

Notice! This page describes the *nature of the error* using a hypothetical example and not the erroneous data of the input test file. You should however be able to apply this information to your error case.



## General description of the error:

**The format of the error message:** cvc-identity-constraint.4.3: Key '{0}' with value '{1}' not found for identity constraint of element '{3}'.

**Error description in schema standard:** <http://www.w3.org/TR/2007/WD-xmlsche...ity-constraint>

**Possible causes for this error:**

- xs:keyref definition is used in the schema. The value given for element defined as keyref does not have a corresponding key value in the scope defined.

## An example

```
<database>
  <items>
    <shirt number="1">
  </items>
  <products>
    <product>
      <number>0</number>
      <size>2</size>
    </product>
  </products>
</database>
```

**Error message shown:** Error *cvc-identity-constraint.4.3*: Key 'productNumberKeyRef' with value '1' not found for identity constraint of element 'database'.

**How to fix:** You can start of by finding out which element has the incorrect value. In this simple example there is only one element or attribute having value '1' and that is the number attribute of shirt element. You can then check the schema to see to which element the 'prodNumKeyRef' refers to and make the correction based on that. In this case the schema is like follows:

```
<xs:keyref name="productNumberKeyRef" refer="productNumberKey">
  <xs:selector xpath="items/*"/>
  <xs:field xpath="@number"/>
</xs:keyref>
```

As we can see the number attribute is the keyRef and we can also see that *productNumberKeyRef* refers to *productNumberKey*. The *productNumberKey* definition in turn is the following:

```
<xs:key name="productNumberKey">
  <xs:selector xpath="./product"/>
  <xs:field xpath="number"/>
</xs:key>
```

From which we can see that the number attribute should match one of the values of number elements found under the product element(s). How to correct this depends on what you are trying to achieve, but the fix could be e.g. changing the number attribute of shirt to '0', adding new product element or changint the current product to have number child element with value '1'.

```
<database>
  <items>
    <shirt number="1">
  </items>
  <products>
    <product>
      <number>1</number>
      <size>2</size>
    </product>
  </products>
</database>
```