **XML Schema Explorer** é atualizado com o novo conjunto de esquema.

# Como: criar um documento XML com base em um esquema XSD

14/05/2021 • 2 minutes to read

O recurso **gerar XML de exemplo** gera um arquivo XML de exemplo baseado em seu arquivo de esquema XML (XSD).

Você pode usar esta opção para os seguintes situações:

- Para entender o uso de várias construções no seu esquema.
- Para confirmar que o esquema faz o que é esperado dele.

O recurso **gerar XML de exemplo** só está disponível em elementos globais e requer um conjunto de esquema XML válido.

Esse recurso normalmente gera documentos XML válidos. No entanto, se o esquema contiver um ou mais dos seguintes, o exemplo poderá não ser válido:

- As restrições de identidade `xs:key`, `xs:keyref` e `xs:unique`.
- `xs:pattern` facetas.
- Enumerações do tipo `xs:QName`.
- `xs:ENTITY` e `xs:ENTITIES` tipos, e `xs:NOTATION`.

Além disso, observe que o conteúdo de `xs:base64Binary` será gerado apenas se as enumerações ocorrerem no esquema para esse tipo.

## Para gerar um documento de instância XML baseado no arquivo XSD

1. Siga as etapas em [como criar e editar um arquivo de esquema XSD](#).
2. No [Gerenciador de esquema XML](#), clique com o botão direito do mouse no `PurchaseOrder` elemento global e selecione **gerar XML de exemplo**.

Quando você seleciona essa opção, o PurchaseOrder. o arquivo XML com o seguinte conteúdo XML de exemplo será gerado e aberto no editor de XML:

```
<?xml version="1.0" encoding="utf-8"?>
<PurchaseOrder OrderDate="1900-01-01" xmlns="http://tempuri.org/PurchaseOrderSchema.xsd">
  <ShipTo country="US">
    <name>name1</name>
    <street>street1</street>
    <city>city1</city>
    <state>state1</state>
    <zip>1</zip>
  </ShipTo>
  <ShipTo country="US">
    <name>name2</name>
    <street>street2</street>
    <city>city2</city>
    <state>state2</state>
    <zip>-79228162514264337593543950335</zip>
  </ShipTo>
  <BillTo country="US">
    <name>name1</name>
    <street>street1</street>
    <city>city1</city>
    <state>state1</state>
    <zip>1</zip>
  </BillTo>
</PurchaseOrder>
```



# Como adicionar nós de resultado de pesquisa de conjunto de esquema ao espaço de trabalho

14/05/2021 • 2 minutes to read

Este tópico explica como adicionar nós que são realçados no **XML Schema Explorer** como o resultado de uma pesquisa de palavra-chave no espaço de trabalho.

## NOTE

Somente nós globais podem ser adicionados ao [espaço de trabalho](#).

Este exemplo usa o [esquema de ordem de compra](#) de exemplo.

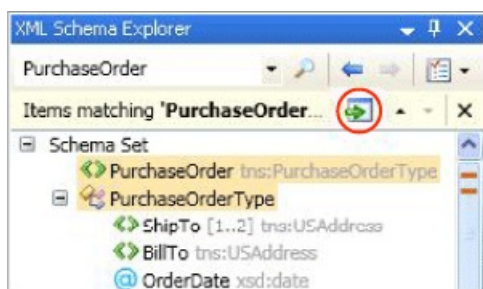
## Para adicionar nós definidos do resultado de esquema

1. Siga as etapas em [como criar e editar um arquivo de esquema XSD](#).
2. Digite "purchaseOrder" na caixa de texto Pesquisar da barra de ferramentas do [Gerenciador de XML](#) e clique no botão Pesquisar.



Os resultados da pesquisa são realçados no **XML Schema Explorer** e marcados por tiques na barra de rolagem vertical.

3. Adicione os resultados da pesquisa ao espaço de trabalho clicando no botão **adicionar nós realçados ao espaço de trabalho** no painel resultados do resumo.



O `purchaseOrder` nó e o `PurchaseOrderType` nó aparecem ao lado um do outro na superfície de design do [modo de exibição de gráfico](#). Porque os dois nós são relacionados (o elemento de `purchaseOrder` é do tipo de `PurchaseOrderType`), uma seta é desenhada entre eles.

# Como alternar entre exibições e o editor de XML

14/05/2021 • 2 minutes to read

Este tópico mostra como alternar entre as exibições do designer de esquema XML (XSD designer) e o editor de XML. Este exemplo usa o [esquema de ordem de compra](#).

## Para alternar entre as exibições e o editor de XML

1. Para criar e editar um novo arquivo de esquema XML, siga as etapas em [como: criar e editar um arquivo de esquema XSD](#).
2. Para alternar para o designer de esquema XML do editor de XML, clique com o botão direito do mouse em qualquer lugar do editor de XML e selecione **Designer de exibição**.
3. Para alternar para o modo de exibição de gráfico usando a marca d' água, clique na **exibição usar o gráfico para ver a relação entre o link nós** na exibição iniciar.
4. Arraste o `USAddress` nó do **XML Schema Explorer** para a exibição de gráfico. Clique com o botão direito do mouse no `USAddress` nó no modo de exibição de gráfico e selecione **Mostrar no modo de exibição de modelo de conteúdo** no menu de contexto.

A exibição do modelo de conteúdo com os detalhes de nó de `USAddress` aparece.

5. Para alternar para o modo de exibição iniciar do modo de exibição de modelo de conteúdo usando a barra de ferramentas, clique no botão **Iniciar exibição** na barra de ferramentas XSD.
6. Para alternar entre as exibições usando as teclas de impressão, pressione **Ctrl + 1** para a exibição de início, **Ctrl + 2** para a exibição de gráfico e **Ctrl + 3** para a exibição do modelo de conteúdo.
7. Para ir para o editor de XML na exibição do modelo de conteúdo, clique com o botão direito do mouse no nó e selecione **Exibir código** no menu de contexto.

# Como adicionar nós ao espaço de trabalho do XML Schema Explorer

14/05/2021 • 2 minutes to read

Este tópico explica como adicionar nós ao espaço de [trabalho do designer de esquema XML](#) do **XML Schema Explorer**. Isso pode ser obtido arrastando e soltando nós do **XML Schema Explorer** em uma exibição de designer xsd ou usando o menu de contexto do **XML Schema Explorer**. Você também pode adicionar nós que são realçados como resultado de uma pesquisa realizada pelo **XML Schema Explorer**. Para obter mais informações, consulte [como adicionar nós de resultado de pesquisa de conjunto de esquema ao espaço de trabalho](#).

## NOTE

Somente nós globais podem ser adicionados ao [espaço de trabalho do designer de esquema XML](#).

## Para adicionar nós por meio do menu de contexto do Gerenciador de XML

1. Siga as etapas em [como criar e editar um arquivo de esquema XSD](#).
2. Clique com o botão direito do mouse no `PurchaseOrderType` nó no Gerenciador de XSD. Selecione **Mostrar no modo de exibição de gráfico**.

O nó de `purchaseOrderType` aparece na superfície de design do modo de gráfico.

## Para arrastar e soltar sobre um nó para uma visualização

1. Clique com o botão direito do mouse no `PurchaseOrderType` nó no modo de exibição de gráfico. Selecione **Mostrar no Gerenciador de esquema XML**.
- O nó é realçado no **XML Schema Explorer**.
2. Clique com o botão direito do mouse no `PurchaseOrderType` nó no **XML Schema Explorer** e selecione **Mostrar todas as referências**.

O nó de `purchaseOrder` é realçado.

3. Arraste o nó de `purchaseOrder` sobre para o modo de exibição gráfico.

O nó de `purchaseOrder` e o nó de `PurchaseOrderType` aparecem próximos uns dos outros na superfície de design do modo de gráfico. Porque os dois nós são relacionados (o elemento de `purchaseOrder` é do tipo de `PurchaseOrderType`), uma seta é desenhada entre eles.

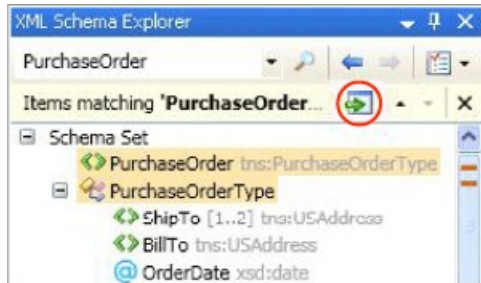
## Para adicionar nós que usam o esquema Explorer pesquisam o recurso

1. Digite "purchaseOrder" na caixa de texto Pesquisar da barra de ferramentas do [Gerenciador de XML](#) e clique no botão Pesquisar.



Os resultados da pesquisa são realçados no **XML Schema Explorer** e marcados por tiques na barra de rolagem vertical.

2. Adicione os resultados da pesquisa ao espaço de trabalho clicando no botão **adicionar nós realçados ao espaço de trabalho** no painel resultados do resumo.



O `purchaseOrder` nó e o `PurchaseOrderType` nó aparecem ao lado um do outro na superfície de design do [modo de exibição de gráfico](#). Porque os dois nós são relacionados (o elemento de `purchaseOrder` é do tipo de `PurchaseOrderType`), uma seta é desenhada entre eles.

## Confira também

- [XML Schema Explorer](#)

# Como adicionar nós ao espaço de trabalho a partir do modo de exibição iniciar

14/05/2021 • 2 minutes to read

Este tópico descreve como adicionar nós globais ao espaço de [trabalho do designer de esquema XML](#) do [modo de exibição iniciar](#).

## Para adicionar nós do início para o modo de workspace

1. Siga as etapas em [como criar e editar um arquivo de esquema XSD](#).
2. Clique com o botão direito do mouse no editor de XML e selecione **Designer de exibição**.

O modo de Início aparece.

Na seção **detalhes do conjunto de esquemas**, observe que há dois (2) **tipos complexos globais**.

3. Clique no link **Adicionar** à direita do título **tipos complexos globais**.

Os nós de tipo complexo de `USAddress` e de `PurchaseOrderType` aparecerão na superfície de design do modo de gráfico.

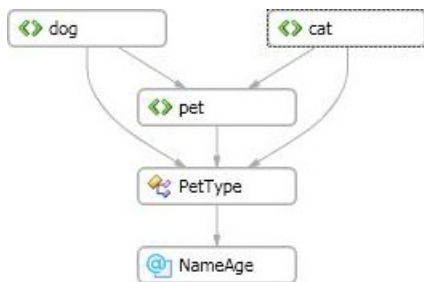
# Como: obter uma visão geral de um conjunto de esquema usando o modo de exibição de gráfico

14/05/2021 • 2 minutes to read

Este tópico descreve como usar o [modo de exibição de gráfico](#) para ver uma exibição de alto nível dos nós em um conjunto de esquema e as relações entre os nós.

## Para criar um novo arquivo XSD e exibir o elemento raiz no modo do modelo de conteúdo

1. Crie um novo arquivo de esquema XML e salve o arquivo como *Relationships.xsd*.
2. Clique no **Editor usar XML para exibir e editar o link do arquivo de esquema XML subjacente** na exibição inicial.
3. Copie o código de exemplo do esquema XML do [esquema XML de exemplo: relações](#) e cole-o para substituir o código que foi adicionado ao novo arquivo XSD por padrão.
4. Clique com o botão direito do mouse em qualquer lugar no editor de XML e selecione **Designer de exibição**.
5. Selecione o modo de exibição de gráfico na **barra de ferramentas XSD**.
6. Selecione o nó **conjunto de esquema** no **XML Schema Explorer** e arraste o nó para a superfície de design do modo de exibição de gráfico. Você deve ver todos os nós globais, e as setas que conectam os nós que possuem relações.



7. Clique em qualquer nó na superfície de design e examine a barra de rastreamento para ver onde o nó selecionado é localizado no conjunto de esquema.
8. Rick-clique em qualquer nó de elemento na superfície de design e selecione **gerar XML de exemplo** para ver o documento de instância XML.

# Como examinar o modelo de conteúdo de nós usando a exibição de modelo de conteúdo

14/05/2021 • 2 minutes to read

Este tópico descreve como explorar seus nós usando o modo de [exibição de modelo de conteúdo](#).

## Para criar um novo arquivo XSD e exibir o elemento raiz no modo do modelo de conteúdo

1. Crie um novo arquivo de esquema XML.
2. Clique em **usar editor de XML para exibir e editar o arquivo de esquema XML** subjacente na exibição inicial.
3. Copie o código de exemplo do esquema XML do [esquema XML de exemplo: ordem de compra esquema](#) e cole-o para substituir o código que foi adicionado ao novo arquivo XSD por padrão.
4. Selecione o `purchaseOrder` elemento no Gerenciador de esquema clicando com o botão direito do mouse no `purchaseOrder` elemento no editor de XML e selecionando **Mostrar no Gerenciador de XML**.
5. Clique com o botão direito do mouse no `purchaseOrder` no Gerenciador de XML e selecione **Mostrar no modo de exibição de modelo de conteúdo**.

A exibição do modelo de conteúdo exibe o elemento de `purchaseOrder` na superfície de design.

6. Expanda `shipTo`, `billTo`, e nós de `items` clicando duas vezes em cada nó ou double clicando na seta à direita de cada nó.

Os nós de elemento de `purchaseOrder` são expandidos e agora você pode ver o modelo de conteúdo do elemento.

7. Clique em qualquer nó no elemento de `purchaseOrder` e examine a barra de rastreamento para ver onde o esquema define o nó selecionado for encontrado.
8. Clique no botão **Mostrar documentação** na barra de ferramentas XSD para alternar a documentação. Você também pode clicar com o botão direito do mouse na superfície de design para ativar /desativar a documentação.
9. Clique com o botão direito do mouse no `purchaseOrder` nó e selecione **gerar XML de exemplo** para ver o documento de instância XML.

# Como: usar o designer de esquema XML com literais XML

14/05/2021 • 2 minutes to read

Este tópico descreve como exibir um esquema associado com um literal XML em um projeto Visual Basic.

## Criar um novo projeto de Visual Basic

1. Abra o Visual Studio.
2. Crie um novo projeto de **aplicativo de Console** Visual Basic chamado **xmlliterais**.

O novo projeto contém um arquivo de origem Visual Basic, *Module1.vb*.

## Adicionar um arquivo XSD existente

1. Abra um novo arquivo de texto no bloco de notas. Copie o código de exemplo do esquema XML do [esquema da ordem de compra](#) e cole-o no arquivo.
2. Salve o arquivo em algum local com o nome *PurchaseOrderSchema.xsd*.
3. Em **Gerenciador de soluções**, clique com o botão direito do mouse no nome do projeto, selecione **Adicionar** e, em seguida, selecione **Item existente**. A caixa de diálogo de **Item Addexisting** é exibida. Navegue até o arquivo *PurchaseOrderSchema.xsd*, selecione-o e clique em **Adicionar**.

O projeto xmlliterais agora contém dois arquivos: *Module1.vb* e *PurchaseOrderSchema.xsd*.

## Incluir código

Para adicionar Visual Basic código com um literal XML, com base no arquivo XSD incluído no projeto:

1. Substitua o código no arquivo *Module1.vb* pelo seguinte código:



```
Imports <xmlns:ns="http://tempuri.org/PurchaseOrderSchema.xsd">

Module Module1
    Sub Main()

        Dim XMLLiteral = <ns:PurchaseOrder OrderDate="1900-01-01">
            <ns:ShipTo country="US">
                <ns:name>name1</ns:name>
                <ns:street>street1</ns:street>
                <ns:city>city1</ns:city>
                <ns:state>state1</ns:state>
                <ns:zip>1</ns:zip>
            </ns:ShipTo>
            <ns:BillTo country="US">
                <ns:name>name1</ns:name>
                <ns:street>street1</ns:street>
                <ns:city>city1</ns:city>
                <ns:state>state1</ns:state>
                <ns:zip>1</ns:zip>
            </ns:BillTo>
        </ns:PurchaseOrder>

    End Sub
End Module
```

2. Clique com o botão direito do mouse em qualquer nó XML em um literal XML ou em uma importação de namespace XML e selecione **Mostrar no Gerenciador de esquema**.

O **XML Schema Explorer** é exibido lado a lado com um arquivo Visual Basic que tem o literal XML associado ao conjunto de esquema XML.

# Como: imprimir diagramas da exibição de gráfico e do modo de exibição de modelo de conteúdo

14/05/2021 • 2 minutes to read

Este tópico descreve como imprimir um diagrama da exibição de gráfico ou da exibição de modelo de conteúdo do designer de esquema XML.

## Para imprimir diagramas de designer de esquema XML

1. Abra um arquivo XSD no Visual Studio e adicione alguns nós ao [espaço de trabalho do designer de esquema XML](#).
2. Exporte o diagrama para um arquivo XPS usando o item de menu **Exportar diagrama como** contexto de imagem (clique com o botão direito do mouse) na superfície de design da exibição de gráfico ou no modo de exibição de modelo de conteúdo.

Quando você exporta um diagrama do modo de exibição de gráfico, toda a superfície de design é exportada para o arquivo XPS. Quando você exporta um diagrama do modo de exibição de modelo de conteúdo e mais de um nó aparece na superfície de design da exibição do modelo de conteúdo, somente o primeiro nó é exportado para o arquivo XPS.

3. Imprimir a imagem salvo no arquivo XPS usando um visualizador XPS.

## Confira também

- [Exibição de gráfico](#)
- [O modo do modelo de conteúdo](#)
- [Workspace de designer de esquema XML](#)

# Arquivo XSD de exemplo: esquema simples

14/05/2021 • 2 minutes to read

O arquivo XSD a seguir é usado em vários exemplos na documentação do Designer de Esquema XSD. Esse arquivo é um esquema de ordem de compra simples.

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://tempuri.org/PurchaseOrderSchema.xsd"
  targetNamespace="http://tempuri.org/PurchaseOrderSchema.xsd"
  elementFormDefault="qualified">
  <xsd:element name="PurchaseOrder" type="tns:PurchaseOrderType"/>
  <xsd:complexType name="PurchaseOrderType">
    <xsd:sequence>
      <xsd:element name="ShipTo" type="tns:USAddress" maxOccurs="2"/>
      <xsd:element name="BillTo" type="tns:USAddress"/>
    </xsd:sequence>
    <xsd:attribute name="OrderDate" type="xsd:date"/>
  </xsd:complexType>

  <xsd:complexType name="USAddress">
    <xsd:sequence>
      <xsd:element name="name" type="xsd:string"/>
      <xsd:element name="street" type="xsd:string"/>
      <xsd:element name="city" type="xsd:string"/>
      <xsd:element name="state" type="xsd:string"/>
      <xsd:element name="zip" type="xsd:integer"/>
    </xsd:sequence>
    <xsd:attribute name="country" type="xsd:NMTOKEN" fixed="US"/>
  </xsd:complexType>
</xsd:schema>
```

## NOTE

As empresas, as organizações, os produtos, os nomes de domínio, os endereços de email, os logotipos, as pessoas, os locais e os eventos de exemplo descritos aqui são fictícios. Nenhuma associação com nenhuma empresa, organização, produto, nome de domínio, endereço de email, logotipo, pessoa, locais ou eventos reais é intencional nem deve ser inferida.

# Arquivo XSD de exemplo: esquema de ordem de compra

14/05/2021 • 2 minutes to read

O arquivo XSD a seguir é usado em vários exemplos na documentação do Designer de Esquema XSD. Esse arquivo é um esquema de ordem de compra.

```

<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://tempuri.org/PurchaseOrderSchema.xsd"
  targetNamespace="http://tempuri.org/PurchaseOrderSchema.xsd"
  elementFormDefault="qualified">
  <xsd:element name='comment' type='xsd:string' />

  <xsd:element name='purchaseOrder' type='tns:PurchaseOrderType' />

  <xsd:complexType name='USAddress'>
    <xsd:annotation>
      <xsd:documentation>
        Purchase order schema for Example.Microsoft.com.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name='name' type='xsd:string' />
      <xsd:element name='street' type='xsd:string' />
      <xsd:element name='city' type='xsd:string' />
      <xsd:element name='state' type='xsd:string' />
      <xsd:element name='zip' type='xsd:decimal' />
    </xsd:sequence>
    <xsd:attribute name='country' type='xsd:NMTOKEN' fixed='US' />
  </xsd:complexType>

  <xsd:simpleType name='SKU'>
    <xsd:restriction base='xsd:string'>
      <xsd:pattern value='\d{3}\w{3}' />
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name='Items'>
    <xsd:sequence>
      <xsd:element name='item' minOccurs='0' maxOccurs='unbounded'>
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name='productName' type='xsd:string' />
            <xsd:element name='quantity'>
              <xsd:simpleType>
                <xsd:restriction base='xsd:positiveInteger'>
                  <xsd:minInclusive value='1' />
                  <xsd:maxExclusive value='100' />
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>
            <xsd:element name='USPrice' type='xsd:decimal' />
            <xsd:element ref='tns:comment' />
            <xsd:element name='shipDate' type='xsd:date' minOccurs='0' />
          </xsd:sequence>
          <xsd:attribute name='partNum' type='tns:SKU' />
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name='PurchaseOrderType'>
    <xsd:sequence>
      <xsd:element name='shipTo' type='tns:USAddress' />
      <xsd:element name='billTo' type='tns:USAddress' />
      <xsd:element ref='tns:comment' minOccurs='0' />
      <xsd:element name='items' type='tns:Items' />
    </xsd:sequence>
    <xsd:attribute name='orderDate' type='xsd:date' />
    <xsd:attribute name='confirmDate' type='xsd:date' use='required' />
  </xsd:complexType>
</xsd:schema>

```

**NOTE**

As empresas, as organizações, os produtos, os nomes de domínio, os endereços de email, os logotipos, as pessoas, os locais e os eventos de exemplo descritos aqui são fictícios. Nenhuma associação com nenhuma empresa, organização, produto, nome de domínio, endereço de email, logotipo, pessoa, locais ou eventos reais é intencional nem deve ser inferida.

# Arquivo XSD de exemplo: relações

14/05/2021 • 2 minutes to read

O arquivo XSD a seguir é usado em vários exemplos na documentação do Designer de Esquema XSD. Este arquivo é um esquema de ordem de compra com anotações e documentação.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="pet" type="PetType"/>

  <xs:attributeGroup name="NameAgeAttributes">
    <xs:attribute name="age" type="xs:integer" use="required"/>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:attributeGroup>

  <xs:complexType name="PetType">
    <xs:attributeGroup ref="NameAgeAttributes"/>
  </xs:complexType>

  <xs:element name="cat" substitutionGroup="pet">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="PetType">
          <xs:sequence>
            <xs:element name="weight" type="xs:integer"/>
            <xs:element name="color" type="xs:string"/>
            <xs:element name="breed" type="xs:integer"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

  <xs:element name="dog" substitutionGroup="pet">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="PetType">
          <xs:sequence>
            <xs:element name="weight" type="xs:integer"/>
            <xs:element name="color" type="xs:string"/>
            <xs:element name="breed" type="xs:integer"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>

</xs:schema>
```

## NOTE

As empresas, as organizações, os produtos, os nomes de domínio, os endereços de email, os logotipos, as pessoas, os locais e os eventos de exemplo descritos aqui são fictícios. Nenhuma associação com nenhuma empresa, organização, produto, nome de domínio, endereço de email, logotipo, pessoa, locais ou eventos reais é intencional nem deve ser inferida.

# Depuração do XSLT

14/05/2021 • 2 minutes to read

Você pode depurar o código XSLT no Visual Studio. O depurador XSLT dá suporte à definição de pontos de interrupção, à exibição de Estados de execução XSLT e assim por diante. O depurador XSLT pode ser usado para depurar folhas de estilo XSLT ou aplicativos XSLT.

Você pode executar o código uma linha de cada vez, passando para, percorrendo ou saindo do código. Os comandos para usar a funcionalidade de depuração de código do depurador XSLT são os mesmos dos outros depuradores do Visual Studio.

Uma vez que você iniciar a depuração, o depurador XSLT abrir janelas para mostrar o documento de entrada e saída XSLT.

## NOTE

O depurador XSLT só está disponível nas edições Professional e Enterprise do Visual Studio.

## Depurar do editor de XML

Você pode iniciar o depurador quando tiver uma folha de estilos ou um arquivo XML de entrada aberto no editor. Isso permite que você depure enquanto cria a folha de estilos.

1. Abra a folha de estilos ou o arquivo XML no Visual Studio.
2. Selecione **Iniciar Depuração XSLT** no menu **XML** ou pressione **ALT + F5**.

## Depurar de um aplicativo que usa XSLT

Você pode entrar no XSLT durante a depuração de um aplicativo. Quando você pressiona **F11** em uma `System.Xml.Xsl.XslCompiledTransform.Transform` chamada, o depurador pode entrar no código XSLT.

## NOTE

Entrar em XSLT da classe de `XslTransform` não é suportado. A classe de `XslCompiledTransform` é o único processador XSLT que suporte em avançar XSLT a depuração.

### Para iniciar a depuração de um aplicativo XSLT

1. Ao criar uma instância do objeto de `XslCompiledTransform`, defina o parâmetro de `enableDebug` a `true` em seu código. Isso informa o processador XSLT para criar informações de depuração quando o código é compilado.
2. Pressione **F11** para entrar no código XSLT.

A folha de estilos XSLT é carregada em uma nova janela de documento e o depurador XSLT é iniciado.

Como alternativa, você pode adicionar um ponto de quebra a folha de estilos e executar o aplicativo.

### Exemplo

O código a seguir é um exemplo de um programa C# XSLT. Mostra como ativar depuração XSLT.



```

using System;
using System.IO;
using System.Xml;
using System.Xml.Xsl;

namespace ConsoleApplication
{
    class Program
    {
        private const string sourceFile = @"c:\data\xsl_files\books.xml";
        private const string stylesheet = @"c:\data\xsl_files\below-average.xsl";
        private const string outputFile = @"c:\data\xsl_files\output.xml";

        static void Main(string[] args)
        {
            // Enable XSLT debugging.
            XslCompiledTransform xslt = new XslCompiledTransform(true);

            // Compile the style sheet.
            xslt.Load(stylesheet);

            // Execute the XSLT transform.
            FileStream outputStream = new FileStream(outputFile, FileMode.Append);
            xslt.Transform(sourceFile, null, outputStream);
        }
    }
}

```

## Criador de perfil XSLT

O [XSLT Profiler](#) é uma ferramenta que permite aos desenvolvedores medir, avaliar e direcionar problemas relacionados ao desempenho no código XSLT criando relatórios de desempenho XSLT detalhados. Para obter mais informações, consulte [XSLT Profiler](#).

## Consulte também

- [Passo a passo: Depurar uma folha de estilos XSLT](#)
- [Primeiro, veja o depurador do Visual Studio](#)
- [Noções básicas de depuração: pontos de interrupção](#)

# Passo a passo: Depurar uma folha de estilos XSLT

14/05/2021 • 4 minutes to read

As etapas nessa explicação passo a passo demonstra como usar o depurador XSLT. As etapas incluem variáveis de exibição, pontos de interrupção, e percorrendo o código. O depurador permite que você execute o código uma linha por vez.

Para se preparar para este passo a passos, primeiro copie os dois [arquivos de exemplo](#) para o computador local. Um é a folha de estilo e um é o arquivo XML que usaremos como entrada para a folha de estilos. Neste tutorial, a folha de estilos que usamos localiza todos os livros cujo custo está abaixo do preço médio do livro.

## NOTE

O depurador XSLT só está disponível nas edições Professional e Enterprise do Visual Studio.

## Iniciar a depuração

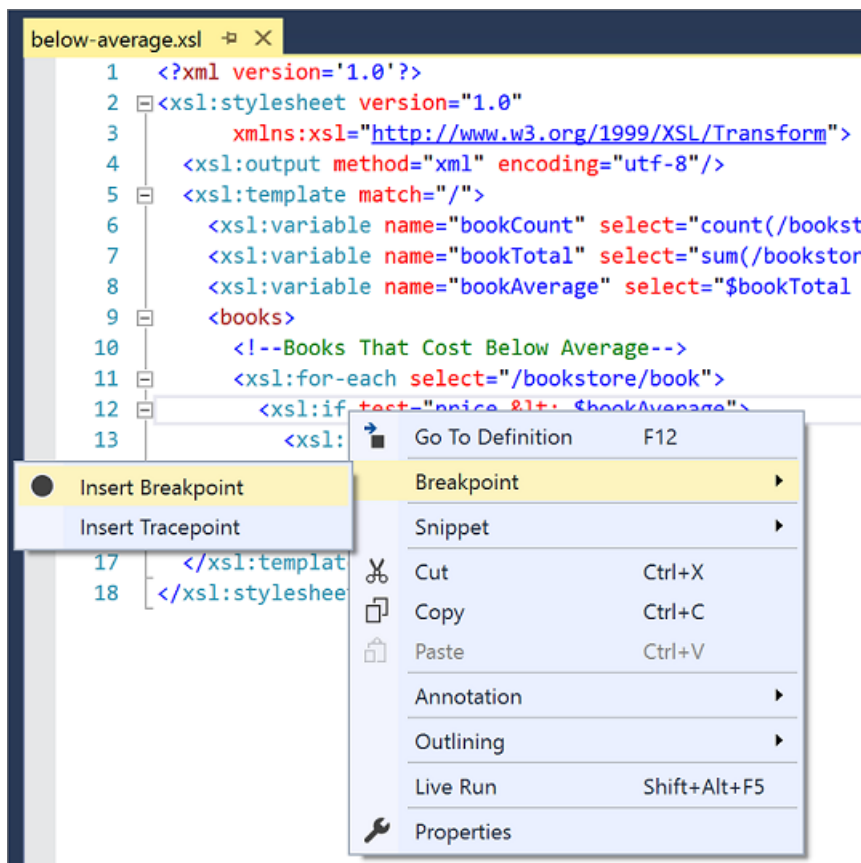
1. No menu **arquivo**, escolha **abrir** > **arquivo**.
2. Localize o arquivo *below-Average.xsl* e escolha **abrir**.

A folha de estilos é aberta no editor de XML.

3. Clique no botão procurar (...) no campo de **entrada** da janela Propriedades do documento. (Se a janela **Propriedades** não estiver visível, clique com o botão direito do mouse em qualquer lugar do arquivo aberto no editor e escolha **Propriedades**.)
4. Localize o arquivo *books.xml* e, em seguida, escolha **abrir**.

Isso define o arquivo de documento de origem que é usado para a transformação XSLT.

5. Defina um [ponto de interrupção](#) na linha 12 de *below-Average.xsl*. Você pode fazer isso de várias maneiras:
  - Clique na margem do editor na linha 12.
  - Clique em qualquer lugar na linha 12 e pressione F9.
  - Clique com o botão direito do mouse na `xsl:if` marca de início e escolha ponto de interrupção de inserção de **pontos** de interrupção > .



6. Na barra de menus, escolha XML > Iniciar Depuração XSLT (ou pressione ALT + F5).

O processo de depuração é iniciado.

No editor, o depurador é posicionado no `<xsl:if>` elemento da folha de estilos. Outro arquivo chamado *below-average.xml* é aberto no editor; Esse é o arquivo de saída que será populado como cada nó no arquivo de entrada *books.xml* é processado.

As janelas **automáticos**, **locais** e **inspecionar 1** aparecem na parte inferior da janela do Visual Studio. A janela **locais** exibe todas as variáveis locais e seus valores atuais. Isso inclui variáveis definidos na folha de estilos e também variáveis que o depurador usa para controlar os nós que estão atualmente no contexto.

## Janela Inspeccionar

Adicionaremos duas variáveis à janela **Watch 1** para que possamos examinar seus valores à medida que o arquivo de entrada for processado. (Você também pode usar a janela **locais** para examinar valores se as variáveis que você deseja observar já estiverem lá.)

1. No menu **depurar**, escolha **Windows > Watch > Watch 1**.

A janela **Watch 1** torna-se visível.

2. Digite `$bookAverage` o campo **nome** e pressione **Enter**.

O valor da `$bookAverage` variável é exibido no campo **valor**.

3. Na próxima linha, digite `self::node()` o campo **nome** e pressione **Enter**.

`self::node()` é uma expressão XPath que é avaliada como o nó de contexto atual. O valor da expressão XPath de `self::node()` é o primeiro nó de livro. Isso é alterado como nós progredimos com a transformação.

4. Expanda o `self::node()` nó e, em seguida, expanda o nó que é o valor `price`.

Watch 1		
Name	Value	Type
\$bookAverage	10.32333333333333	Number
self::node()	book	Element
@genre	autobiography	Attribute
@publicationdate	1981	Attribute
@ISBN	1-861003-11-0	Attribute
node()[1]	\n	Whitespace
node()[2]	title	Element
node()[3]	\n	Whitespace
node()[4]	author	Element
node()[5]	\n	Whitespace
node()[6]	price	Element
node()[1]	8.99	Text
node()[7]	\n	Whitespace

Você pode ver o valor do preço do livro para o nó do livro atual e compará-lo com o `$bookAverage` valor. Como o preço do livro está abaixo da média, a `xs1:if` condição deve ter sucesso quando você continua o processo de depuração.

## Percorrer o código

1. Pressione **F5** para continuar.

Como o primeiro nó de livro satisfaz a `xs1:if` condição, o nó de livro é adicionado ao arquivo de saída *below-average.xml*. O depurador continuará a ser executado até que está localizado novamente no elemento de `xs1:if` na folha de estilos. O depurador agora está posicionado no segundo nó de livro no arquivo de *books.xml*.

Na janela **Watch 1**, o `self::node()` valor muda para o segundo nó de livro. Examinando o valor do elemento de preço, você pode determinar que o preço está acima da média, então a condição de `xs1:if` deve falhar.

2. Pressione **F5** para continuar.

Como o segundo nó de livro não atende à `xs1:if` condição, o nó de livro não é adicionado ao arquivo de saída *below-average.xml*. O depurador continuará a ser executado até ser posicionado novamente no `xs1:if` elemento na folha de estilos. O depurador agora está posicionado no terceiro `book` nó do arquivo de *books.xml*.

Na janela **Watch 1**, o `self::node()` valor muda para o terceiro nó de livro. Examinando o valor do `price` elemento, você pode determinar se o preço está abaixo da média. A `xs1:if` condição deve ter sucesso.

3. Pressione **F5** para continuar.

Como a `xs1:if` condição foi satisfeita, o terceiro livro é adicionado ao arquivo de saída *below-average.xml*. Todos os livros no documento XML foram processados e paradas do depurador.

## Arquivos de exemplo

Os seguintes dois arquivos são usados por passo a passo.

**below-Average. xsl**

```

<?xml version='1.0'?>
<xsl:stylesheet version="1.0"
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="xml" encoding="utf-8"/>
  <xsl:template match="/">
    <xsl:variable name="bookCount" select="count(/bookstore/book)"/>
    <xsl:variable name="bookTotal" select="sum(/bookstore/book/price)"/>
    <xsl:variable name="bookAverage" select="($bookTotal div $bookCount)"/>
    <books>
      <!--Books That Cost Below Average-->
      <xsl:for-each select="/bookstore/book">
        <xsl:if test="price < $bookAverage">
          <xsl:copy-of select="."/>
        </xsl:if>
      </xsl:for-each>
    </books>
  </xsl:template>
</xsl:stylesheet>

```

## books.xml

```

<?xml version='1.0'?>
<!-- This file represents a fragment of a book store inventory database -->
<bookstore>
  <book genre="autobiography" publicationdate="1981" ISBN="1-861003-11-0">
    <title>The Autobiography of Benjamin Franklin</title>
    <author>
      <first-name>Benjamin</first-name>
      <last-name>Franklin</last-name>
    </author>
    <price>8.99</price>
  </book>
  <book genre="novel" publicationdate="1967" ISBN="0-201-63361-2">
    <title>The Confidence Man</title>
    <author>
      <first-name>Herman</first-name>
      <last-name>Melville</last-name>
    </author>
    <price>11.99</price>
  </book>
  <book genre="philosophy" publicationdate="1991" ISBN="1-861001-57-6">
    <title>The Gorgias</title>
    <author>
      <name>Plato</name>
    </author>
    <price>9.99</price>
  </book>
</bookstore>

```

## Confira também

- [Depuração de XSLT](#)

# Avaliar expressões XPath

14/05/2021 • 2 minutes to read

Você pode avaliar as expressões XPath usando a janela **QuickWatch** durante a depuração. A expressão XPath deve ser válido de acordo com a recomendação XPath 1,0 W3C. O contexto XSLT atual (ou seja, o `self::node()` nó na janela **locais** ) fornece o contexto de avaliação para a expressão XPath.

Ao avaliar uma expressão XPath:

- As funções internas XPath são suportadas.
- Não há suporte para funções XSLT internas e funções definidas pelo usuário.

## NOTE

A depuração XSLT só está disponível na edição Enterprise do Visual Studio.

## Avaliar uma expressão XPath

O procedimento a seguir usa o *below-Average.xsl* e os arquivos de *books.xml* da [instrução: Depurar uma página de folha de estilos XSLT](#).

1. Inserir um ponto de interrupção na tag de início de `xsl:if`.
2. Para iniciar a depuração, escolha **XML > Iniciar Depuração XSLT** na barra de menus (ou pressione **ALT + F5**).

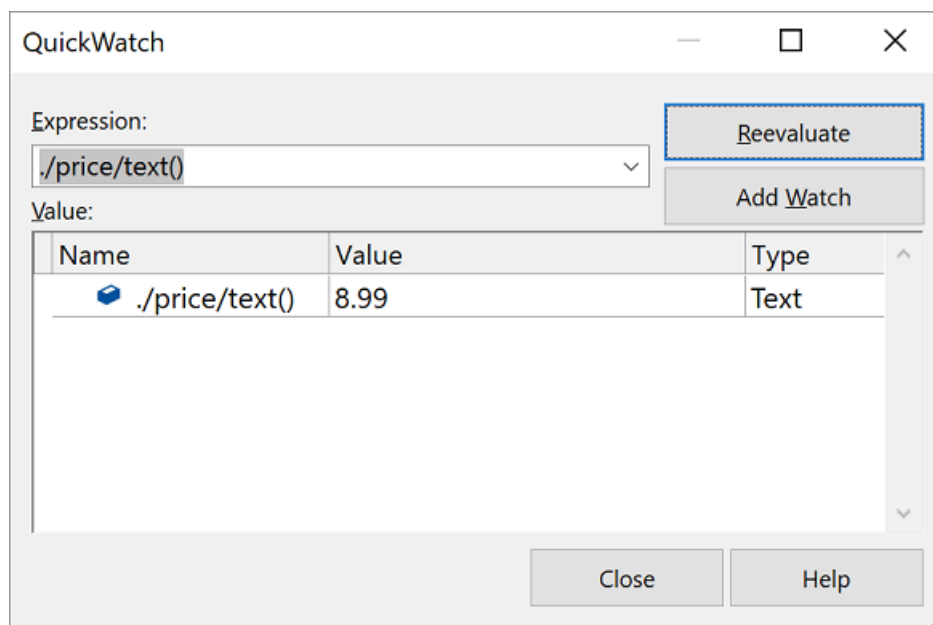
Inicia e as quebras do depurador na marca `xsl:if`.

3. Clique com o botão direito do mouse e selecione **QuickWatch**.

A janela **QuickWatch** é aberta.

4. Insira `./price/text()` no campo **expressão** da caixa de diálogo **QuickWatch** e escolha **reavaliar**.

O preço do nó do livro atual aparece na caixa **valor**.



5. Altere a expressão XPath para `./price/text() < $bookAverage` e clique em **reavaliar**.

A caixa **valor** mostra que a expressão XPath é avaliada como `true` .

## Consulte também

- [Depuração de XSLT](#)

# XSLT (interface do usuário do depurador)

14/05/2021 • 3 minutes to read

Este artigo descreve as janelas do depurador e as caixas de diálogo. Ele aborda apenas as partes da interface do usuário que têm comportamento de depuração específico ao XSLT.

Para obter mais informações, consulte a [referência da interface do usuário de depuração](#).

## Janela Locais

A janela locais exibe informações sobre todas as variáveis definidos na folha de estilos. A janela locais contém três colunas de informações:

### Nome

Esta coluna contém os nomes de todas as variáveis locais no escopo atual. Os conjuntos de nós têm um controle de árvore que você pode fazer uma busca detalhada para ver suas subpastas.

### Valor

Esta coluna mostra o valor contido por cada variável. O atributo, a instrução de processamento, o comentário, texto, e os nós CDATA exibem o valor de texto do nó. Os nós de namespace exibem URI de namespace.

### Tipo

Esta coluna identifica o tipo de dados de cada variável listada na coluna **nome**.

A janela locais também exibe as variáveis predefinidos de contexto que acompanham o contexto de transformação XSLT. A tabela a seguir descreve as variáveis predefinidos de contexto usados pelo depurador XSLT.

NOME	DESCRIÇÃO
<code>last()</code>	O tamanho do contexto.
<code>position()</code>	A posição, ou número de índice, o nó de contexto, relativo ao tamanho do contexto.
<code>self::node()</code>	O valor do nó de contexto.

## janela Saída

A janela de saída mostra todas as mensagens de erro ou exceções de segurança que ocorrem durante a depuração. Ele também mostra a saída do depurador.

## Lista de Tarefas

O **lista de tarefas** lista todos os erros de compilação na folha de estilos. Clique duas vezes no erro leva o cursor para a linha com o erro.

O **lista de tarefas** inclui todos os erros que ocorrem nos blocos de script no arquivo XSLT.



#### NOTE

O depurador XSLT não tem avisos, então eles nunca aparecem na **lista de tarefas**.

## Janela Pontos de Interrupção

A janela de pontos de interrupção mostra os pontos de interrupção definidos no projeto atual. Se um ponto de interrupção é adicionado quando a janela está em modo de exibição, a janela é atualizado automaticamente para mostrar o novo ponto de interrupção.

A janela de pontos de interrupção deve se comportar da mesma maneira que outros depuradores do Visual Studio.

## Janela Inspeccionar

A janela de observação é usada para avaliar as variáveis. Você também pode alterar os valores das variáveis.

Variáveis exibidos na janela de observação são para o contexto atual (o item top-most na pilha de chamadas). Se você alterar o contexto, a janela de observação atualiza e exibe as variáveis definidas para esse contexto.

## janela de Pilha de Chamadas

A janela **pilha de chamadas** é usada para exibir os nomes das funções na pilha de chamadas, os tipos de parâmetro e os valores de parâmetro. Informações de pilha de chamadas é mostrada somente quando o programa que está sendo depurado está em um estado de interrupção.

A pilha de chamadas representa os vários contextos que a execução de fonte está atravessando. Por exemplo, se houver uma chamada do modelo "a" para o modelo "b", o modelo "a" e o modelo "b" aparecerão na janela **pilha de chamadas** com o contexto atual na parte superior da lista. O usuário pode ver a consulta que está em execução atualmente.

Se os modelos não têm um nome para o arquivo XSLT, os nomes gerados pelo processador XSLT são usados.

Clique em um item diferente de aquele na parte superior da lista indica ao visualizador onde a ramificação de execução XSLT ocorreu usando realce padrão de verde e setas verde.

## QuickWatch (caixa de diálogo)

A caixa de diálogo **QuickWatch** é usada para avaliar as expressões XPath 1,0. O nó de contexto (o nó de `self::node()` da janela locais) fornece o contexto para a execução da expressão XPath. O resultado de executar a expressão XPath é exibido na janela de observação.

A lista a seguir descreve as restrições sobre a avaliação da expressão XPath:

- Somente as funções internas XPath são permitidas.
- Funções XSLT internas, como `document()` e `key()` não são permitidas.
- As funções definidas pelo usuário não são permitidas.

Para obter mais informações, consulte [como: avaliar uma expressão XPath](#).

## janela de Desmontagem

A janela de desmontagem mostra o código do assembly que é gerado pelo compilador XSLT. Esta janela pode ser usada da mesma forma como quaisquer outras janelas de desmontagem do Visual Studio.

Para obter mais informações, [como: usar a janela de desmontagem](#).

## Confira também

- [Depuração de XSLT](#)
- [Introdução ao depurador](#)
- [Inspeccionar variáveis nas janelas automáticas e locais no Visual Studio](#)

# Modelos padrão XSLT

14/05/2021 • 2 minutes to read

Um modelo padrão será usado durante XSLT que processa quando não há nenhuma regra explícita modelo correspondente da folha de estilos. O modelo padrão, também conhecido como regra de modelo interno, é definido na seção 5,8 de recomendação W3C XSLT 1,0. O modelo padrão permite que o processador XSLT processe um nó, mesmo que não haja nenhuma regra explícita de modelo que corresponde a ele. Entretanto, porque a regra de modelo interno não é explicitamente definida na folha de estilos, isso pode levar a resultados inesperados ou confundindo de transformação XSLT.

O depurador XSLT agora exibe o código de modelos de opção XSLT. Quando você percorre uma transformação XSLT, se um modelo padrão é usado, o depurador exibe o modelo padrão em uma janela. Isso permite que você percorrer o código de modelo padrão e os pontos de interrupção em suas declarações.

## Confira também

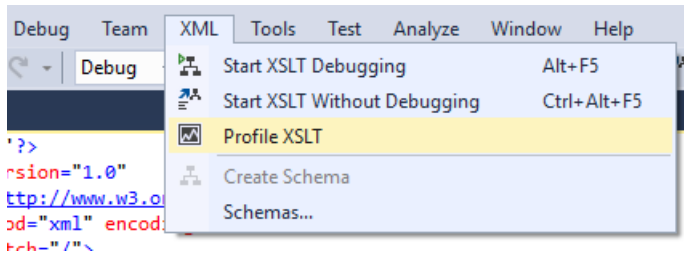
- [Depuração de XSLT](#)

# O criador de perfil XSLT

14/05/2021 • 4 minutes to read

O profiler XSLT criar relatórios de desempenho detalhados XSLT que a medida da ajuda, avalia, e problemas de desempenho relacionados de destino no código XSLT. O profiler XSLT inclui dicas úteis para XSL e otimizações de folha de estilos XSLT. Para aplicativos que requerem XSLT máximo desempenho, essa ferramenta pode ser essencial.

O XSLT Profiler faz parte do Visual Studio e está disponível no menu **XML**.



## NOTE

O XSLT Profiler só está disponível na edição Enterprise do Visual Studio 2017.

## Criar um relatório de desempenho

1. Abra um documento XSLT no Visual Studio 2017.
2. Na barra de menus, escolha > **XSLT de perfil XML**.
3. Fornecer um documento XML de entrada. Se um documento XML ele já não estiver aberto, você será solicitado para o arquivo.

Inicia a análise, e uma barra de progresso exibem o progresso no editor. A saída XSLT também é visível.

4. Após a conclusão da sessão de desempenho, verifique o relatório de desempenho para analisar o desempenho do XSLT.

## Obter todas as exibições disponíveis

1. Clique na lista suspensa **exibição atual** para obter todas as exibições disponíveis.
2. Selecione a opção **modo de exibição de resumo** na lista suspensa **exibição atual**. Por padrão, um relatório de desempenho é exibido na **exibição de resumo**. Esta exibição é um ponto de partida para determinar problemas de desempenho com documentos XSLT. A **exibição de resumo** lista os seguintes pontos de dados:
  - A maioria chamaram funções
  - Funções com a maioria de trabalho individual
  - Funções que usam o tempo os mais longa de executar

Por padrão, há três colunas para cada ponto de dados: o nome da função, o número de chamadas o valor absoluto, e um valor percentual de função chamado para chamadas de função total. De cada ponto de dados na **exibição de resumo**, você pode navegar para exibições mais detalhadas clicando com o botão

direito do mouse nos pontos de dados da função.

3. Selecione a opção de **exibição de função** na lista suspensa **exibição atual**. A **exibição de função** lista as funções chamadas durante a criação de perfil. Você pode classificar os dados em um nome de coluna. As colunas exibidas por padrão são:

- Nome da função
- Tempo Inclusivo Decorrido
- Tempo Exclusivo Decorrido
- Tempo Inclusivo do Aplicativo
- Tempo Exclusivo do Aplicativo
- Número de Chamadas

Todas as colunas de tempo são exibidas em valores absolutos e em porcentagens. O termo **exclusivo** refere-se ao tempo total que uma função gastou em execução exclusiva de tempo gasto por outras funções chamadas durante a execução dessa função.

O termo **inclusivo** refere-se ao tempo total que uma função gastou executando, incluindo o tempo de execução de todas as funções que ele chamou e se alguma delas chamou funções chamadas de outras funções.

## O modo de seleção do chamador/receptor

Selecione o modo de exibição **chamador/receptor** na lista suspensa **exibição atual**. O modo de exibição **chamador/receptor** tem as três partes distintas a seguir:

- **Funções que chamaram:** todas as funções que chamaram uma função específica são listadas na parte superior da exibição.
- **Função atual:** a função específica que foi chamada está listada na parte intermediária do modo de exibição.
- **Funções que foram chamadas por:** todas as funções que foram chamadas pela função específica são listadas na parte inferior da exibição.

Se uma função chamada `SyncToNavigator` aparece na parte média de exibição, todas as funções que chamaram a função de `SyncToNavigator` aparecem na parte superior de exibição, e em todas as funções que foram chamados por `SyncToNavigator` aparecem na parte de fundo de exibição.

- Você pode alterar a função na parte média de exibição clicando duas vezes em algumas das funções listadas em duas outras partes de exibição. A exibição é atualizado para refletir automaticamente as alterações.
- Você também pode classificar os dados clicando em nomes de coluna.

## Selecionar exibição de árvore de chamada

- Selecione **modo de exibição de árvore de chamada** na lista suspensa **exibição atual**. Esta exibição é um modo de exibição de árvore de execução do programa.

O **modo de exibição de árvore de chamadas** mostra a raiz da árvore como o nome do processo. As funções são os nós de árvore. Esta exibição permite que você fure em rastreamentos específicos de chamada e analise que os rastreamentos têm o maior impacto de desempenho. A exibição é semelhante à **exibição da pilha de chamadas** disponível durante a depuração. Além das colunas no **modo de**

**exibição de função**, no modo de exibição de árvore de **chamada**, há uma coluna adicional para exibir o **nome do módulo**.

- Selecione **marcas** na lista suspensa **exibição atual** .

Com o XSLT Profiler, há marcas que aparecem no fluxo de coleta de dados com um comentário associado. As marcas são locais no código que têm contadores. Quando você indica que o profiler XSLT para coletar contadores de desempenho XSLT, os contadores obtém como cada vez que uma dessas marcas é executado. Os dados são exibidos em uma tabela que contém a **ID de marca**, o nome da **marca** (**Iniciar programa**, **encerrar programa**) e o **carimbo de data/hora**. As marcas não são agregadas e aparecem em ordem cronológica na exibição de **marcas** do relatório de desempenho.

## Módulos selecionados na exibição atual

- Selecione **módulos** na lista suspensa **exibição atual** .

Exibição de módulos é uma lista plana das funções agregadas para o nível de módulo. Expandir ou recolher o nome do módulo para exibir ou fechar a exibição de dados de desempenho do módulo. Você pode classificar os dados em um nome de coluna. Por padrão, há valores absolutos e números percentuais para **tempo inclusivo decorrido**, **tempo exclusivo decorrido**, **tempo inclusivo do aplicativo**, **tempo exclusivo do aplicativo** e **número de chamadas**.

- Selecione **processar** na lista suspensa **exibição atual** .

A exibição processo exibe uma tabela que inclui a **ID do processo**, o **nome do processo**, a hora de **início** e a **hora de término**. Os dados podem ser classificados clicando em nomes de coluna.

## Confira também

- [Walkthrough: usando a hierarquia XSLT](#)

# Considerações de segurança ao trabalhar com dados XML

14/05/2021 • 2 minutes to read

Este tópico aborda os problemas de segurança que você precisa saber ao trabalhar com o editor de XML ou o depurador XSLT.

## Editor de XML

O editor de XML é baseado no editor de texto do Visual Studio. Depende de classes de [System.Xml](#) e de [System.Xml.Xsl](#) para manipular muitos dos processos XML.

- As transformações XSLT são executadas em um domínio de aplicativo. As transformações XSLT estão em *área restrita*; ou seja, a política de segurança de acesso do código do seu computador é usada para determinar as permissões restritas com base em onde a folha de estilos XSLT está localizada. Por exemplo, folhas de estilos de um local da Internet tem as permissões as mais rígidas, enquanto as folhas de estilos copiaram ao seu disco rígido executado com confiança total.
- A classe de [XslCompiledTransform](#) é usado para compilar XSLT a Microsoft intermediate language para aumentar o desempenho durante a execução.
- Esquemas que apontam para um local externo no arquivo de catálogo são baixados automaticamente quando o editor de XML é carregado pela primeira vez. A classe de [XmlSchemaSet](#) é usada para criar esquemas. O arquivo de catálogo fornecido com o editor de XML não tem links para esquemas externos. O usuário precisa adicionar explicitamente uma referência ao esquema externo antes que o editor de XML Baixe o arquivo de esquema. O download de HTTP pode ser desabilitado por meio da página de **Opções de ferramentas diversas** para o editor de XML.
- O editor de XML usa as [System.Net](#) classes para baixar esquemas

## Depurador XSLT

O depurador XSLT usa o mecanismo e as classes gerenciadas Visual Studio de depuração de [System.Xml](#) e do espaço de [System.Xml.Xsl](#) .

- O depurador XSLT executa cada transformação XSLT em um domínio de aplicativo na área restrita. A política de segurança de acesso a código do seu computador é usada para determinar as permissões restritas com base em onde a folha de estilos XSLT é encontrada. Por exemplo, folhas de estilos de um local da Internet tem as permissões as mais rígidas, enquanto as folhas de estilos copiaram ao seu disco rígido executado com confiança total.
- A folha de estilos XSLT é compilada usando a classe de [XslCompiledTransform](#) .
- O avaliador de expressão XSLT é carregado pelo mecanismo gerenciado de depuração. O mecanismo gerenciado de depuração supõe que qualquer código é executado do computador local do usuário. Da mesma forma, a classe de [XslCompiledTransform](#) download do arquivo fonte para o computador local do usuário. A possibilidade que um ataque de elevação de privilégio em execução pode ocorrer é abrandada executando todas as transformações XSLT em um domínio de aplicativo com permissões restritas

## Confira também

- Domínios de aplicativo