



Description of the method

Description:	<p>Logical operator implies consist of two parts. Whenever the first part is true, the second part has to be true as well</p> <p><i>expr1 implies expr2</i>: returns true when both expressions are true</p> <p><i>expr1 implies expr2</i>: when expr1 is not true, true is returned (in other words, expr2 value matters only when expr1 applies)</p>
Available for:	base64binary, boolean, datetime, double/decimal, date, hexBinary, integer, string, double/decimal and integer
Parameters:	-
Return type:	boolean

Example1

Context:	TransactionType1
OCL:	<p>self.Amount > 200 implies</p> <p>self.Debtor.Name->size() = 1 and self.Creditor.Name->size() = 1</p>
Description:	The example rule mandates the usage of both Creditor and Debtor name when Amount exceeds 200

The XML snippet below would pass this check. Note the second occurrence of transaction, expression is true because amount is not above 200.

```
<Transaction>
  <Id>203</Id>
  <Amount>201</Amount>
  <Debtor>
```

```

    <Name>Debtor2</Name>
  </Debtor>
  <Creditor>
    <Name>Creditor2</Name>
  </Creditor>
</Transaction>
<Transaction>

<Transaction>
  <Id>203</Id>
  <Amount>1</Amount>
  <Creditor>
    <Name>Creditor2</Name>
  </Creditor>
</Transaction>
<Transaction>

```

The other snippet below however would not pass

```

<Transaction>
  <Id>203</Id>
  <Amount>201</Amount>
  <Creditor>
    <Name>Creditor2</Name>
  </Creditor>
</Transaction>
<Transaction>

```

Please note that in order to make the feedback for user as accurate as possible, this exact rule may be divided into two separate rules. The individual rules would be:

if Amt exceeds 200, then Creditor Name is mandatory

if Amt exceeds 200, then Debtor Name is mandatory

This way error report will always point out the erroneous case, as the query and error message for each rule can be different.