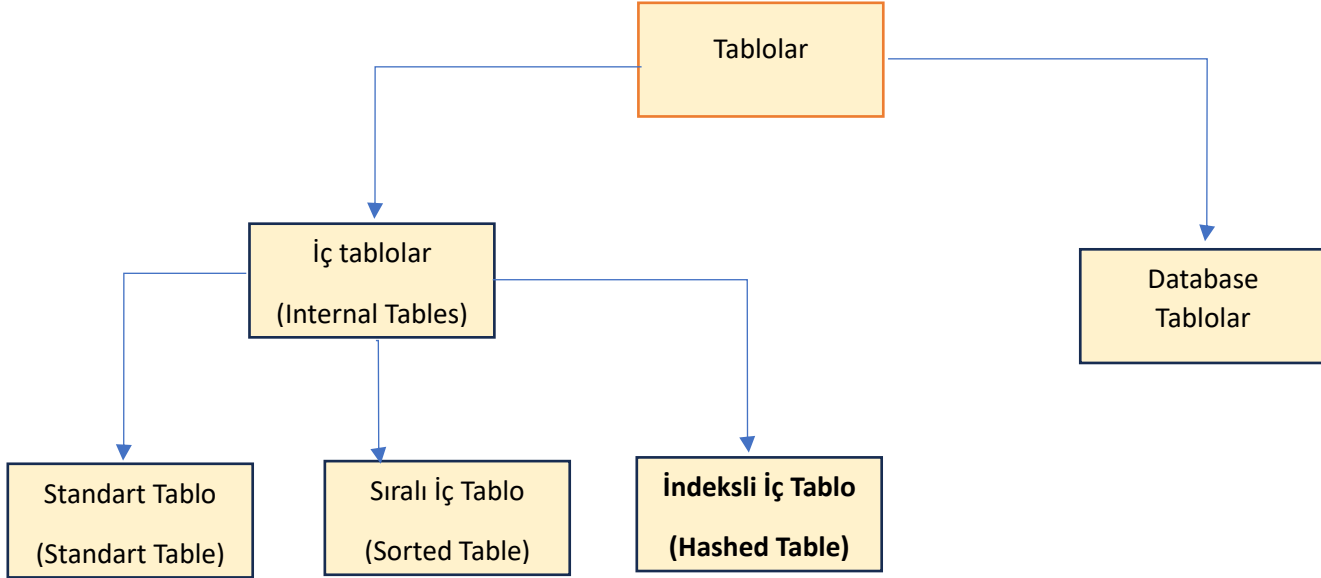


Hash Table Nedir ?

ABAP'ta tablolar genellikle iki ana kategoriye ayrılır: iç tablolar (internal tables) ve veritabanı tabloları (database tables). İki kategori, farklı kullanım amaçlarına ve işlevlere sahiptir.



İç Tablolar (Internal Tables):

İç tablolar, program içinde geçici olarak veri depolamak veya işlemek amacıyla kullanılır.

İç tablolar, programın çalışma süresi boyunca yaşar ve genellikle programın bir bölümünde tanımlanır ve kullanılır.

Veritabanı Tabloları (Database Tables):

Veritabanı tabloları, SAP veritabanında kalıcı veri depolamak için kullanılır.

Internal Tablolar Arasındaki Farklar:

1.Standart İç Tablolar (Standard Internal Tables):

Tanım:

- Tek boyutlu tablolardır.
- Her bir satırda bir kayıt içerirler.

Eriřim:

- Satır numarasına göre eriřim saęlarlar.
- Örneęin, **READ TABLE** komutu ile belirli bir satıra eriřilebilir.

Sıralama:

- Sıralı deęillerdir. Eklenen sırayla verileri içerirler.

Kullanım:

- Küçük veri setleri veya geçici veri depolama için kullanılır.

Komut :

DATA: lt_standard TYPE STANDART TABLE OF string.

2.Sıralı İç Tablolar (Sorted Internal Tables):**Tanım:**

- Standart iç tablolara benzer, ancak belirli bir anahtara göre sıralıdırlar.

Eriřim:

- Anahtar alanına göre hızlı eriřim saęlarlar.

READ TABLE veya **LOOP AT** ile anahtar deęeri ile eriřim yapılabilir.

Sıralama:

- Anahtar alanına göre sıralıdırlar.

Kullanım:

- Sıralama gereksinimi olan durumlar için kullanılır.

Komut :

DATA: lt_sorted TYPE SORTED TABLE OF string WITH UNIQUE KEY table_field.

3.İndeksli İç Tablolar (Hashed Internal Tables):**Tanım:**

- Anahtar alanıyla belirlenen bir karma (hash) fonksiyonunu kullanarak veriyi depolarlar.
- Anahtar alanları benzersiz olmalıdır.

Eriřim:

- Hızlı eriřim saęlarlar. Anahtarın karma deęeri kullanılarak doğrudan eriřim yapılır.
- **READ TABLE** veya **LOOP AT** ile anahtar deęeri ile eriřim yapılabilir.

Sıralama:

- Sıralı değillerdir. Anahtar alanına göre düzenlenirler.

Kullanım:

- Büyük veri setleri ve hızlı erişim gerektiren durumlar için kullanılır.

Komut :

DATA: `lt_hashed TYPE HASHED TABLE OF string WITH UNIQUE KEY table_field.`

Bu tabloların içine veri eklemek için APPEND veya INSERT ifadeleri kullanılır.

Standart İç Tablolar (Standard Internal Tables):**Avantajlar:**

Küçük veri setleri için etkilidir.

Satır numarasına dayalı erişim ve döngüler hızlıdır.

Dezavantajlar:

Büyük veri setlerinde performans düşebilir.

Sıralama ve filtreleme işlemleri için ek işlem gerekebilir.

Sıralı İç Tablolar (Sorted Internal Tables):**Avantajlar:**

Anahtar alanına göre hızlı erişim sağlar.

Sıralama işlemleri doğrudan desteklenir.

Dezavantajlar:

Ekleme ve silme işlemleri maliyetlidir, çünkü sırayı korumak için yeniden düzenleme yapılabilir.

İndeksli İç Tablolar (Hashed Internal Tables):**Avantajlar:**

Çok hızlı erişim sağlar, çünkü anahtar tablo indeksi olarak kullanılır.

Anahtar alanına dayalı benzersizlik sağlar.

Dezavantajlar:

Sıralı değildir, bu nedenle sıralama işlemleri doğrudan desteklenmez.

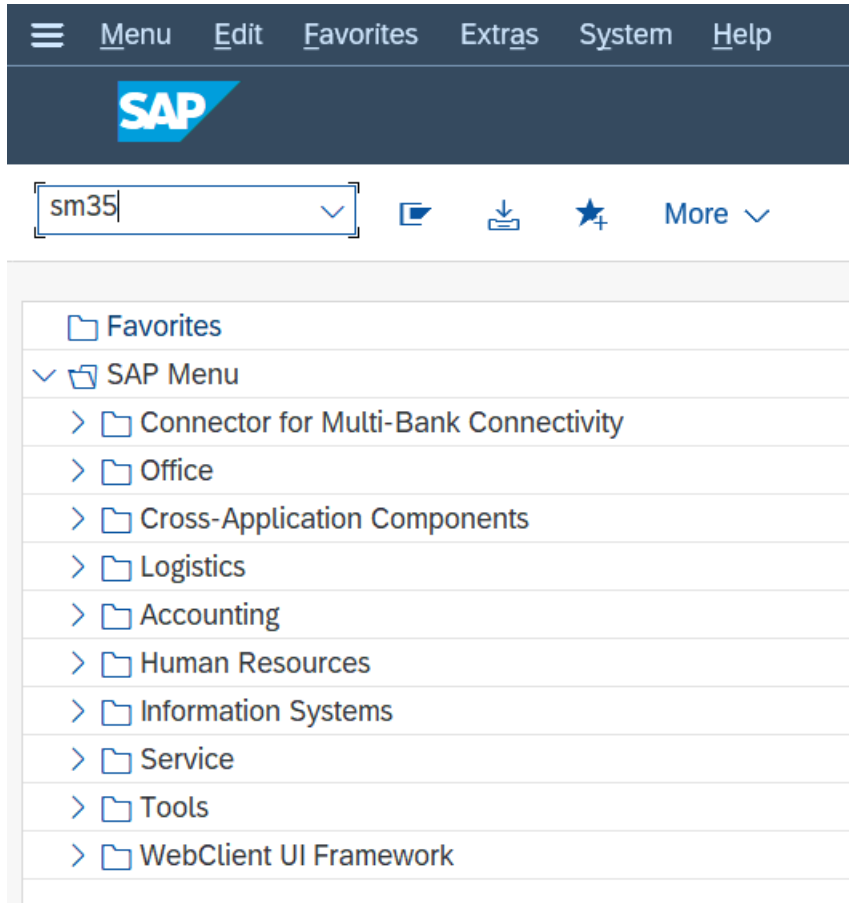
Büyük veri setleri için hafıza kullanımı daha yüksek olabilir.

Batch Input Nedir?

Batch Input, SAP sistemine büyük miktarda veri eklemek veya güncellemek gibi yüksek hacimli veri girişi görevlerini otomatikleştirmek için kullanılan bir yöntemdir. SAP içerirsinde mevcut olan bir süreci, örneğin malzeme oluşturma, satın alma siparişinin girişi, fatura girişi ya da muhasebe kaydının atılması gibi adımları otomatik hale getirebiliriz. Bunu da batch input yapısı kullanarak yaparız.

Batch input normalde manual olarak yapılan bütün işlemleri kendisi otomatik olarak arka planda kaydeder ve bu kayıt işlemi siz bir programa entegre ettiğinizde otomatik olarak çalışır.

Örnek verecek olursak satış yapan bir mağazayı düşünelim; her ana kendi ürün yelpazesini genişletiyor ve bu yeni ürünlerin SAP’de tanımlanması lazım, günde bin tane iki bin tane malzeme oluşturulması lazım, bunu bir insanın manuel olarak yapması ve bütün alanların tek tek doldurulması imkânsız. Bu yüzden batch input kullanılır.



SAP sisteminin giriş ekranından SM35 transaction kodunu girerek Batch Input ekranına geçebiliriz.

Session Edit Goto Utilities System Help

< **SAP** Batch Input: Session Overview

Analysis Process Statistics Log Recording

Selection criteria Edit recordings (Ctrl+F8)

Sess.: * From: To: Created by: *

New Incorrect Processed In Process In Background Being Created Locked

Session name	Stat...	Created By	Date	Time	Created by Prog...	Lock Date	Authorizat.	Application Area	Σ Tran
<input type="checkbox"/> TXCA BUSE/10	🔥	BPINST	10/28/2022	22:07:36	SAPMSSY1		BPINST	GL_TX_DCL	1
<input type="checkbox"/> TXCA BUSE/9T	🔥	BPINST	10/28/2022	22:07:36	SAPMSSY1		BPINST	GL_TX_DCL	4
<input type="checkbox"/> TXCA BUSE/8T	🔥	BPINST	10/28/2022	22:07:36	SAPMSSY1		BPINST	GL_TX_DCL	20
<input type="checkbox"/> TXCA BUSE/7T	🔥	BPINST	10/28/2022	22:07:36	SAPMSSY1		BPINST	GL_TX_DCL	8
<input type="checkbox"/> TXCA BUSE/6T	🔥	BPINST	10/28/2022	22:07:35	SAPMSSY1		BPINST	GL_TX_DCL	8
<input type="checkbox"/> TXCA BUSE/5T	🔥	BPINST	10/28/2022	22:07:35	SAPMSSY1		BPINST	GL_TX_DCL	6

Bu ekranın alt kısmında önceden recording yapılmışsa görünmektedir. Bu ekranda bulunan “Recording” tuşuna basıldığı zaman bizi recording ekranına atar.

< **SAP** Transaction Recorder: Recording Overview

New recording Process Session Program Test data Function mc

Database selection for recordings

Recording: * frm: To: Created by: *

Recording	CreatedBy	Date	Time	Transact.	Screens
<input type="radio"/> ZIKK_MM01	B_KAANK	12/26/2023	20:46:07	1	38

Bu ekranda yeni kayıt, kayıt değiştirme, kayıt silme gibi alanlar bulunmaktadır. New record dediğimizde yeni bir Batch kaydı oluştururuz. Örneğin burada bir malzeme oluşturma adımını record ettiğin zaman yeni bir malzeme oluşturmak istediğinde kayıtlı olan Batch yardımıyla otomatik olarak alanları doldurabilirsin.

Transaction Recorder: Recording Overview

aw recording

63

► Process

Session

Program

Test data

Function module

ngs

frm:

To:

Created by: *

y	Date	Time	Transact.	Screens	
	12/26/2023	20:46:07	1	38	

Yazılan Batch kaydını bir programda kullanmak istediğimiz zaman “Program” adlı butona bastığında SAP bu Batch kaydını programa çeker, yani program içerisinde kullanabileceğiniz bir kod bloğu oluşturuyor ve o kod bloğunu alıp program içerisinde kullanabilirsiniz.

Aynı şekilde fonksiyonda kullanmak isterseniz de “Function module” kısmından kullanabilirsiniz.

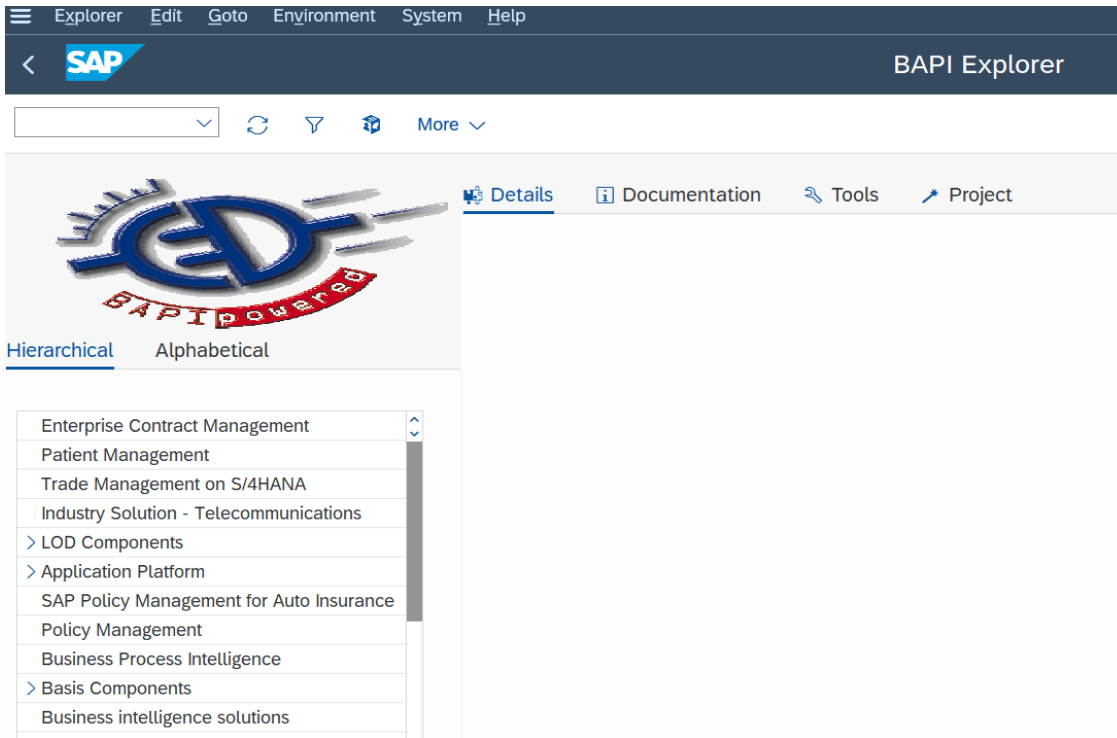
6

BAPI Nedir?

BAPI, "Business Application Programming Interface" kelimelerinin kısaltmasıdır ve temelde SAP tarafından sağlanan bir standart arayüzdür. Bu arayüz, farklı sistemler arasında veri alışverişi yapmamıza (RFC çağrıları) veya sadece bir SAP sistemi içindeki fonksiyon modüllerinde bulunan yöntemleri kullanmamıza olanak tanır.

BAPI'ler genellikle üçüncü taraf sistemlere bağlanmak, uzak fonksiyon çağrıları yapmak, uygulama sırasında SAP'e veri getirmek gibi durumlar için kullanılır. Ayrıca, Java, C#, Visual Basic .NET gibi diğer programlama dilleri ile iletişim kurmak veya SAP sistemlerini diğer ağlar veya internetle bağlamak için de kullanılabilir.

BAPI'ler ile çalışabilmek için öncelikle "BAPI" transaction kodunu girmek gerekiyor.



BAPI gezinti alanı. Burda sol taraftan kullanmak istediğiniz methodları seçebilirsiniz.

SAP BAPI Explorer

Method (BAPI)

Method:

Business object:

Short description:

New in Release:

Function module:

ALE message type:

☒ Instance-dep. ☐ Dialog

Status

Release status:

Hierarchical Alphabetical

- SAP Policy Management for Auto Insurance
 - Policy Management
 - Business Process Intelligence
 - Basics Components
 - Business Management
 - Computer Center Management System (CCMS)
 - ABAP Workbench, Java IDE and Infrastructure
 - FlightBooking
 - AirlineID
 - BookingNumber
 - Cancel
 - Confirm
 - CreateFromData
 - GetList
 - FlightCustomer
 - FlightConnection
 - Flight
 - FlightTrip

Örneğin Uçuş Rezervasyonu altındaki Uçuş rezervasyonu methodu seçildiğinde “Details” kısmında method detaylarını görebilirsiniz. Burda “Function Module” kısmından kaynak kodlara da ulaşabilirsiniz.(bkz : Aşağıdaki görsel)

SAP Function Builder: Display BAPI_FLBOOKING_CANCEL

Function module: Active

Attributes Import Export Changing Tables Exceptions Source code

```

1 FUNCTION BAPI_FLBOOKING_CANCEL.
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28

```

***** Identify internal "key" *****

* The concatenated key of the BO is stored in msg_hlp_key and used *
 * later for providing final return message *

***** End *****

The screenshot shows the SAP documentation interface for the **FlightBooking.Cancel** method. The left sidebar contains a hierarchical tree structure with the following items: Application Platform, SAP Policy Management for Auto Insurance, Policy Management, Business Process Intelligence, Basis Components, Business Management, Computer Center Management System (CCMS), ABAP Workbench, Java IDE and Infrastructure, FlightBooking, AirlineID, BookingNumber, **Cancel** (highlighted), Confirm, CreateFromData, GetList, FlightCustomer, FlightConnection, and Flight. The main content area is titled "Method" and includes the following sections: **FlightBooking.Cancel** (Cancel flight booking), **Functionality** (You can use this method to cancel a flight booking.), **Notes** (This method is only used in the flight data model for demonstration and training purposes. It is not intended for production use. Unlike the BAPI concept, we cannot guarantee its stability. Authorization object S_FLBOOK with activity 02 (change) is checked. The parameters *AirlineID* and *BookingNumber* identify an individual flight booking. If the values of this parameter are unknown, use the relevant *GetList* method to determine them. You can use this method for bookings (flight bookings with the status **Booked**) and for reservations (flight bookings with the status **Reserved**). It cannot be used for flight bookings that include flights in the past. This causes an error method and an error appears in the *Return* parameter. Use the *TestRun* parameter to define whether you want the method to run in simulation mode. If you choose simulation mode, the changes to the object instance are only simulated, and no data is written to the database.), and **Return Codes** (Messages are returned in the *Return* parameter. The parameter documentation shows the return codes and their meanings.).

“Documentation” kısmında bu methodu ne için, hangi durumlarda kullanabilirsiniz gibi bilgilendirmelere ulaşabilirsiniz.

The screenshot shows the SAP Tools selection menu. The top navigation bar includes "Details", "Documentation", "Tools" (highlighted), and "Project". The "Tools selection" section lists the following tools: Create Web Service, Business Object Builder, Function Builder, ABAP Dictionary, and Create BAPI List.

“Tools” kısmında Web Servis oluşturma Bapi Listesi oluşturma gibi kısımları kullanabilirsiniz.

DetailsDocumentationToolsProject

BAPI Project

User Name: B_GAYEA Project -- New Project --

NewSaveDeleteother user

Implementation Project for New BAPIs

This form will guide you through all the work steps necessary for implementing BAPIs. It also enables you to create a checklist to document the steps as you complete them, thereby providing you with a record of the creation process for your BAPI.

Complete this form for each of your projects. One project may cover either a single BAPI, or several BAPIs within the framework of one scenario. However, there must only be one person responsible for each project.

Overview:

1. Project Organization	6. Defining BAPI Methods in the BOR
2. Scenario Analysis	7. Documenting BAPIs
3. BAPI Design	8. BAPI/ALE Integration - Generating IDocs
4. Defining BAPI Data Structures	9. ToDo Checks and Tests
5. Creating BAPI Function Modules	10. Release

“Project” kısmında bir form var ve bu form aracılığıyla adım adım proje oluşturabilirsiniz.

Bu form, BAPI'lerin uygulanması için gerekli tüm çalışma adımlarında size yol gösterecektir. Ayrıca, adımları tamamlarken belgelemek için bir kontrol listesi oluşturmanıza da olanak tanır ve böylece size BAPI'niz için oluşturma sürecinin bir kaydını sağlar.

Bu formu her projeniz için doldurun. Bir proje tek bir BAPI'yi kapsayabileceği gibi, bir senaryo çerçevesinde birden fazla BAPI'yi de kapsayabilir. Ancak her projeden yalnızca bir kişinin sorumlu olması gerekir.

View ve SQL Cümlecikleri Arasındaki Farklar Nelerdir?

Öncelikle view kavramını nedir ve nasıl view oluşturulur kısaca bundan bahsedelim.

SAP ABAP'ta "view" kavramı, bir veritabanı nesnesi olarak kullanılır ve veritabanındaki tablolardan bir veya daha fazlasının birleştirilmiş bir şeklini temsil eder. ABAP görünümü, özel bir ABAP programı olmadan, belirli bir kullanıcı veya kullanıcı grubuna sunulabilen bir mantıklı veri kümesini tanımlar.

İşte SAP ABAP'ta view kavramıyla ilgili bazı temel bilgiler:

1.Tanım ve Amaç:

- **View (Görünüm):** SAP sistemindeki veritabanı tablolarının veya başka bir görünümün birleştirilmiş bir görünümünü sağlar. Bu, veritabanındaki verileri daha organize bir şekilde sunmayı mümkün kılar.

2.Özellikler:

- **Join İşlemleri:** Görünümler, bir veya daha fazla tabloyu birleştirmek için SQL JOIN ifadelerini kullanabilir. Bu, kullanıcının tablolar arasındaki ilişkileri fark etmeden veriye erişmesini sağlar.
- **Filtreleme:** Görünümler, belirli koşulları karşılayan verileri içerecek şekilde filtreleme yapabilir. Bu, kullanıcılara belirli kriterlere uygun verilere odaklanma yeteneği sağlar.
- **Veri Güncelleme:** Görünümler, veritabanındaki tablolar üzerinde güncelleme işlemleri gerçekleştirmek için kullanılabilir. Ancak, bu işlemler genellikle sadece belirli koşulları karşılayan verilere uygulanabilir.

3.Kullanım Alanları:

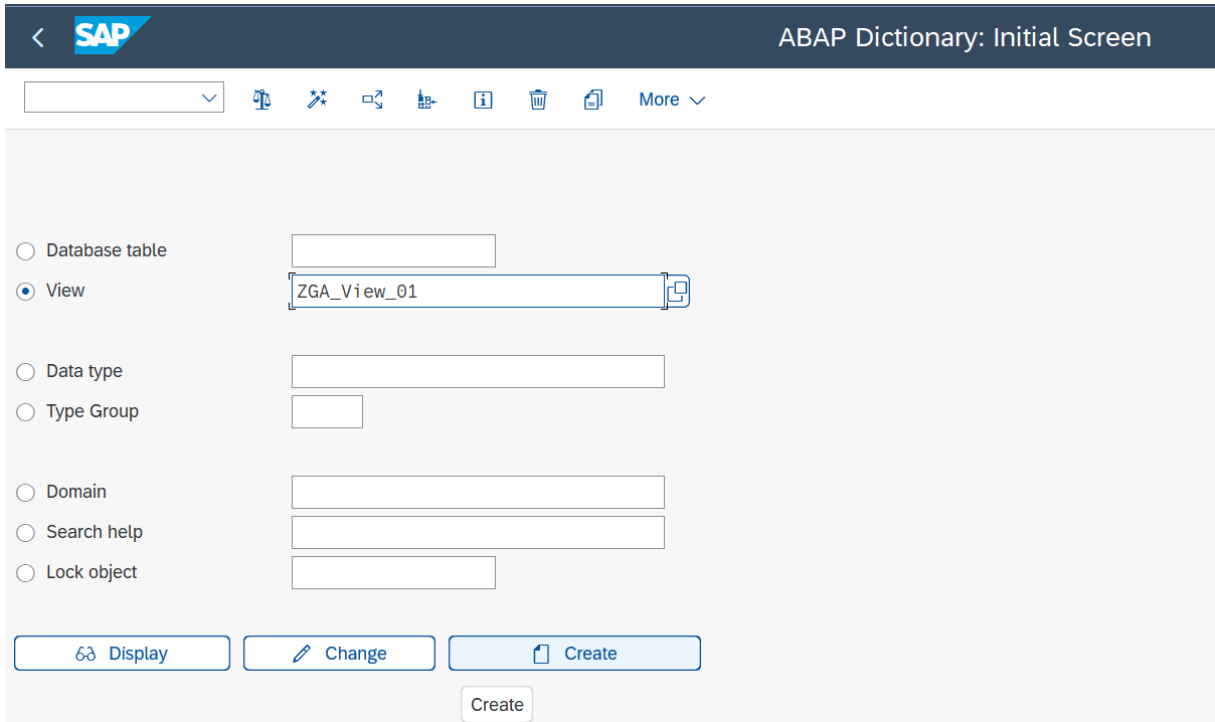
- **Raporlar:** Görünümler, raporlarda kullanılmak üzere verilerin daha okunabilir ve anlamlı bir biçimde düzenlenmesine olanak tanır.
- **İş Süreçleri:** İş süreçlerini yöneten programlarda, görünümler genellikle iş mantığını basitleştirmek ve veri erişimini kolaylaştırmak için kullanılır.

4.Görünüm Türleri:

- **Join View:** Birden çok tabloyu birleştiren bir görünümdür.
- **Projection View:** Belirli sütunları içeren bir görünümdür.
- **Help View:** Kullanıcılara veri girişi sırasında yardım sağlamak amacıyla oluşturulan bir görünümdür.

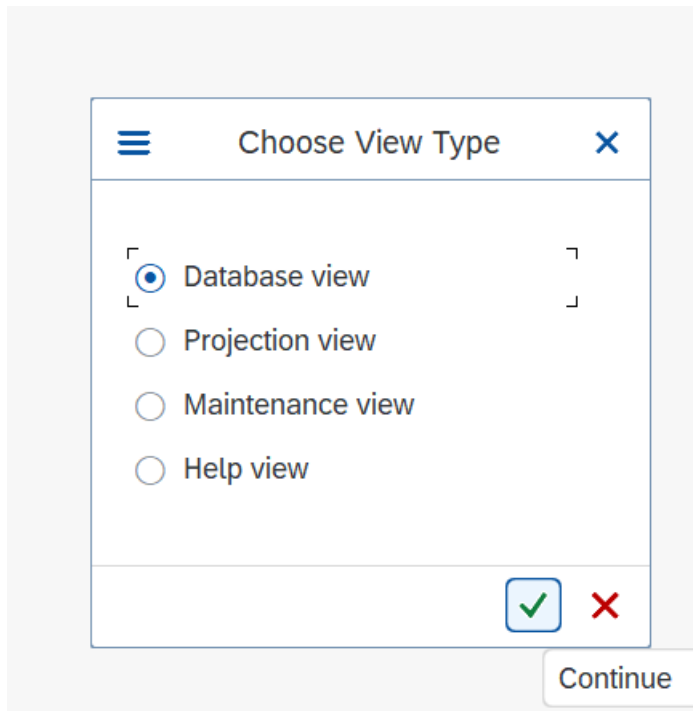
Görünümler, veri erişimini kolaylaştırmak, karmaşıklığı azaltmak ve kullanıcılara daha anlamlı bir veri sunmak için kullanılır. ABAP görünümleri, SAP uygulamalarının veritabanı işlemlerini daha etkili ve kullanıcı dostu hale getirme konusunda önemli bir rol oynar.

View oluşturacağımız zaman yine tablo oluşturmak için kullandığımız se11 transaction kodunu kullanırız.



The image shows the 'ABAP Dictionary: Initial Screen' in SAP. The top bar has the SAP logo and the title 'ABAP Dictionary: Initial Screen'. Below the bar is a toolbar with icons for search, edit, delete, and other actions. The main area contains several radio buttons for object types: 'Database table', 'View' (selected), 'Data type', 'Type Group', 'Domain', 'Search help', and 'Lock object'. Each radio button is followed by a text input field. The 'View' field contains 'ZGA_View_01'. At the bottom, there are three buttons: 'Display', 'Change', and 'Create'. A 'Create' button is also visible below the 'Create' button.

ABAP : Dictionary Initial Screen ekranından “View” butonuna tıklarız ve bir View ismi girip yeni bir view oluşturmak için create’e basarız.



The image shows a 'Choose View Type' dialog box. It has a title bar with a menu icon, the title 'Choose View Type', and a close button. The dialog contains four radio buttons: 'Database view' (selected), 'Projection view', 'Maintenance view', and 'Help view'. At the bottom right, there are two buttons: a green checkmark button and a red 'X' button. Below the dialog, a 'Continue' button is visible.

Create dedikten sonra açılan pencerede, oluşturmak istediğiniz view çeşidini seçebilirsiniz. Ben “Database view” seçtim.

SAP Dictionary: Change View

Database View: ZGA_VIEW_01 New(Revised)

Short Description: * Database View 01

Attributes Table/Join Conditions View Fields Selection Conditions Maint.Status

Tables

Table
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>

Join conditions

Table	Field Name	=	Table	Field name
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				

Relationships

Verilen ekranda tablo bilgilerini girerek view oluşturabilirsiniz, isterseniz Z'li tabloları isterseniz standart tabloları kullanabilirsiniz.

ABAP'ta SQL cümlecikleri ve view'ler, veritabanı işlemleri ve veri erişimini sağlama amaçlarına hizmet eden farklı kavramlardır. Şimdi ,View ile Sql cümlecikleri arasındaki farka gelecek olursak;

1. Amaç:

- **SQL Cümlecikleri:** SQL cümlecikleri, SAP ABAP içinde veritabanı işlemleri gerçekleştirmek için kullanılır. SELECT, INSERT, UPDATE, DELETE gibi SQL ifadeleri kullanılarak veri çekme, ekleme, güncelleme ve silme işlemleri yapılır.
- **View:** ABAP'taki view'ler, veritabanındaki tabloların veya başka view'ların birleştirilmiş bir görünümünü oluşturarak veri erişimini kolaylaştırmak için kullanılır. View'ler, genellikle belirli bir kullanıcıya veya kullanıcı grubuna daha anlamlı ve düzenli bir veri kümesi sunmak amacıyla oluşturulur.

2. Tanımlama ve Kullanım:

- **SQL Cümlecikleri:** SQL cümlecikleri, belirli bir veritabanı işlemini gerçekleştirmek için kullanılan kod parçacıklarıdır. Bu cümlecikler, bir program içinde belirli bir noktada yazılarak çağrılabilir veya doğrudan çalıştırılabilir.

- **View:** View'ler, genellikle SAP veritabanında tanımlanan nesnelerdir. View'lar, bir kez tanımlandıktan sonra, veri çekme ve manipölasyon işlemleri için kullanılabilir ve bu işlemler bir ABAP programında çağrılabilir.

3. İşlemler ve Kapsam:

- **SQL Cümlecikleri:** SQL cümlecikleri genellikle belirli bir tablo üzerinde doğrudan bir veritabanı işlemi gerçekleştirir. Örneğin, bir tablodan veri çekmek için SELECT cümlecikleri kullanılır.
- **View:** View'lar, bir veya daha fazla tablonun veya başka bir view'ın birleştirilmiş bir şeklini temsil eder. View'lar, genellikle JOIN ifadeleri kullanılarak oluşturulurlar ve birleştirilen tablolar arasındaki ilişkileri yansıtarlar.

4. Dinamiklik:

- **SQL Cümlecikleri:** SQL cümlecikleri genellikle derleme zamanında belirlenir ve programın çalışma zamanında değiştirilemez. Bu, SQL cümleciklerinin statik olmasını sağlar.
- **View:** View'lar, dinamik olarak kullanılmak üzere oluşturulabilir ve çalışma zamanında belirli koşullara göre filtrelenmiş verileri sağlamak için kullanılabilir.

Her iki kavram da SAP ABAP içinde veritabanı işlemleri ve veri erişimi için önemli roller oynar, ancak farklı seviyelerde soyutlama ve işlevsellik sağlarlar. SQL cümlecikleri daha temel ve spesifik bir seviyede işlem yapmaya odaklanırken, view'ler daha geniş bir veri erişimini sağlamak için kullanılır ve belirli bir mantıklı veri kümesini temsil eder.

Background Jobs Nedir?

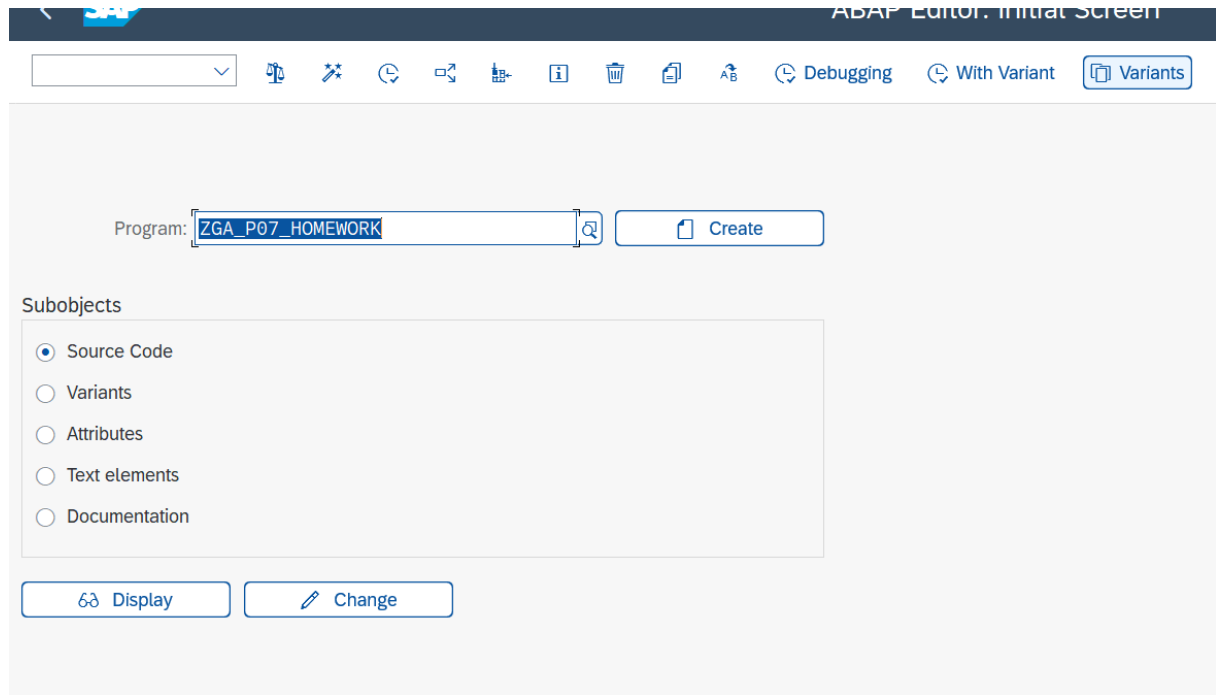
SAP sistemi içinde "background job" terimi, sistemde çalışan ve kullanıcının etkileşimine gerek olmayan, genellikle planlı, sürekli veya tekrarlanan işlemleri ifade eder. Bu işlemler, genellikle büyük veri setleri üzerinde uzun süren işlemleri içerir ve kullanıcı tarafından doğrudan etkileşime gerek duymazlar.

İşlemler genellikle "SM37" veya "SM36" gibi SAP transaction kodları aracılığıyla izlenebilir ve yönetilebilir.

SAP sistemi, background job'ları belirli bir zaman diliminde veya belirli bir periyotta otomatik olarak çalıştırmak için planlama (scheduling) özellikleri sunar.

Bir job oluşturmadan önce job içerisinde kullanılacak variant oluşturulmalıdır.

Se38 ekranından variant oluşturmak istediğimiz bir programı seçeriz.



Se38 ekranının sağ üst kısmında bulunan “variants” kısmına tıklayın.

< **SAP** ABAP: Variants - Initial Screen

Catalog More

Program:
 Variant: Create
 Create

Subobjects

☒ Values
☐ Attributes

Display Change

Variant ekranında program ismi ve varian ismi girilerek create butonuyla yeni variant oluşturma ekranına geçilir.

Variant Edit Goto Environment System Help

< **SAP** Variant Attributes

More

Variant Name:
 Description*:

☒ Only for Background Processing
☒ Protect Variant
☐ Only Display in Catalog
☐ System Variant (Automatic Transport)

Screen Assignment

	Created	Selection Screen
<input type="checkbox"/>	<input checked="" type="checkbox"/>	1000

Fields of selection screen

	Dynpro	Field name	Type	Protect field	Hide field	Hide field 'TO'	Do not save values	Switch GPA off	Required field	Sel. Variable	Type	Option
<input type="checkbox"/>	1,000	P_TEXT	P	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			

Varyant oluşturmak için gerekli bilgiler girilerek program kaydedilir.

Varyant oluřturduktan sonra artık job oluřturma kısmına geilebilir. Sm36 transaction koduyla job oluřturma ekranı aılır.

Start condition Step Job selection Own jobs Job wizard Job Repository

General Data

Job Name: ZGA_JOB_ODEV

Job Class: A

Status: Scheduled

Target:

E-Mail Notification

Spool List Recipient

Job Start

Recipient Determination

Recipient: D IEMUR

General attributes

☐ Copy ☐ Blind copy

☐ Express ☐ No Forwarding

Copy Fax entry X.400 entry Address

Burada job gnderilecek kiři “Spool List Recipient “ kısmından girilir. Yine aynı řekilde “Target” kısmına da bulunduėun server girilir.

Start condition Step Job selection Own jobs

Define steps (F6)

General Data

Job Name: ZGA_JOB_ODEV

Job Class: A

Status: Scheduled

E-Mail Notification

Daha sonra “step” kısmı seilerek job’ın step’i oluřturulur.

User: B_GAYEA

Program values

ABAP program

External command

External program

ABAP program

Name: ZGA_P07_HOMEWORK

Variant: ZGA_VARIANT_02

Language: EN

Bu kısımda da gerekli yerler doldurulur ve adım kaydedilir.

≡

Step Edit Goto List System Help

< SAP

Step List Overview

▼

✎

📄

🔍

📊

68

🗑️

📄 Spool

📅

⏪

<

>

⏩

More ▼

No.	Program name/command	Prog. type	Spool list	Parameters	User	Lang.
1	ZGA_P07_HOMEWORK	ABAP		ZGA_VARIANT_02	B_GAYEA	EN

Kaydedilmiş adımlar listelenir.

< SAP Define

Start condition Step Job selection Own jobs Job

Specify start condition (F5)

General Data

Job Name: ZGA_JOB_ODEV

Job Class: A

Status: Scheduled

Target:

E-Mail Notification

Spool List Recipient

Daha sonra sm36 ekranındaki “Start condition” kısmından job’ın durumu belirlenir.

Job Edit Goto System Help

< SAP

Start condition Step

General Data

Job Name: ZGA_JOB_ODEV

Job Class: A

Status: Scheduled

Target:

Job Start

Period Values

Hourly

Daily

Weekly

Monthly

Other period

Save X Check

Job Steps

1 Step(s) defined

Start Time

Immediate Date/Time After Job After Event Operation Mode

Date/Time

Immediate Start

After Job

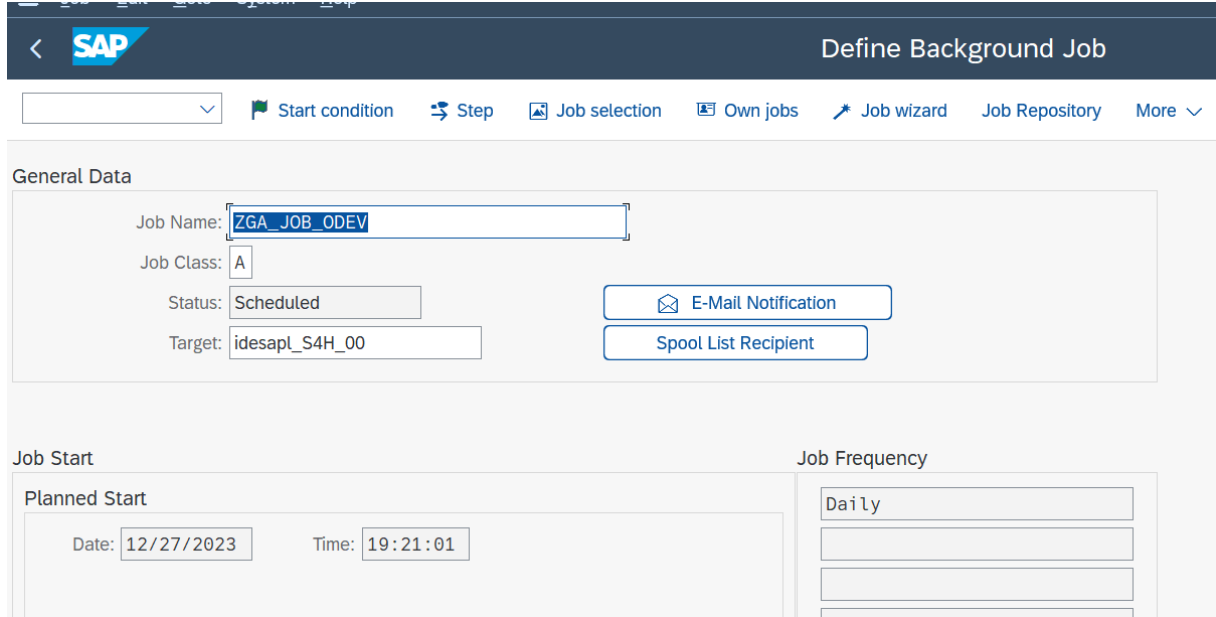
Operation Mode

After Event

Periodic Job

Save X Check Period values Restrictions

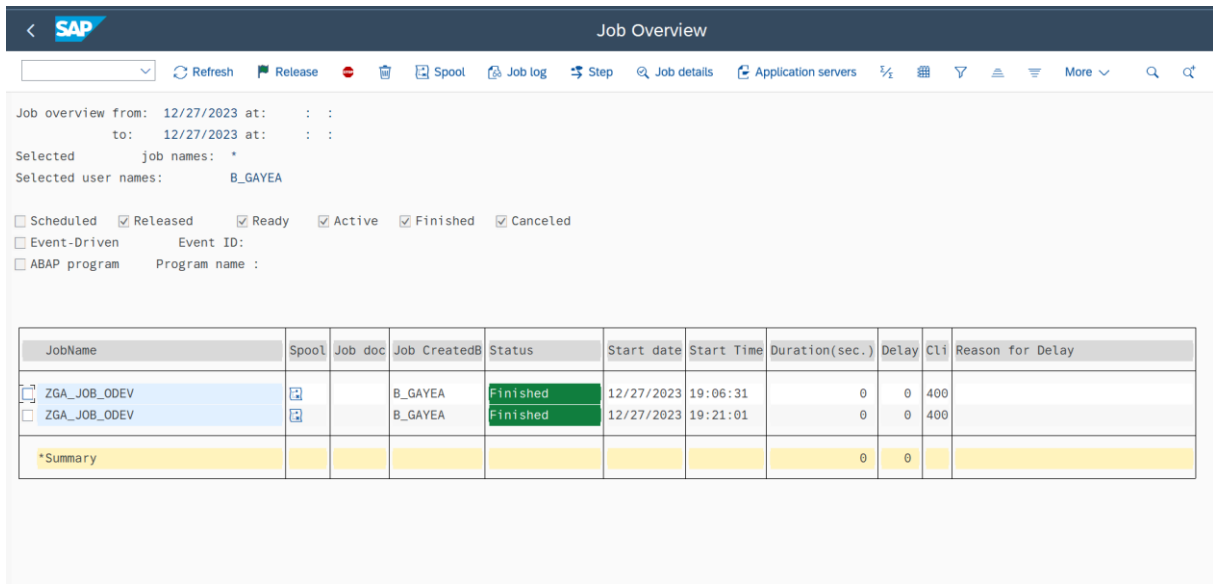
Yukarıda verilen görseldeki gibi job'ın çalışmasını istediğimiz tarih, zaman aralığı ya da kaç dakikada bir çalışacağı belirlenebilir.



The screenshot shows the 'Define Background Job' interface in SAP. The 'General Data' section includes fields for Job Name (ZGA_JOB_ODEV), Job Class (A), Status (Scheduled), and Target (idesapl_S4H_00). There are buttons for 'E-Mail Notification' and 'Spool List Recipient'. The 'Job Start' section shows 'Planned Start' with Date (12/27/2023) and Time (19:21:01). The 'Job Frequency' section shows 'Daily'.

Her aşama tamamlandıktan sonra job kaydedilir.

Oluşturulmuş job'ları görüntülemek için sm37 tcode kullanılabilir.



The screenshot shows the 'Job Overview' interface in SAP. It displays a table of jobs with columns: JobName, Spool, Job doc, Job CreatedBy, Status, Start date, Start Time, Duration(sec.), Delay, Cl, and Reason for Delay. Two jobs are listed, both with status 'Finished'. A summary row is also present.

JobName	Spool	Job doc	Job CreatedBy	Status	Start date	Start Time	Duration(sec.)	Delay	Cl	Reason for Delay
ZGA_JOB_ODEV			B_GAYEA	Finished	12/27/2023	19:06:31	0	0	400	
ZGA_JOB_ODEV			B_GAYEA	Finished	12/27/2023	19:21:01	0	0	400	
*Summary							0	0		

Oluşturulan job'ların görünümü.