

ABAP SYSTEM STRUCTURE KOMUTLARI ¹

1. **SY-SUBRC** : Son yapılan işlemin başarıyla tamamlanıp tamamlanmadığını belirtir. Değer 0 ise işlem başarılı, 4 ise başarısızdır.
2. **SY-MSGID** : Mesaj kimliği (Message ID) - Hata veya uyarı mesajlarının tanımlandığı mesaj kataloğu ID'si.
3. **SY-MSGTY** : Mesaj türü (Message Type) - Mesajın türünü belirten değer (S, W, E, I gibi).
4. **SY-MSGNO** : Mesaj numarası (Message Number) - Mesajın numarası.
5. **SY-INDEX** : Bir döngü içinde kaçınıcı iterasyonun gerçekleştiğini belirtir.
6. **SY-LOOPC** : Bir döngü içindeki geçerli durumu belirlemek için kullanılır. **DO**, **WHILE**, **LOOP**, veya **LOOP AT** gibi döngülerdeki kontrol ifadelerinde kullanılabilir.
7. **SY-CPROG** : Çağırılan programın adını içerir.
8. **SY-REPID** : Geçerli ABAP programının adını içerir.
9. **SY-DYPROG** : Geçerli ekran programının adını içerir. Geçerli ekran programını tanımlamak ve üzerinde işlemler yapmak için kullanılır.
10. **SY-DATUM** : Programın çalıştığı anki tarih bilgisini içerir. Bu tarih, SAP sistem saatine ve tarihine göre belirlenir.
11. **SY-UZEIT** : Bu alan, SAP sistemi çalıştığı sunucunun saatini tutar. Bu alanın kullanımı, programın ihtiyacına bağlıdır. Programın veritabanıyla ilgili işlem yapmadığı durumlarda, global zaman kullanmak daha doğru olabilir.
12. **SY-TCODE** : kullanıcının şu anda hangi SAP transaksyon kodu (transaction code) ekranında bulunduğunu içerir. Bu, programların çalışma zamanında kullanıcının hangi ekran üzerinde işlem yaptığını belirlemek için kullanılır.
13. **SY-LSIND** bir döngü içinde bulunulan anın döngünün kaçınıcı iterasyonu olduğunu içerir. "Loop index" olarak da adlandırılabilir.
14. **SY-PFKEY** : bir ekran üzerindeki kullanıcının basılan fonksiyon tuşunu (PF tuşu) içerir. PF tuşları, kullanıcıya belirli işlevleri gerçekleştirme yeteneği sağlayan işlev

tuşlarıdır. SAP ekranı üzerindeki bir işlem sırasında hangi PF tuşunun basıldığını takip etmek için kullanılır.

15. **SY-BATCH** : bir SAP programının işlem modunu belirtir. Eğer bir program arka planda (batch) çalışıyorsa, **SY-BATCH** değişkeni **X** değerini alır; ön planda (foreground) çalışıyorsa, değeri boştur (' ').
16. **SY-DIALOG** : bir SAP programının hangi modda çalıştığını belirler. Eğer bir program kullanıcı etkileşimine açık bir ekran üzerinde çalışıyorsa (dialog mode), **SY-DIALOG** değişkeni **X** değerini alır. Eğer bir program bir işlem içinde, arka planda (background mode) veya başka bir sistem çağrısında çalışıyorsa, **SY-DIALOG** değişkeni boş (' ') değerini alır. Bir programın kullanıcı etkileşimine ihtiyaç duyup duymadığını kontrol etmek için **SY-DIALOG** değişkeni kullanılabilir.
17. **SY-MANDT** : SAP sistem mandant numarasını içerir. Mandant, bir SAP sistemi içinde bağımsız bir iş ortamını temsil eder ve bu iş ortamında çalışan verileri ve yapılandırmayı izole eder. **SY-MANDT** değişkeni, program içindeki mantıksal kontrol ve yönlendirmelerde kullanılabilir. SAP sistemi içinde birden fazla mandant bulunuyorsa ve bir programın hangi mandant üzerinde çalıştığını kontrol etmek gerekiyorsa, **SY-MANDT** kullanılabilir.

OpenSQL SORGULARI 2

SUM,AVG,MAX,MIN,COUNT Sorgu Cümlecikleri

Marka	Model	Yıl	Satış Adedi
Ford	Focus	2022	120
Toyota	Corolla	2022	150
Honda	Civic	2022	100
Ford	Mustang	2022	80
Toyota	Camry	2022	90
Honda	Accord	2022	70
Ford	F-150	2022	200
Toyota	RAV4	2022	180
Honda	CR-V	2022	150

Yukarıda yer alan tablodaki verilerin ‘OpenSQL’ ortamında istatistiksel verilerini toplamak için *"SUM", "AVG", "MAX", "MIN", "COUNT"* komutlarını kullanırız.

Toplam satış adedi için :

```
SQL : SELECT SUM("Satış Adedi") FROM SATIS .
```

ABAP: DATA lv_sum TYPE i.

```
SELECT SUM( "Satış Adedi" ) INTO lv_sum FROM ZXX_TABLE.
```

Ortalama satış adedi için :

```
SQL : SELECT AVG("Satış Adedi") FROM SATIS .
```

ABAP: DATA lv_avg TYPE i.

```
SELECT AVG( "Satış Adedi" ) INTO lv_avg FROM ZXX_TABLE.
```

En yüksek satış adedi için :

```
SQL :      SELECT MAX("Satış Adedi") FROM SATIS .
```

```
ABAP :      DATA lv_max TYPE i.  
            SELECT MAX( "Satış Adedi" ) INTO lv_max FROM ZXX_TABLE.
```

En düşük satış adedi için :

```
SQL :      SELECT MIN("Satış Adedi") FROM SATIS .
```

```
ABAP :      DATA lv_min TYPE i.  
            SELECT MIN( "Satış Adedi" ) INTO lv_min FROM ZXX_TABLE.
```

Satış adedi sayısı için :

```
SQL :      SELECT COUNT(*) FROM SATIS.
```

```
ABAP :      DATA lv_count TYPE i.  
            SELECT COUNT( "Satış Adedi" ) INTO lv_count FROM ZXX_TABLE.
```

OPEN SQL

INSERT, UPDATE, DELETE, MODIFY KOMUTLARI ³

INSERT :

```
DATA: lv_malzeme TYPE matnr VALUE 'MAT123',  
      lv_tanim TYPE string VALUE 'Malzeme Adı',  
      lt_deger TYPE TABLE OF sflight.
```

```
INSERT into sflight  
VALUES ( lv_malzeme, lv_tanim ).
```

```
IF      sy-subrc = 0.  
    WRITE: / 'Insert işlemi tamamlandı'.  
ELSE.  
    WRITE: / 'İşlem esnasında hata oluştu: bkz. ', sy-msgid, sy-msgty, sy-  
            msgno, sy-msgty.  
ENDIF.
```

UPDATE:

DATA: lv_malzeme TYPE matnr VALUE 'MAT123',
 lv_tanim_yeni TYPE string VALUE 'Güncellenmiş Malzeme Adı'.

UPDATE sflight

SET description = lv_tanim_yeni
WHERE matnr = lv_malzeme.

IF *sy-subrc* = 0.

WRITE: / 'İnsert işlemi tamamlandı'.

ELSE.

WRITE: / 'İşlem esnasında hata oluştu: bkz. ', sy-msgid, sy-msgty, sy-
msgno, sy-msgty.

ENDIF.

DELETE:

DATA: lv_malzeme TYPE matnr VALUE 'MAT123'.

DELETE FROM sflight
 WHERE matnr = lv_malzeme.

IF *sy-subrc* = 0.

WRITE: / 'İşlem tamamlandı'.

ELSE.

WRITE: / 'İşlem esnasında hata oluştu: bkz. ', sy-msgid, sy-msgty, sy-
msgno, sy-msgty.

ENDIF.

MODIFY:

```
DATA:          lv_malzeme  TYPE matnr VALUE 'MAT123',  
          lv_tanim_yeni TYPE string VALUE 'Güncellenmiş Malzeme Adı'.
```

MODIFY sflight

```
FROM          lv_malzeme  
SET           tanim = lv_tanim_yeni.
```

IF *sy-subrc* = 0.

WRITE: / 'Insert işlemi tamamlandı'.

ELSE.

WRITE: / 'İşlem esnasında hata oluştu: bkz. ', sy-msgid, sy-msgty, sy-
msgno, sy-msgty.

ENDIF.

KAYNAKÇA

- 1 - <https://answers.sap.com/>
- 2 - <https://www.yazilimkodlama.com/>
- 3 - <https://www.mssqltips.com/>