

1 - Hash Table Nedir?:

ABAP'te "hash table", SAP sistemi içinde kullanılan bir veri yapısıdır. ABAP (Advanced Business Application Programming), SAP'nin kendi geliştirdiği bir programlama dilidir ve iş uygulamaları için yaygın olarak kullanılır.

Hash table, veri saklamak için kullanılan bir yapıdır ve genellikle anahtar-değer çiftleri şeklinde veri saklar. Bu yapı, veriye hızlı bir şekilde erişim sağlamak için tasarlanmıştır. Bir hash table, veriye erişirken anahtarın hash fonksiyonu tarafından hesaplanan bir hash değerini kullanır. Bu, özellikle büyük veri kümeleriyle çalışırken veri arama sürelerini önemli ölçüde azaltabilir.

ABAP'te hash tablolar, iç tabloların bir türüdür ve genellikle performansı iyileştirmek amacıyla kullanılır. ABAP'te bir hash table oluştururken, tablonun türü olarak **HASHED TABLE** belirtilir. Örneğin, bir hash table tanımlamak için aşağıdaki gibi bir kod kullanılabilir:

```
1 DATA: lt_my_table TYPE HASHED TABLE OF my_structure WITH UNIQUE KEY table_key.
```

Bu örnekte, **lt_my_table** isimli bir hash table oluşturulmuştur. **my_structure** veri yapısına sahip ve **table_key** anahtarına göre benzersiz değerler içerir. Hash tablolar, özellikle tekrar eden veri erişim işlemlerinde ve büyük veri setlerinde oldukça etkilidir, çünkü anahtar değerine göre hızlı veri erişimi sağlarlar.

2 - View Nedir View İle Tablo Arasındaki Farklar Nelerdir

a) - View Nedir?

Bir "view" (görünüm), veritabanlarında kullanılan sanal bir tablodur. Gerçek veritabanı tablolarından veri çekerek oluşturulur ancak kendisi fiziksel olarak veri saklamaz. Temelde, bir veya daha fazla gerçek tablodaki verilerin filtrelenmiş, birleştirilmiş veya özelleştirilmiş bir görünümünü sunar. SQL sorguları kullanılarak tanımlanır ve genellikle veri güvenliği, erişim kolaylığı ve karmaşık sorguların basitleştirilmesi amacıyla kullanılır.

b) - View ile Tablo Arasındaki Farklar

Fiziksel Saklama:

- **Tablo:** Verileri fiziksel olarak saklar.
- **View:** Fiziksel veri saklamaz, mevcut tablolardaki verilerin sanal bir görünümünü sunar.

Tanımlama:

- **Tablo:** Sütunlar ve veri tipleri belirtilerek oluşturulur.
- **View:** SQL sorguları kullanılarak tanımlanır ve bir veya daha fazla tablonun verilerini temsil edebilir.

Güncellenebilirlik:

- **Tablo:** Doğrudan ekleme, güncelleme veya silme işlemlerine izin verir.
- **View:** Bazı durumlarda güncellenebilir, ancak bu genellikle sınırlıdır ve altındaki tablolara bağlıdır.

Performans:

- **Tablo:** Veri doğrudan tabloda saklandığı için erişim genellikle daha hızlıdır.
- **View:** Karmaşık sorguları içerebilir, bu da performansı etkileyebilir.

Güvenlik ve Erişim Kontrolü:

- **Tablo:** Erişim kontrolü genellikle daha az esneklik.
- **View:** Kullanıcıların sadece belirli verilere erişmesine izin vermek için kullanılabilir, bu da daha iyi veri güvenliği sağlar.

Kullanım Amaçları:

- **Tablo:** Veri depolama ve yönetiminin temelini oluşturur.
- **View:** Karmaşık sorguları basitleştirmek, veri güvenliğini artırmak ve kullanıcı dostu veri sunumları sağlamak için kullanılır.

3 - BAPI ve Batch Input Nedir?

a) - BAPI (Business Application Programming Interface), SAP sistemleri ile etkileşimde bulunan, standardize edilmiş bir dizi programlama arayüzüdür. BAPI'ler, SAP'nin çeşitli işlevlerine ve süreçlerine dış sistemlerden erişim sağlamak için kullanılır. Bu arayüzler, SAP'nin iş uygulamalarıyla veri alışverişi yapmak, işlemleri tetiklemek ve SAP iş süreçlerini uzaktan kontrol etmek için kullanılır.

Özellikleri:

- **Standartlaştırılmış:** BAPI'ler, SAP'nin geniş bir işlev yelpazesi için standartlaştırılmış arayüzler sunar.
- **Güvenli:** SAP'nin güvenlik ve veri bütünlüğü standartlarına uygun olarak tasarlanmıştır.
- **Uzaktan Erişim:** Dış sistemlerden SAP işlevlerine uzaktan erişim sağlar.

b) - Batch Input Nedir?

Batch Input, SAP'de veri aktarımı ve otomatik veri girişi için kullanılan bir tekniktir. Bu yöntem, kullanıcıların ekranda manuel olarak gireceği verileri otomatik olarak sisteme girer. Batch Input, özellikle büyük miktarda verinin SAP sistemine aktarılması gerektiğinde kullanışlıdır.

Özellikleri:

- **Veri Aktarımı:** Büyük miktarda veriyi güvenli bir şekilde SAP sistemine aktarır.
- **Otomasyon:** Manuel veri girişi işlemlerini otomatize eder.
- **Kayıt ve Oynatma:** Kullanıcı eylemlerini kaydedebilir ve bu eylemleri otomatik olarak tekrar oynatabilir.

5 - Job Tanımlama

SAP Easy Access ekranına gidin.

"SM36" komutunu kullanarak Job Definition ekranını açın.

Job Adı Girin: Bir job için anlamlı bir ad girin.

Job Sınıfı Seçin: İşin önceliğini belirleyin (A - Yüksek, B - Orta, C - Düşük).

Başlangıç Koşulları Belirleyin:

- **Hemen Başlatma:** İşi hemen çalıştırmak için.
- **Zamanlanmış Başlatma:** İşi belirli bir tarihte ve saatte çalıştırmak için.

a. Adımların Eklenmesi

"Step" sekmesine gidin.

Program/rapor adı girin: Çalıştırmak istediğiniz ABAP programının veya raporunun adını girin.

Parametreler (varsa): Program veya rapor için gerekli parametreleri girin.

b. İşin Kontrolü (Opsiyonel)

- **"Job Monitoring" (SM37):** Oluşturulan işin durumunu izlemek için kullanılır.
- **Loglar:** İşin yürütülmesi sırasında oluşan logları inceleyebilirsiniz.

c. İşi Kaydet ve Başlat

- **"Save" butonuna tıklayarak işi kaydedin.**
- Eğer işi zamanlamadıysanız, "Immediate start" seçeneğiyle hemen başlatabilirsiniz.

Önemli Notlar

- **Yetkilendirme:** İş oluşturacak kullanıcının gerekli yetkilere sahip olması gerekmektedir.
- **Hata Ayıklama:** İş başarısız olursa, hata ayıklama için logları kontrol edin.
- **Test Etmek Önemlidir:** Yeni bir işi canlı ortamda çalıştırmadan önce test ortamında denemek önemlidir.

6 - Günlük Job Oluşturma

SAP Easy Access Ekranına Giriş:

- SAP sisteminize giriş yapın.
- SAP Easy Access ekranına erişin.

Job Wizard'u Başlatın:

- Komut alanına "SM36" yazarak Job Definition ekranını açın.

Job Adı: "DailyReportJob" gibi anlamlı bir isim verin. (Örnekte RKDailyReportJob)

Job Sınıfı: Bu iş için uygun bir öncelik sınıfı seçin, örneğin "C" (Düşük).

Define Background Job

Start condition Step Job selection Own jobs Job wizard Job Repository

General Data

Job NameRKDailyReportJob

Job ClassC

StatusScheduled

Target

E-Mail Notification

Spool List Recipient

Başlatma Zamanı:

- "Start Condition" sekmesine gidin.
- "Date/Time" seçeneğini işaretleyin.
- İş her gün saat 07:00'de çalıştırmak için uygun tarih ve saati ayarlayın.

Start Time

Immediate

Date/Time

After Job

After Event

Operation Mode

Date/Time

Scheduled Start

Date

12/29/2023

Time

07:00:00

No Start After

Date

12/30/2023

Time

07:00:00

System Zone

UTC

☐ Use Alternative Time Zone

After Job

Operation Mode

After Event

☒ Periodic Job

✓ Check

Period values

Restrictions

✗

Adım ve Ekle Program Bilgileri:

- "Step" sekmesine gidin.
- "ABAP Program" seçeneğini belirleyin.
- Program Adı: Örneğin, "ZDAILY_REPORT" (var olan bir rapor adı) (Örnekte RKDailyReportJob).
- Variant: Eğer varsa, rapor için bir variant belirtin.

Start condition Step Job selection Own jobs Job wizard Job Repository

General Data

Create Step 1

User B_REMZIK

Program values

ABAP program External command External program

ABAP program

Name ZRMKDAILY_REPORT

Variant

İş Kaydet:

- Tüm bilgileri girdikten sonra "Save" butonuna basın.

Zamanlama:

- İş hemen başlatmak istemiyorsanız, "Save" ile işi kaydedin ve planlanan zamanda çalışmasını bekleyin.
- İş hemen başlatmak istiyorsanız, "Immediate start" seçeneğini işaretleyin ve "Save" butonuna basın.

İş Durumunu Kontrol Etme:

- İşin durumunu kontrol etmek için "SM37" komutunu kullanarak Job Monitoring ekranına gidin ve "DailyReportJob" (Örnekte RKDailyReportJob) işini aratın.