

CoreOperationOr.java - CoreOperationAnd.java

```

1/*
2 * Licensed to the Apache Software Foundation (ASF) under one or more
3 * contributor license agreements. See the NOTICE file distributed with
4 * this work for additional information regarding copyright ownership.
5 * The ASF licenses this file to You under the Apache License, Version 2.0
6 * (the "License"); you may not use this file except in compliance with
7 * the License. You may obtain a copy of the License at
8 *
9 *     http://www.apache.org/licenses/LICENSE-2.0
10 *
11 * Unless required by applicable law or agreed to in writing, software
12 * distributed under the License is distributed on an "AS IS" BASIS,
13 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
14 * See the License for the specific language governing permissions and
15 * limitations under the License.
16 */
17package org.apache.commons.xpath.ri.compiler;
18
19import org.apache.commons.xpath.ri.EvalContext;
20import org.apache.commons.xpath.ri.InfoSetUtil;
21
22/**
23 * Implementation of {@link Expression} for the operation "or".
24 *
25 * @author Dmitri Plotnikov
26 * @version $Revision: 652845 $ $Date: 2008-05-02 12:46:46 -0500 (Fri, 02 May 2008)
27 */
28public class CoreOperationOr extends CoreOperation {
29
30    /**
31     * Create a new CoreOperationOr.
32     * @param args or'd Expression components
33     */
34    public CoreOperationOr(Expression[] args) {
35        super(args);
36    }
37
38    public Object computeValue(EvalContext context) {
39        for (int i = 0; i < args.length; i++) {
40            if (InfoSetUtil.booleanValue(args[i].computeValue(context))) {
41                return Boolean.TRUE;
42            }
43        }
44        return Boolean.FALSE;
45    }
46
47    protected int getPrecedence() {
48        return OR_PRECEDENCE;
49    }
50
51    protected boolean isSymmetric() {
52        return true;
53    }
54
55    public String getSymbol() {

```

```

1/*
2 * Licensed to the Apache Software Foundation (ASF) under one or more
3 * contributor license agreements. See the NOTICE file distributed with
4 * this work for additional information regarding copyright ownership.
5 * The ASF licenses this file to You under the Apache License, Version 2.0
6 * (the "License"); you may not use this file except in compliance with
7 * the License. You may obtain a copy of the License at
8 *
9 *     http://www.apache.org/licenses/LICENSE-2.0
10 *
11 * Unless required by applicable law or agreed to in writing, software
12 * distributed under the License is distributed on an "AS IS" BASIS,
13 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
14 * See the License for the specific language governing permissions and
15 * limitations under the License.
16 */
17package org.apache.commons.xpath.ri.compiler;
18
19import org.apache.commons.xpath.ri.EvalContext;
20import org.apache.commons.xpath.ri.InfoSetUtil;
21
22/**
23 * Implementation of {@link Expression} for the operation "and".
24 *
25 * @author Dmitri Plotnikov
26 * @version $Revision: 652845 $ $Date: 2008-05-02 12:46:46 -0500 (Fri, 02 May 2008)
27 */
28public class CoreOperationAnd extends CoreOperation {
29
30    /**
31     * Create a new CoreOperationAnd.
32     * @param args to combine
33     */
34    public CoreOperationAnd(Expression[] args) {
35        super(args);
36    }
37
38    public Object computeValue(EvalContext context) {
39        for (int i = 0; i < args.length; i++) {
40            if (!InfoSetUtil.booleanValue(args[i].computeValue(context))) {
41                return Boolean.FALSE;
42            }
43        }
44        return Boolean.TRUE;
45    }
46
47    protected int getPrecedence() {
48        return AND_PRECEDENCE;
49    }
50
51    protected boolean isSymmetric() {
52        return true;
53    }
54
55    public String getSymbol() {

```

CoreOperationOr.java - CoreOperationAnd.java

```
56     return "or";  
57 }  
58 }  
59
```

```
56     return "and";  
57 }  
58 }  
59
```