XML Schema complexType Element

[❮ Complete XML Schema Reference](https://www.w3schools.com/xml/schema_elements_ref.asp)

Definition and Usage

The complexType element defines a complex type. A complex type element is an XML element that contains other elements and/or attributes.

Element Information

* **Parent elements:** element, redefine, schema

Syntax

<complexType  
id=ID  
name=NCName  
abstract=true|false  
mixed=true|false  
block=(#all|list of (extension|restriction))  
final=(#all|list of (extension|restriction))  
*any attributes*  
>  
  
(annotation?,(simpleContent|complexContent|((group|all|  
choice|sequence)?,((attribute|attributeGroup)\*,anyAttribute?))))  
  
</complexType>

(The ? sign declares that the element can occur zero or one time, and the \* sign declares that the element can occur zero or more times inside the complexType element)

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| id | Optional. Specifies a unique ID for the element |
| name | Optional. Specifies a name for the element |
| abstract | Optional. Specifies whether the complex type can be used in an instance document. True indicates that an element cannot use this complex type directly but must use a complex type derived from this complex type. Default is false |
| mixed | Optional. Specifies whether character data is allowed to appear between the child elements of this complexType element. Default is false. If a simpleContent element is a child element, the mixed attribute is not allowed! |
| block | Optional. Prevents a complex type that has a specified type of derivation from being used in place of this complex type. This value can contain #all or a list that is a subset of extension or restriction:   * extension - prevents complex types derived by extension * restriction - prevents complex types derived by restriction * #all - prevents all derived complex types |
| final | Optional. Prevents a specified type of derivation of this complex type element. Can contain #all or a list that is a subset of extension or restriction.   * extension - prevents derivation by extension * restriction - prevents derivation by restriction * #all - prevents all derivation |
| *any attributes* | Optional. Specifies any other attributes with non-schema namespace |

Example 1

The following example has an element named "note" that is of a complex type:

<xs:element name="note">  
  <xs:complexType>  
    <xs:sequence>  
  <xs:element name="to" type="xs:string"/>  
  <xs:element name="from" type="xs:string"/>  
  <xs:element name="heading" type="xs:string"/>  
  <xs:element name="body" type="xs:string"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>

Example 2

The following example has a complex type, "fullpersoninfo", that derives from another complex type, "personinfo", by extending the inherited type with three additional elements (address, city and country):

<xs:element name="employee" type="fullpersoninfo"/>  
  
<xs:complexType name="personinfo">  
  <xs:sequence>  
    <xs:element name="firstname" type="xs:string"/>  
    <xs:element name="lastname" type="xs:string"/>  
  </xs:sequence>  
</xs:complexType>  
  
<xs:complexType name="fullpersoninfo">  
  <xs:complexContent>  
    <xs:extension base="personinfo">  
      <xs:sequence>  
        <xs:element name="address" type="xs:string"/>  
        <xs:element name="city" type="xs:string"/>  
        <xs:element name="country" type="xs:string"/>  
      </xs:sequence>  
    </xs:extension>  
  </xs:complexContent>  
</xs:complexType>

In the example above the "employee" element must contain, in sequence, the following elements: "firstname", "lastname", "address", "city", and "country".