

Relational operators

Relational operators enable two values or variables to be “compared”, to create a boolean expression. The boolean expression returns a Boolean value, either “true or false”. Boolean expressions are used in conditions of control structures (eg Binary selection and WHILE loop). Table1 shows the syntax of the relational operators in some common programming languages.

| Relational operator | PHP | VB | Java | Pascal | C | Pseudocode |
|-----------------------|-----|----|------|--------|----|------------|
| Equal | == | = | == | = | == | |
| Not equal | != | <> | != | <> | /= | |
| Greater than | > | > | > | > | > | |
| Less than | < | < | < | < | < | |
| Greater than or equal | >= | >= | >= | >= | >= | |
| Less than or equal | <= | <= | <= | <= | <= | |

Table1 Relational operators

Three examples where relational operators are used

1. IF guess = computer THEN
2. IF guess < computer THEN
3. WHILE guess = computer

Activity 1

Write the following lines of algorithms as PHP code

| Algorithm | PHP Code |
|--|----------|
| IF guess = computer THEN Print well done ENDIF | |
| IF guess < computer THEN Print Too low ELSE Print Too high ENDIF | |
| WHILE guess <> computer count = count + 1 Enter guess ENDWHILE | |

Logical operators

The **logical operators** are AND, OR and NOT. They allow two or more boolean expressions to be combined to create logical expressions.

AND – Both parts must be true for the whole to be true

| Example expressions | Interpretation | Gender | Age | Logic |
|-------------------------------|---|--------|-----|-------|
| Age >= 18 AND Gender = “male” | This expression would be true, for a male aged 18 or over | Male | 19 | True |
| | | Male | 18 | True |
| | | Male | 17 | False |
| | | Female | 19 | False |

OR – Either or both parts can be true for the whole to be true

| Example expressions | Interpretation | Value | End | Logic |
|--------------------------|--|-------|-----|-------|
| Value <> 10 OR End = "T" | This expression would be true, when <i>value</i> is not equal to 10, or End is equal to "T". | 9 | T | True |
| | | 9 | F | True |
| | | 10 | T | True |
| | | 10 | F | False |

NOT – The opposite value is returned

| Example expressions | Interpretation | Finished | Logic |
|---------------------|---|---------------|---------------|
| NOT(Finished) | Finished is a Boolean variable. The Boolean value in Finished would be reversed | True False | False True |

The syntax of logical operators in some common languages include:

| Logical operator | PHP | VB | Java | Pascal | C |
|------------------|-----|-----|------|--------|----|
| AND | && | AND | && | AND | && |
| OR | | OR | | OR | |
| NOT | ! | NOT | ! | NOT | ! |

Activity 2

Write the following lines of algorithms as PHP code

| Algorithm | PHP Code |
|-------------------------------------|----------|
| IF Value <> 10 OR End = "T" THEN | |
| IF Value <> 10 AND End = false THEN | |
| WHILE NOT(Finished) | |

Activity 3

- List the 3 types of logical operators.
- List all the relational operators.
- Complete the following tables

| Example expressions | Sex | Age | Logic |
|------------------------|--------|-----|-------|
| Sex = Male OR Age > 16 | Male | 18 | |
| | Male | 14 | |
| | Female | 18 | |
| | Female | 14 | |

| Example expressions | Age | Mark | Logic |
|-----------------------|-----|------|-------|
| Age < 20 AND Mark > 0 | 15 | 0 | |
| | 25 | 50 | |
| | 15 | 50 | |
| | 25 | 0 | |