**MANTIKSAL İFADELER**

**MANTIKSAL İFADELER**

İçerikleri (değer, nesne, …) bir koşula göre kontrol eden ifadelerdir. İfadenin sonucu doğru veya yanlış olabilir. Koşul sonucu direkt olarak bir veri nesnesine atanamaz. Kullanılan ifadeler aşağıdadır.

**Karşılaştırma Operatörleri**

Değerler arasında karşılaştırma yapmak için kullanılan mantıksal ifadelerdir. Kullanım şekli aşağıdaki gibidir.

icerik1 karsilastirma\_operatoru icerik2

Bütün veri tipleri için karşılaştırma operatörleri aşağıdaki tabloda verilmiştir.

|  |  |  |
| --- | --- | --- |
| **Operatör** | **Kontrol** | **Tanım** |
| =, EQ | Equal  Eşit | Karşılaştırılan iki değerin eşitliğini kontrol eder. |
| <>, NE | Not Equal  Eşit değil | Karşılaştırılan iki değerin eşit olmadığını kontrol eder. |
| <, LT | Less Than  Küçük | Birinci değerin, ikinci değerden küçük olup olmadığını kontrol eder. |
| >, GT | Greater Than  Büyük | Birinci değerin, ikinci değerden büyük olup olmadığını kontrol eder. |
| <=, LE | Less Equal  Küçük Eşit | Birinci değerin, ikinci değerden küçük veya eşit olup olmadığını kontrol eder. |
| >=, GE | Greater Equal  Büyük Eşit | Birinci değerin, ikinci değerden büyük veya eşit olup olmadığını kontrol eder. |

Tablo ??? Bütün veri tipleri için karşılaştırma operatörü

Karakter-benzeri veri nesneleri için karşılaştırma operatörleri aşağıdaki tabloda verilmiştir.

|  |  |  |
| --- | --- | --- |
| **Operatör** | **Kontrol** | **Tanım** |
| CO | Contains Only  İçeriyorsa | İlk veri nesnesindeki tüm karakterler, ikinci veri nesnesi içerisinde bulunuyor ise doğru olur. Küçük, büyük harf duyarlıdır. |
| CN | Contains Not Only  İçermiyorsa | İlk veri nesnesindeki tüm karakterler, ikinci veri nesnesi içerisinde bulunmuyor ise doğru olur. Küçük, büyük harf duyarlıdır. |
| CA | Contains Any  Karakter içeriyor ise | İkinci veri nesnesi karakterlerinden herhangi birisi, birinci veri nesnesi içerisinde bulunuyor ise doğru olur. Küçük, büyük harf duyarlıdır. |
| NA | Contains Not Any  Karakter içermiyor ise | İkinci veri nesnesi karakterlerinden herhangi birisi, birinci veri nesnesi içerisinde bulunmuyor ise doğru olur. Küçük, büyük harf duyarlıdır. |
| CS | Contains String  Dizi içeriyor ise | İlk veri nesnesi içerisinde, ikinci veri nesnesi bulunuyor ise doğru olur. Küçük, büyük harf duyarlı değildir. |
| NS | Contains No String  Dizi içeriyormiyor ise | İlk veri nesnesi içerisinde, ikinci veri nesnesi bulunmuyor ise doğru olur. Küçük, büyük harf duyarlı değildir. |
| CP | Covers Pattern  Şablon içeriyor mu | İlk veri nesnesi içerisinde, ikinci veri nesnesinde belirtilen joker karakterlere sahip kalıp bulunuyor ise doğru olur. Küçük, büyük harf duyarlı değildir. |
| NP | No Pattern  Şablon içermiyor mu | İlk veri nesnesi içerisinde, ikinci veri nesnesinde belirtilen joker karakterlere sahip kalıp bulunmuyor ise doğru olur. Küçük, büyük harf duyarlı değildir. |

Tablo ??? Karakter benzeri veri nesneleri için karşılaştırma operatörleri

Bayt benzeri nesneleri karşılaştırma için aşağıdaki operatörler kullanılır.

· BYTE-CO

· BYTE-CN

· BYTE-CA

· BYTE-NA

· BYTE-CS

· BYTE-NS

Bit deseni için karşılaştırma operatörleri

· O

· Z

· M

**BETWEEN**

Veri nesnesi değerinin belirtilen aralıkta olup olmadığı kontrolünü yapar.

icerik [NOT] BETWEEN icerik1 AND icerik2

**IS ASSIGNED**

Alan sembolünün bir nesneye atanıp, atanmadığını kontrol eder.

<fieldsymbols> IS [NOT] ASSIGNED

**IS BOUND**

Referansın değişkenin geçerli bir referans içerip içermediğini kontrol eder.

referans IS [NOT] BOUND

**IS INITIAL**

Bir veri nesnesinin değerinin tanımlı olduğu tipteki başlangıç değerine sahip olup olmadığını kontrol eder. Bu değer DATA ifadesinde VALUE kelimesi kullanarak tanımlanan değer değildir.

icerik IS [NOT] INITIAL

**IS SUPPLIED**

Bir prosedüre aktarılan parametrenin doluluğunu veya gerekliliğini kontrol eder.

para IS [NOT] SUPPLIED

**IN**

Veri nesnesinin değeri seçim tablosundaki değerler veya aralıklarda mı kontrolunu yapar.

icerik [NOT] IN seçim\_tablosu

**MANTIKSAL İFADE ÖRNEKLERİ**

1. İki sayısal değişkenin değerlerini karşılaştıran örnek.

DATA: gv\_degisken1 TYPE i VALUE 1,  
gv\_degisken2 TYPE i VALUE 2.  
IF gv\_degisken1 <> gv\_degisken2.  
WRITE ‘Birinci değişken ikinci değişkene eşit değil!’.  
ENDIF.  
gv\_degisken1 = 2.  
IF gv\_degisken1 = gv\_degisken2.  
WRITE / ‘Birinci değişken ikinci değişkene eşit.’.  
ENDIF.  
gv\_degisken1 = 3.  
IF gv\_degisken1 GT gv\_degisken2.  
WRITE / ‘Birinci değişken ikinci değişkenden büyük.’.  
ENDIF.

http://iuyanik.com/abaptr/wp-content/uploads/2013/11/img_527543aa928ec.png

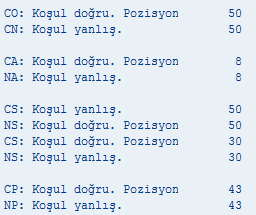
2. İki d (tarih) tipine sahip değişkenin değerlerini karşılaştıran örnek.

DATA: gv\_degisken1 TYPE d,  
gv\_degisken2 TYPE d.  
gv\_degisken1 = sy-datum.  
gv\_degisken2 = gv\_degisken1 – 1.  
IF gv\_degisken1 <> gv\_degisken2.  
WRITE ‘Birinci değişken ikinci değişkene eşit değil!’.  
ENDIF.  
gv\_degisken2 = sy-datum.  
IF gv\_degisken1 = gv\_degisken2.  
WRITE / ‘Birinci değişken ikinci değişkene eşit.’.  
ENDIF.  
gv\_degisken1 = gv\_degisken1 + 1.  
IF gv\_degisken1 GT gv\_degisken2.  
WRITE / ‘Birinci değişken ikinci değişkenden büyük.’.  
ENDIF.

http://iuyanik.com/abaptr/wp-content/uploads/2013/11/img_527543b9d6da2.png

3. Karakter-benzeri veri tipleri için karşılaştırma operatörü kullanımını gösteren örnek.

DATA: gv\_degisken1 TYPE c LENGTH 50,  
gv\_degisken2 TYPE c LENGTH 50.  
gv\_degisken1 = ‘bmik’.  
gv\_degisken2 = ‘Değerler bu metin içerisinde kontrol edilecek.’.  
IF gv\_degisken1 CO gv\_degisken2.  
WRITE: ‘CO: Koşul doğru.’, ‘Pozisyon’, 30 sy-fdpos.  
ELSE.  
WRITE: ‘CO: Koşul yanlış.’, 30 sy-fdpos.  
ENDIF.  
IF gv\_degisken1 CN gv\_degisken2.  
WRITE: / ‘CN: Koşul doğru.’, ‘Pozisyon’, 30 sy-fdpos.  
ELSE.  
WRITE: / ‘CN: Koşul yanlış.’, 30 sy-fdpos.  
ENDIF.  
SKIP.  
gv\_degisken1 = ‘Değerler bu metin içerisinde kontrol edilecek.’.  
gv\_degisken2 = ‘bmik’.  
IF gv\_degisken1 CA gv\_degisken2.  
WRITE: / ‘CA: Koşul doğru.’, ‘Pozisyon’, 30 sy-fdpos.  
ELSE.  
WRITE: / ‘CA: Koşul yanlış.’, 30 sy-fdpos.  
ENDIF.  
IF gv\_degisken1 NA gv\_degisken2.  
WRITE: / ‘NA: Koşul doğru.’, ‘Pozisyon’, 30 sy-fdpos.  
ELSE.  
WRITE: / ‘NA: Koşul yanlış.’, 30 sy-fdpos.  
ENDIF.  
SKIP.  
gv\_degisken1 = ‘Değerler bu metin içerisinde kontrol edilecek.’.  
gv\_degisken2 = ‘tONT’.  
IF gv\_degisken1 CS gv\_degisken2.  
WRITE: / ‘CS: Koşul doğru.’, ‘Pozisyon’, 30 sy-fdpos.  
ELSE.  
WRITE: / ‘CS: Koşul yanlış.’, 30 sy-fdpos.  
ENDIF.  
IF gv\_degisken1 NS gv\_degisken2.  
WRITE: / ‘NS: Koşul doğru.’, ‘Pozisyon’, 30 sy-fdpos.  
ELSE.  
WRITE: / ‘NS: Koşul yanlış.’, 30 sy-fdpos.  
ENDIF.  
gv\_degisken1 = ‘Değerler bu metin içerisinde kontrol edilecek.’.  
gv\_degisken2 = ‘ONTR’.  
IF gv\_degisken1 CS gv\_degisken2.  
WRITE: / ‘CS: Koşul doğru.’, ‘Pozisyon’, 30 sy-fdpos.  
ELSE.  
WRITE: / ‘CS: Koşul yanlış.’, 30 sy-fdpos..  
ENDIF.  
IF gv\_degisken1 NS gv\_degisken2.  
WRITE: / ‘NS: Koşul doğru.’, ‘Pozisyon’, 30 sy-fdpos.  
ELSE.  
WRITE: / ‘NS: Koşul yanlış.’, 30 sy-fdpos.  
ENDIF.  
SKIP.  
gv\_degisken2 = ‘\*EK.’.  
IF gv\_degisken1 CP gv\_degisken2.  
WRITE: / ‘CP: Koşul doğru.’, ‘Pozisyon’, 30 sy-fdpos.  
ELSE.  
WRITE: / ‘CP: Koşul yanlış.’, 30 sy-fdpos.  
ENDIF.  
IF gv\_degisken1 NP gv\_degisken2.  
WRITE: / ‘NP: Koşul doğru.’, ‘Pozisyon’, 30 sy-fdpos.  
ELSE.  
WRITE: / ‘NP: Koşul yanlış.’, 30 sy-fdpos.  
ENDIF.



4. BETWEEN mantıksal ifadesi ile sayısal değişkenin belirtilen aralıkta olup olmadığını kontrol eden örnek.

DATA: gv\_degisken1 TYPE i VALUE 250,  
gv\_degisken2 TYPE i VALUE 100,  
gv\_degisken3 TYPE i VALUE 999.  
IF gv\_degisken1 BETWEEN gv\_degisken2 AND gv\_degisken3.  
WRITE ‘Değişken aralık içerisinde.’.  
ENDIF.  
gv\_degisken1 = 5000.  
IF gv\_degisken1 BETWEEN gv\_degisken2 AND gv\_degisken3.  
WRITE / ‘Değişken aralık içerisinde.’.  
ELSE.  
WRITE / ‘Değişken aralık içerisinde değil!’.  
ENDIF.

http://iuyanik.com/abaptr/wp-content/uploads/2013/11/img_527543d00dded.png

5. Alan sembolüne değer atanıp atanmadığını IS ASSIGNED mantıksal ifadesi ile kontrol eden örnek.

DATA gv\_degisken1 TYPE c LENGTH 40.  
FIELD-SYMBOLS <fs1> TYPE c.  
gv\_degisken1 = ‘Alan sembolüne atanan metin.’.  
IF <fs1> IS NOT ASSIGNED.  
WRITE ‘<fs1> alan sembolüne değer ataması yapılıyor.’.  
ASSIGN gv\_degisken1 TO <fs1>.  
WRITE / <fs1>.  
ENDIF.

http://iuyanik.com/abaptr/wp-content/uploads/2013/11/img_527543d7d8b2a.png

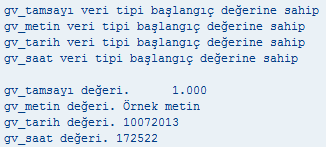
6. Referans değişkenin geçerli bir referans içerip içermediğini IS BOUND mantıksal ifadesi ile kontrol eden örnek.

DATA: gs\_dref  TYPE REF TO data,  
gt\_spfli TYPE TABLE OF spfli.  
IF gs\_dref IS NOT BOUND.  
WRITE ‘Referans geçerli bir değişene sahip değil!’.  
ENDIF.  
SELECT \* FROM spfli INTO CORRESPONDING FIELDS OF TABLE gt\_spfli.  
READ TABLE gt\_spfli REFERENCE INTO gs\_dref INDEX 1.  
IF gs\_dref IS BOUND.  
WRITE / ‘Referans geçerli bir değişene sahip.’.  
ENDIF.

http://iuyanik.com/abaptr/wp-content/uploads/2013/11/img_527543e596046.png

7. Farklı veri tiplerine sahip değişkenlerin başlangıç değerine sahip olup olmadığını INITIAL mantıksal ifadesi ile kontrol eden örnek.

DATA: gv\_tamsayi TYPE i,  
gv\_metin   TYPE c LENGTH 20,  
gv\_tarih   TYPE d,  
gv\_saat    TYPE t.  
IF gv\_tamsayi IS INITIAL.  
WRITE ‘gv\_tamsayı veri tipi başlangıç değerine sahip’.  
ENDIF.  
IF gv\_metin IS INITIAL.  
WRITE / ‘gv\_metin veri tipi başlangıç değerine sahip’.  
ENDIF.  
IF gv\_tarih IS INITIAL.  
WRITE / ‘gv\_tarih veri tipi başlangıç değerine sahip’.  
ENDIF.  
IF gv\_saat IS INITIAL.  
WRITE / ‘gv\_saat veri tipi başlangıç değerine sahip’.  
ENDIF.  
SKIP.  
gv\_tamsayi = 1000.  
gv\_metin   = ‘Örnek metin’.  
gv\_tarih   = sy-datum.  
gv\_saat    = sy-uzeit.  
IF gv\_tamsayi IS NOT INITIAL.  
WRITE: / ‘gv\_tamsayı değeri.’, gv\_tamsayi.  
ENDIF.  
IF gv\_metin IS NOT INITIAL.  
WRITE: / ‘gv\_metin değeri.’, gv\_metin.  
ENDIF.  
IF gv\_tarih IS NOT INITIAL.  
WRITE: / ‘gv\_tarih değeri.’, gv\_tarih.  
ENDIF.  
IF gv\_saat IS NOT INITIAL.  
WRITE: / ‘gv\_saat değeri.’, gv\_saat.  
ENDIF.



8. Sınıfa bağlı yöntemi önce parametre kullanmayarak, daha sonra parametre kullanarak çalıştıran ve parametre kontrolünü IS SUPPLIED mantıksal ifadesi ile kontrol eden örnek.

CLASS cls1 DEFINITION.  
PUBLIC SECTION.  
CLASS-METHODS veri\_bul IMPORTING aranan\_metin TYPE c OPTIONAL.  
ENDCLASS.  
DATA cls TYPE REF TO cls1.  
CLASS cls1 IMPLEMENTATION.  
METHOD veri\_bul.  
IF aranan\_metin IS SUPPLIED.  
WRITE / ‘Yöntem aranan metin değeri belirtilerek çağrıldı.’.  
ELSE.  
WRITE / ‘Yöntem aranan metin değeri belirtilmeden çağrıldı!’.  
ENDIF.  
ENDMETHOD.  
ENDCLASS.  
START-OF-SELECTION.  
CREATE OBJECT cls.  
CALL METHOD cls->veri\_bul.  
CALL METHOD cls->veri\_bul(  
EXPORTING  
aranan\_metin = ‘metin’ ).

http://iuyanik.com/abaptr/wp-content/uploads/2013/11/img_527543f7a4aac.png

9. gv\_carrid değişkenine atanan ‘AF’ isimli uçuşu tanımlayıcısının, seçim tablosuna girilen değer(ler)/aralık(lar) içerisinde kontrol örnek.

TABLES spfli.  
DATA gv\_carrid TYPE spfli-carrid value ‘AF’.  
SELECT-OPTIONS: so\_carri FOR spfli-carrid OBLIGATORY.  
IF gv\_carrid IN so\_carri.  
WRITE: / ‘Değişken değeri seçim tablosunda bulundu.’, gv\_carrid.  
ELSE.  
WRITE: / ‘Değişken değeri seçim tablosunda bulunamadı!’, gv\_carrid.  
ENDIF.

http://iuyanik.com/abaptr/wp-content/uploads/2013/11/img_527544043b146.png

http://iuyanik.com/abaptr/wp-content/uploads/2013/11/img_5275440c31891.png