

Data file

The Template Editor allows you to create a new data file and import it to the Report Wizard Template Editor dialog. Click the Import button to import the data file. The data file is written in the *XML* format. The following is the example of an *XML* data file schema, see the code snippet below.

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="data">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="elements">
          <xs:complexType>
            <xs:sequence>
              <xs:element minOccurs="0" maxOccurs="unbounded" name="element">
                <xs:complexType>
                  <xs:all>
                    <xs:element minOccurs="1" maxOccurs="1" name="name" type="xs:string" />
                    <xs:element minOccurs="0" maxOccurs="1" name="description" type="xs:string" />
                    <xs:element minOccurs="0" maxOccurs="1" name="members">
                      <xs:complexType>
                        <xs:sequence>
                          <xs:element minOccurs="0" maxOccurs="unbounded" name="function">
                            <xs:complexType>
                              <xs:sequence>
                                <xs:element minOccurs="1" maxOccurs="1" name="name" type="xs:string" />
                                <xs:element minOccurs="0" maxOccurs="1" name="description" type="xs:string" />
                                <xs:element minOccurs="0" maxOccurs="unbounded" name="param">
                                  <xs:complexType>
                                    <xs:all>
                                      <xs:element minOccurs="1" maxOccurs="1" name="name" type="xs:string" />
                                      <xs:element minOccurs="0" maxOccurs="1" name="description" type="xs:string" />
                                      <xs:element minOccurs="0" maxOccurs="1" name="type" nillable="true" type="xs:string" />
                                      <xs:element minOccurs="0" maxOccurs="1" name="direction">
                                        <xs:simpleType>
                                          <xs:restriction base="xs:string">
                                            <xs:enumeration value="in" />
                                            <xs:enumeration value="out" />
                                          </xs:restriction>
                                        </xs:simpleType>
                                      </xs:element>
                                    </xs:all>
                                  </xs:complexType>
                                </xs:element>
                              </xs:sequence>
                            </xs:complexType>
                          </xs:element>
                        </xs:sequence>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="attribute">
          <xs:complexType>
            <xs:all>
              <xs:element minOccurs="1" maxOccurs="1" name="name" type="xs:string" />
              <xs:element minOccurs="0" maxOccurs="1" name="description" type="xs:string" />
              <xs:element minOccurs="0" maxOccurs="1" name="type" nillable="true" type="xs:string" />
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:attribute name="often" type="xs:boolean" />
</xs:schema>

```

The table below lists the XML Data File Schema Elements and Their Descriptions

Element	Attribute / Element	Description
data		A root element.
	name : string	A data name. This value will be used as a category name in (3) Category.
	elements : complexType	A group of <element> elements.
elements		A group of <element> elements.
element		An element display on (4) List of fields.
	name : string	An element's name.
	description : string	An element description.
	often : boolean	If an attribute is true, the element will be managed as a suggested element.
	members : complexType	A group of <function> or <attribute>.
function		A function member.
	name : string	A function's name.
	description : string	A function description.
	param : complexType	A group of function parameters.
param		A function parameter.
	name : string	A parameter's name.
	description : string	A parameter description.
	type : string	A parameter type.
	direction : simpleType	The direction of a parameter: • in indicates this parameter is an input. • out indicates this parameter is an output.
attribute		An attribute member.
	name : string	An attribute's name.
	description : string	An attribute description.
	type : string	An attribute type.

The following code snippet is an example of an XML data file.

```

<?xml version="1.0" encoding="UTF-8"?>
<data name="Built-in Tools" xmlns:xsi="http://www.w3.org/2001/XMLSchemainstance"
xsi:noNamespaceSchemaLocation="fields.xsd">
  <elements>
    <element often="true">
      <name>array</name>
      <description>Use for creating the Array or HashSet instance.</description>
      <members>
        <function>
          <name>subList</name>
          <description>Create ArrayList of the portion of list in given size.</description>
          <param>
            <name>list</name>
            <direction>in</direction>
            <type>List</type>
            <description>a original list.</description>
          </param>
          <param>
            <name>size</name>
            <direction>in</direction>
            <type>int</type>
            <description>view size of given list</description>
          </param>
          <param>
            <name>list</name>
            <direction>out</direction>
            <type/>
            <description>a of view of the specified size within given list.</description>
          </param>
        </function>
      </members>
    </element>
  </elements>
</data>

```