

İçindekiler

SYST STRUCTURE ALANLARI.....	2
OPEN SQL TEMEL HESAPLAMA FONKSİYONLARI .	4
OPEN SQL VERİ TABANI İŞLEMLERİ	6

SYST STRUCTURE ALANLARI

SAP abap dilinde SYST yapıları kullanıcıların sistem ile alakalı bilgilere ulaşmalarını sağlamaktadır. Ulaşıla bilgiler aracılığıyla çalıştırılan programlarda bu bilgiler kullanılmaktadır. Daha ayrıntılı bilgi için <https://help.sap.com/docs/>¹ adresinde SYST yapıları listelenmektedir. Bu listede yer alan 20 adet SYST yapısının açıklaması aşağıda yer almaktadır.

SY-DATUM:

Sy-datum yapısı kullanıcıya bulunduğu sunucunun tarih bilgisi Gün, Ay, Yıl formatında vermektedir.

SY-SUBRC:

Kendisinden önce dönen ifadenin başarılı bir şekilde çalışıp çalışmadığını ifade etmektedir. Eğer sy-subrc değeri 0 dönüyorsa bu işlemin başarılı bir şekilde çalıştığı anlamına gelmektedir.

SY-TABIX:

İç tablo döngüsünün sayısını temsil etmektedir.

SY-ABCDE:

Latin alfabesini temsil etmektedir. Latin alfabesine doğrudan erişimi sağlamaktadır.

SY-UZEIT:

Sistemin saatini saat, dakika ve saniye cinsinden ifade etmektedir.

¹ https://help.sap.com/doc/saphelp_nw73ehp1/7.31.19/en-US/7b/fb96c8882811d295a90000e8353423/content.htm?no_cache=true: ABAP dilinde yer alan sistem alanlarının ve yapılarının listesi.

SY-REPID:

Kullanıcıya ilgili programın isim bilgisini vermektedir.

SY-LANGU:

Kullanıcının sistemde aktif olan dil seçeneğini temsil etmektedir. Eğer kullanıcı dil seçeneğini İngilizce seçtiyse sy-langu değeri EN olarak güncellenmektedir.

SY-INDEX:

Kullanılan programda yer alan döngünün indexini ifade etmektedir.

SY-UNAME:

Kullanıcının SAP sistemine giriş yaptığı kullanıcı adını temsil etmektedir. Örneğin Bekir isimli bir kişinin kullanıcı adı BAsim2023 olsun. Bu kullanıcı sisteme giriş yaptığında sy-uname değeri BAsim2023 olarak güncellenmektedir.

SY-BATCH:

Kullanıcı bir programı arka planda çalıştırdığında ilgili programın sy-batch değişkeni 'X' değerini almaktadır.

SY-HOST:

Kullanıcının çalıştırdığı programın bağlı olduğu sunucunun adını vermektedir.

SY-TCODE:

SAP sisteminde belli işlemleri yerine getirmek için transaction kodlar kullanılmaktadır. Belli bir uygulamanın içindeyken sy-tcode değeri o uygulamaya gitmek için kullanılan tcode değerini temsil etmektedir.

SY-MSGID:

SAP sisteminde mesaj sınıfları yer almaktadır. Sy-msgid yapısı ise kullanıcının gönderdiği son mesajın mesaj sınıfını ifade etmektedir.

SY-DBNAME:

SAP sisteminde kullanıcıların yaptıkları işlemleri veri tabanına kaydedebilmektedirler. Sy-database ise mevcut kullanılan veri tabanını temsil etmektedir.

SY-DBSYS:

Merkezi veri tabanı sistemini ifade etmektedir. Bu değer default olarak HANA veri tabanıdır.

OPEN SQL TEMEL HESAPLAMA FONKSİYONLARI

SAP abap dilinde open sql hesaplama komutları kullanarak temel matematik işlemleri gerçekleştirilmektedir. İşte bu işlemlerden bazıları:

SUM:

Sum Open Sql komutu, bir sütunda yer alan değerlerin toplamını vermektedir. Daha iyi anlaşılması için abap dilinde bir kod örneği aşağıda verilmiştir.

AVG:

Avg Open Sql komutu adından da anlaşılacağı üzere bir ortalama alma işlemidir.

MAX:

Max Open Sql komutu bir alandaki en büyük değeri ifade etmektedir.

MIN:

Min Open Sql komutu ise max komutunun aksine alandaki en küçük değeri ifade etmektedir.

COUNT:

Count Open Sql komutu belirli bir koşul çerçevesindeki satır sayısını ifade etmektedir.

Open sql komutlarını daha iyi anlamak için aşağıda bir örneklendirme yapılmıştır.

Personel Maaş tablosu aşağıdaki gibidir:

ZBA_Flper_T

perID	perAd	perSoyad	perMaas
1	Ali	Erciyes	15000
2	Veli	Kitap	17500
3	Füsun	Koç	16500
4	İbrahim	Ocak	18000
5	Ferhunde	Çarşı	20000

REPORT ZBA_CALC_Prog

DATA: lv_sum_maas TYPE i,
 lv_avg_maas TYPE i,
 lv_min_maas TYPE i,
 lv_max_maas TYPE i,
 lv_count_maas TYPE i.

```
SELECT SUM ( perMaas ) INTO lv_sum_maas
      AVG ( perMaas ) INTO lv_avg_maas
      MIN ( perMaas ) INTO lv_min_maas
      MAX ( perMaas ) INTO lv_max_maas
      COUNT ( * ) INTO lv_count_maas
FROM ZBA_Flper_T.
```

```
WRITE : / 'Toplam Personel Maaşı : ', lv_sum_maas,
        / 'Ortalama Personel Maaşı : ', lv_avg_maas,
        / 'En Düşük Personel Maaşı : ', lv_min_maas,
        / 'En Yüksek Personel Maaşı : ', lv_max_maas,
        / 'Toplam Maaş Sayısı : ', lv_count_maas.
```

OPEN SQL VERİ TABANI İŞLEMLERİ

SAP abap dilinde temel sql veri tabanı işlemleri ile tablolara yeni veriler eklenebilir, eklenen veriler düzenlenebilir, güncellenebilir ve silinebilmektedir. İşte temel open sql veri tabanı işlemleri:

INSERT:

Kullanıcının veri tabanına yeni bir kayıt eklemesini sağlamaktadır. Bu yeni kayıt eklenirken işlemin sağlıklı bir şekilde gerçekleşmesi için veri tabanında yer alan obligatory sütunların mutlaka doldurulmuş bir şekilde gönderilmesi gerekmektedir.

UPDATE:

Kullanıcı veri tabanında yer alan ve daha önceden girilmiş bir verinin değerlerini güncellemek için kullanılmaktadır.

DELETE:

Veri tabanındaki kayıtların silinmesi için kullanılmaktadır.

MODIFY:

Modify komutu kullanıcıya, veri tabanındaki belirli bir satırı güncelleme fırsatı sunmaktadır. Belirtilen satır eğer veri tabanında yoksa yeni girişi olarak da kullanılabilir. Farklı senaryolara göre insert ya da modify komutlarını kullanmak avantaj ya da dezavantaja dönüşebilmektedir.

Konunun daha iyi anlaşılması için aşağıda SAP abap dilinde örnek bir kod uygulaması yer almaktadır.

Personel Maaş tablosu aşağıdaki gibidir:

ZBA_FIper_T

perID	perAd	perSoyad	perMaas
1	Ali	Erciyes	15000
2	Veli	Kitap	17500
3	Fusun	Koç	16500
4	İbrahim	Ocak	18000
5	Ferhunde	Çarşı	20000

REPORT ZBA_DB_Prog

DATA: lt_per_data TYPE TABLE OF ZBA_FIper_T,
ls_per_data. TYPE ZBA_FIper_T

***** INSERT İŞLEMİ *****

ls_per_data-perID = 6.
ls_per_data-perAD = 'Zeynep'.
ls_per_data-perSOYAD = 'Zümra'.
ls_per_data-perMaas = 25000.

APPEND ls_per_data TO lt_per_data.

INSERT INTO ZBA_FIper_T FROM TABLE lt_per_data.

Veri tabanının yeni görüntüsü şu şekildedir:

ZBA_DB_T

perID	perAd	perSoyad	perMaas
1	Ali	Erciyes	15000
2	Veli	Kitap	17500
3	Fusun	Koç	16500
4	İbrahim	Ocak	18000
5	Ferhunde	Çarşı	20000
6	Zeynep	Zümra	25000

.....

*****UPDATE İŞLEMİ*****

*Güncellenecek kaydın verileri

ls_per_data-perID = 3.
ls_per_data-perMaas = 20000.

APPEND ls_per_data TO lt_per_data.

UPDATE ZBA_FIper_T SET perMaas = ls_per_data-perMaas
Where perID = ls_per_data-perID.

Veri tabanının yeni görüntüsü şu şekildedir:

ZBA_FIper_T

perID	perAd	perSoyad	perMaas
1	Ali	Erciyes	15000
2	Veli	Kitap	17500
3	Fusun	Koç	20000
4	İbrahim	Ocak	18000
5	Ferhunde	Çarşı	20000
6	Zeynep	Zümra	25000

.....

*****DELETE İŞLEMİ*****

*Silinecek kaydın verileri

ls_per_data-perID = 1.

APPEND ls_per_data TO lt_per_data.

DELETE FROM ZBA_FIper_T where perID = ls_per_data-perID.

Veri tabanının yeni görüntüsü şu şekildedir:

ZBA_FIper_T

perID	perAd	perSoyad	perMaas
2	Veli	Kitap	17500
3	Fusun	Koç	20000
4	İbrahim	Ocak	18000
5	Ferhunde	Çarşı	20000
6	Zeynep	Zümra	25000

.....

***** MODIFY İŞLEMİ *****

*Güncellenecek kaydın bilgileri

ls_per_data-perID = 5.

ls_per_data-perSOYAD = 'Çarşı Ocak'.

APPEND ls_per_data TO lt_per_data.

MODIFY ZBA_FIper_T

FROM ls_per_data

TRANSPORTING perSOYAD

where perID = ls_per_data-perID.

Veri tabanının yeni görüntüsü şu şekildedir:

ZBA_FIper_T

perID	perAd	perSoyad	perMaas
2	Veli	Kitap	17500
3	Fusun	Koç	20000
4	İbrahim	Ocak	18000
5	Ferhunde	Çarşı Ocak	20000
6	Zeynep	Zümra	25000

.....

Bekir Asım ÇELİK

Kullanılan Kaynaklar

https://help.sap.com/doc/saphelp_nw73ehp1/7.31.19/en-US/7b/fb96c8882811d295a90000e8353423/content.htm?no_cache=true