

1. DimDate

هدف: ایجاد یک جدول تقویم کامل که تمام تاریخ‌ها را از «تاریخ شروع» (مثلاً ۲۲ ژوئن ۲۰۲۵) تا امروز در اختیار قرار می‌دهد.
نحوه بارگذاری اولیه

۱. رکوردهای پیشین را پاک می‌کند و شناسه (سروگیت) دوباره از ۱ شروع می‌شود.
۲. با استفاده از یک CTE عددی، به تعداد روزهای بین تاریخ شروع و امروز، یک ردیف ایجاد می‌کند؛ برای هر ردیف، ستون‌های سال، ماه، روز، شماره هفته در سال، نام ماه و نام روز هفته و پرچم تعطیل/غیرتعطیل (Weekend) محاسبه و درج می‌شود.
از این جدول برای Join کردن تمام جدول‌های فکت بر مبنای تاریخ (تاریخ فاکتور، تاریخ پرداخت، ...) استفاده می‌شود.

2. DimCustomer (SCD Type 2)

هدف نگهداری تاریخچه تغییرات اطلاعات مشتری (نام، نوع، کشور و ...) به صورت SCD2
بارگذاری اولیه

۱. تمامی رکوردها را خالی می‌کند.
۲. برای هر مشتری از استیجینگ، یک ردیف جدید با `StartDate = Today`، `EndDate = NULL` و `IsCurrent = 1` درج می‌کند.
نگهداری تغییرات
 - در به‌روزرسانی‌های بعدی وقتی اطلاعات مشتری (مثلاً آدرس یا نوع) تغییر می‌کند، ردیف جاری با `IsCurrent = 1` به‌روزرسانی می‌شود `EndDate = Today` و `IsCurrent = 0`، سپس یک ردیف جدید با مقادیر به‌روز و `IsCurrent = 1` درج می‌شود.
 - امکان گزارش‌گیری «در یک تاریخ مشخص چه اطلاعاتی برای مشتری وجود داشته» فراهم می‌شود.

3. DimServiceType (SCD Type 1)

هدف فهرست انواع خدمات فعلی؛ تاریخچه نگهداری نمی‌شود. (SCD Type 1)
بارگذاری اولیه

۱. جدول را truncate کرده و تمام ردیف‌های `ServiceType` از استیجینگ را درج می‌کند.
نگهداری تغییرات
 - در هر بار به‌روزرسانی، سطرهای موجود را با داده‌های جدید جایگزین (update/overwrite) می‌کند یا سرویس جدید را insert می‌کند.
 - این دایمنش برای Lookup کردن در فکت `InvoiceLine` استفاده می‌شود.

4. DimTax (SCD Type 2)

هدف نگهداری اطلاعات مالیات با تاریخچه – نرخ، نوع و بازه اثرگذاری. (EffectiveFrom/To)
بارگذاری اولیه

۱. تمام رکوردها پاک می‌شوند و هر ردیف با $\text{StartDate} = \text{Today}$ و $\text{IsCurrent} = 1$ درج می‌شود.
نگهداری تغییرات

- اگر نرخ یا تاریخ اثرگذاری مالیات جدیدی در استیج تغییر کند، ردیف جاری به $\text{EndDate} = \text{Today}$ و $\text{IsCurrent} = 0$ بروزرسانی می‌شود و یک ردیف جدید جاری درج می‌گردد.

- از این دایمنش برای محاسبه مالیات ردیف‌های فاکتور بر اساس تاریخ صدور استفاده می‌شود.

5. DimBillingCycle (SCD Type 1)

هدف تعریف دوره صدور فاکتور (ماهانه، فصلی، ...).
بارگذاری اولیه

۱. Truncate و سپس درج همه دوره‌ها از استیجینگ.
نگهداری تغییرات

- دوره‌های جدید یا تغییر یافته را update/insert می‌کند؛ تاریخچه نگهداری نمی‌شود.

- به قراردادهای (Contracts) پیوسته و در snapshot فکت ماهانه استفاده می‌شود.

6. DimPaymentMethod (SCD Type 1)

هدف لیست روش‌های پرداخت (نقد، کارت، انتقال و ...).
بارگذاری اولیه

۱. Truncate و insert مقادیر متمایز از ستون PaymentMethod در جدول پرداخت‌های استیجینگ.

نگهداری تغییرات

- فقط روش‌های جدید را insert می‌کند؛ تاریخچه نگهداری نمی‌شود.

- برای پیوستن به فکت FactCustomerPaymentTransaction کاربرد دارد.

7. DimContract (SCD Type 3)

نگهداری «وضعیت فعلی» یک قرارداد و تا دو مرتبه «وضعیت قبلی» آن. بارگذاری اولیه

۱. truncate و insert همه قراردادها با PrevContractStatus1/2 = NULL.

نگهداری تغییرات

- هنگام تغییر وضعیت قرارداد، وضعیت قبلی اول به PrevContractStatus1 و آن یکی به PrevContractStatus2 منتقل می شود و وضعیت فعلی به ستون ContractStatus نوشته می شود.

Active → Suspended

مثلاً اگر پرداختی به موقع انجام نشود یا اختلاف حقوقی پیش بیاید، ممکن است وضعیت به «معلق (Suspended)» تغییر کند.

Suspended → Active

بعد از رفع مشکل (پرداخت بدهی، حل اختلاف)، قرارداد دوباره به حالت «فعال» برمی گردد.

Active → Expired → Renewed

بعد از پایان دوره ی اولیه (مثلاً یک ساله)، وضعیت به «Expired» می رسد. اگر طرفین تصمیم به تمدید بگیرند، ممکن است وضعیت به «Active» یا «Renewed» تغییر کند.

Terminated

در موارد خاص، ممکن است قرارداد زودتر از موعد به «Terminated» (قطع شده) تغییر کند.

در fact انباشته (accumulating snapshot) برای تحلیل چرخه عمر فاکتور استفاده می شود.

8. DimInvoice (SCD Type 1)

هدف ذخیره مشخصات ثابت سربرگ فاکتور (شماره، وضعیت، تاریخ ایجاد، ...). بارگذاری اولیه

۱. truncate و insert تمام فاکتورها.

نگهداری تغییرات

- فقط فاکتورهای جدید اضافه می شوند؛ آپدیت صورت نمی گیرد.
- کلید مرجع برای همه فکت های مرتبط با فاکتور.

1. LoadFactInvoiceLineTransactionInitialLoad

هدف: ایجاد فکت تراکنشی جزئیات سطرهای فاکتور برای گزارش گیری بر اساس ابعاد فاکتور، مشتری، نوع خدمت، مالیات و تاریخ.

بارگذاری اولیه

حذف تمام رکوردهای قبلی از FactInvoiceLineTransaction

اعتبارسنجی اینکه ستونهای کلیدی InvoiceID, staging.InvoiceLine, InvoiceID, Quantity, ServiceTypeID, UnitPrice, TaxAmount, NetAmount خالی نباشند.

درج همه رکوردها با JOIN به DimInvoice (InvoiceNumber), DimCustomer (CustomerCode), DimServiceType, DimDate (InvoiceDate), DimTax

نگهداری مقادیر TaxAmount, NetAmount, DiscountPercent, UnitPrice, Quantity و

بهروزرسانیهای بعدی

استخراج رکوردهای جدید یا تغییر یافته از staging.InvoiceLine

INSERT شرطی با WHERE NOT EXISTS برای جلوگیری از درج مضاعف

ثبت تعداد رکوردهای درج شده در ETLLog

2. LoadFactCustomerPaymentTransactionInitialLoad

هدف: ایجاد فکت تراکنشی پرداخت مشتریان برای تحلیل جریان نقدی و رفتار پرداخت شامل مبلغ، تأخیر، مانده و روش پرداخت.

بارگذاری اولیه

حذف تمام رکوردهای قبلی از FactCustomerPaymentTransaction

اعتبارسنجی NULL در ستونهای InvoiceID, PaymentDate, Amount, PaymentMethod, staging.Payment

JOIN به DimInvoice, DimCustomer, DimPaymentMethod, DimDate

محاسبه RemainingAmount, DaysToPayment, IsFullPayment و درج PartialPaymentCount صفر

بهروزرسانیهای بعدی

استخراج رکوردهای جدید از staging.Payment

INSERT شرطی با WHERE NOT EXISTS

ثبت تعداد رکوردهای درج شده در ETLLog

3. LoadFactCustomerBillingMonthlySnapshotInitialLoad

هدف: ایجاد Periodic Snapshot ماهانه از وضعیت مالی مشتری شامل تعداد فاکتورها، مبلغ خالص و مالیات، مجموع تخفیف، مجموع پرداخت، تأخیر متوسط و بیشینه مانده.

بارگذاری اولیه

حذف همه رکوردها از FactCustomerBillingMonthlySnapshot

اعتبارسنجی NULL در InvoiceDate و ContractID از staging.Invoice

CTE اول: تجمیع InvoiceLine بر اساس مشتری و تاریخ اولین روز ماه TotalInvoiceCount ، TotalNetAmount ، TotalDiscount ، TotalTaxAmount

CTE دوم: تجمیع Payment بر اساس کلید TotalPaid ، AveragePaymentDelay

CTE سوم: محاسبه بیشینه مانده Outstanding

JOIN همه CTE ها با ابعاد DimCustomer ، DimDate ، DimBillingCycle و INSERT نهایی

به روزرسانی های بعدی

تکرار aggregation برای داده های جدید (ماه های اضافه شده)

INSERT ، شرطی با WHERE NOT EXISTS روی کلید مرکب DimDateID ، DimCustomerID

ثبت لاگ تعداد درج ها

4. LoadFactInvoiceLifecycleAccumulatingInitialLoad

هدف: ساخت یک Accumulating Snapshot برای دنبال کردن چرخه حیات هر فاکتور تا آخرین پرداخت: تعداد پرداخت، تأخیر اول و آخر، روز تا بسته شدن.

بارگذاری اولیه

حذف رکوردها از FactInvoiceLifecycleAccumulating

اعتبارسنجی NULL در InvoiceDate و DueDate

CTE برای تجمیع پرداخت ها Amount SUM ، COUNT ، MIN و MAX تاریخ پرداخت

با ابعاد DimInvoice ، DimCustomer ، DimContract ، DimDate برای InvoiceDateID ، DueDateID

FinalPaymentDateID ، FirstPaymentDateID

داده های محاسبه شده

PaidAmount ، OutstandingAmount ، PaymentCount ، FirstPaymentDelay ، FinalPaymentDelay ، DaysToClose

به روزرسانی های بعدی

CTE PayAgg برای یک رکورد پرداخت

CTE SrcRaw با ROW_NUMBER برای اطمینان از یک به یک بودن بر DimInvoiceID

MERGE بین Src و جدول فکت

WHEN MATCHED → UPDATE – ستون های متغیر

WHEN NOT MATCHED → INSERT – رکورد جدید

OUTPUT به یک جدول موقت MergeLog و شمارش INSERT/UPDATE

ثبت لاگ نهایی

5. LoadFactCustomerContractActivationInitialLoad

هدف: ثبت تاریخ فعال سازی هر قرارداد برای هر مشتری (factless) تا بتوان تعداد فعال سازی ها بر اساس تاریخ را تحلیل کرد.

بارگذاری اولیه

پاک کردن FactCustomerContractActivationFactless

JOIN staging.Contract → staging.Customer → DimCustomer و DimContract و DimDate (بر اساس StartDate)

INSERT سه تایی (DimDateID, DimCustomerID, DimContractID)

به روز رسانی های بعدی

INSERT رکوردهای جدید با WHERE NOT EXISTS

ثبت تعداد رکوردهای جدید در ETLLLog