

Project 2 - RECREATE THE BICLASS DATABASE START SCHEMA

Group 3 - *Damian Rozpedowski, Hannah Kurian, Hasnatul Hosna, Essmer Sanchez, Zicheng Yang*

Due: 4/14/2024

Project Procedures

```
In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Damian Rozpedowski
-- Procedure:       [Process].[usp_ShowWorkflowSteps]
-- Create date:     04-12-24
-- Description:      Displays the [Process].[WorkflowSteps] table
--                  which was derived from TrackWorkFlow procedure.
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

ALTER PROCEDURE [Process].[usp_ShowWorkflowSteps]
AS
BEGIN
    SELECT *
    FROM [Process].[WorkFlowSteps];

END;
```

```
In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Damian Rozpedowski
-- Procedure:       [Process].[usp_TrackWorkFlow]
-- Create date:     04-12-24
-- Description:      A stored procedure that is called in every other stored
--                  procedure that is used to load the star schema.
--                  Traces the elapsed time for each load and procedure.
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Process].[usp_TrackWorkFlow]
    @WorkflowDescription NVARCHAR(100),
    @WorkflowStepTableRowCount INT,
    @UserAuthorization INT,
    @startTime DATETIME2,
    @endTime DATETIME2
AS
BEGIN
    SET NOCOUNT ON;
    INSERT INTO Process.WorkflowSteps
```

```

VALUES
    (WorkFlowStepKey, WorkFlowStepDescription, WorkFlowStepTableRowCount, Us
(NEXT VALUE FOR [PkSequence].[WorkFlowStepsSequenceObject], @WorkFlowDes
END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Damian Rozpedowski
-- Create date: 04-12-24
-- Description: Add Foreign Keys to the Star Schema
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[AddForeignKeysToStarSchemaData]
    -- Add the parameters for the stored procedure here
    @GroupMemberUserAuthorizationKey int
AS
BEGIN
    declare @start as datetime2, @end as datetime2;
    set @start = SYSDATETIME()

    ALTER TABLE [CH01-01-Fact].[Data] WITH CHECK ADD CONSTRAINT [FK_Data_DimGender]
FOREIGN KEY([Gender]) REFERENCES [CH01-01-Dimension].[DimGender] ([Gender])
    ALTER TABLE [CH01-01-Fact].[Data] CHECK CONSTRAINT [FK_Data_DimGender]
    ALTER TABLE [CH01-01-Fact].[Data] WITH CHECK ADD CONSTRAINT [FK_Data_DimCustomer]
FOREIGN KEY([CustomerKey]) REFERENCES [CH01-01-Dimension].[DimCustomer] ([CustomerKey])
    ALTER TABLE [CH01-01-Fact].[Data] CHECK CONSTRAINT [FK_Data_DimCustomer]
    ALTER TABLE [CH01-01-Fact].[Data] WITH CHECK ADD CONSTRAINT [FK_Data_DimMaritals]
FOREIGN KEY([MaritalStatus]) REFERENCES [CH01-01-Dimension].[DimMaritalStatus] ([Marital
    ALTER TABLE [CH01-01-Fact].[Data] CHECK CONSTRAINT [FK_Data_DimMaritalStatus]
    ALTER TABLE [CH01-01-Fact].[Data] WITH CHECK ADD CONSTRAINT [FK_Data_DimOccupati]
FOREIGN KEY([OccupationKey]) REFERENCES [CH01-01-Dimension].[DimOccupation] ([Occupation
    ALTER TABLE [CH01-01-Fact].[Data] CHECK CONSTRAINT [FK_Data_DimOccupation]
    ALTER TABLE [CH01-01-Fact].[Data] WITH CHECK ADD CONSTRAINT [FK_Data_DimOrderDat]
FOREIGN KEY([OrderDate]) REFERENCES [CH01-01-Dimension].[DimOrderDate] ([OrderDate])
    ALTER TABLE [CH01-01-Fact].[Data] CHECK CONSTRAINT [FK_Data_DimOrderDate]
    ALTER TABLE [CH01-01-Fact].[Data] WITH CHECK ADD CONSTRAINT [FK_Data_DimTerritor]
FOREIGN KEY([TerritoryKey]) REFERENCES [CH01-01-Dimension].[DimTerritory] ([TerritoryKey]
    ALTER TABLE [CH01-01-Fact].[Data] CHECK CONSTRAINT [FK_Data_DimTerritory]
    ALTER TABLE [CH01-01-Fact].[Data] WITH CHECK ADD CONSTRAINT [FK_Data_SalesManage]
FOREIGN KEY([SalesManagerKey]) REFERENCES [CH01-01-Dimension].[SalesManagers] ([SalesMan
    ALTER TABLE [CH01-01-Fact].[Data] CHECK CONSTRAINT [FK_Data_SalesManagers]
    ALTER TABLE [CH01-01-Dimension].[DimProductSubcategory] WITH CHECK ADD CONSTRAIN
FOREIGN KEY([ProductCategoryKey]) REFERENCES [CH01-01-Dimension].[DimProductCategory] ([
    ALTER TABLE [CH01-01-Dimension].[DimProductSubcategory] CHECK CONSTRAINT [FK_Dim
    ALTER TABLE [CH01-01-Dimension].[DimProduct] WITH CHECK ADD CONSTRAINT [FK_DimPr
FOREIGN KEY([ProductSubcategoryKey]) REFERENCES [CH01-01-Dimension].[DimProductSubcatego
    ALTER TABLE [CH01-01-Dimension].[DimProduct] CHECK CONSTRAINT [FK_DimProduct_Dim
    ALTER TABLE [CH01-01-Fact].[Data] WITH CHECK ADD CONSTRAINT [FK_Data_DimProduct]
FOREIGN KEY([ProductKey]) REFERENCES [CH01-01-Dimension].[DimProduct] ([ProductKey])
    ALTER TABLE [CH01-01-Fact].[Data] CHECK CONSTRAINT [FK_Data_DimProduct]

    declare @rowcount as int
    set @rowcount = 0
    set @end = SYSDATETIME()
    EXEC [Process].[usp_TrackWorkFlow]
        @WorkFlowDescription = N'[Project2].[AddForeignKeysToStarSchemaData]: Ad
        @WorkFlowStepTableRowCount = @rowcount,
        @UserAuthorization = @GroupMemberUserAuthorizationKey,
        @startTime = @start,
        @endTime = @end
END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:                Damian Rozpedowski
-- Create date: 04-12-24
-- Description: Drop the Foreign Keys from the Star Schema
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[DropForeignKeysFromStarSchemaData]
    @GroupMemberUserAuthorizationKey INT
AS
BEGIN
    ---DECLARE @start AS datetime2, @end AS datetime2;
    --SET @start = SYSDATETIME();

    -- Dropping All foreign keys as part of the load process to for referential inte
    ALTER TABLE [CH01-01-Fact].[Data]
DROP CONSTRAINT IF EXISTS [FK_Data_DimOccupation]
    ALTER TABLE [CH01-01-Fact].[Data]
DROP CONSTRAINT IF EXISTS [FK_Data_DimOrderDate]
    ALTER TABLE [CH01-01-Fact].[Data]
DROP CONSTRAINT IF EXISTS [FK_Data_DimCustomer]
    ALTER TABLE [CH01-01-Fact].[Data]
DROP CONSTRAINT IF EXISTS [FK_Data_DimGender]
    ALTER TABLE [CH01-01-Fact].[Data]
DROP CONSTRAINT IF EXISTS [FK_Data_DimMaritalStatus]
    ALTER TABLE [CH01-01-Fact].[Data]
DROP CONSTRAINT IF EXISTS [FK_Data_DimProduct]
    ALTER TABLE [CH01-01-Fact].[Data]
DROP CONSTRAINT IF EXISTS [FK_Data_DimProductCategory]
    ALTER TABLE [CH01-01-Fact].[Data]
DROP CONSTRAINT IF EXISTS [FK_Data_DimTerritory]
    ALTER TABLE [CH01-01-Fact].[Data]
DROP CONSTRAINT IF EXISTS [FK_Data_DimProductSubcategory]
    ALTER TABLE [CH01-01-Fact].[Data]
DROP CONSTRAINT IF EXISTS [FK_Data_SalesManagers]
    ALTER TABLE [CH01-01-Dimension].DimProduct
DROP CONSTRAINT IF EXISTS [FK_DimProduct_DimProductSubcategory]
    ALTER TABLE [CH01-01-Dimension].DimProductSubcategory
DROP CONSTRAINT IF EXISTS [FK_DimProductSubcategory_DimProductCategory]

    -- Wont work since truncate would delete this in the workflowsteps, added this into
    -- DECLARE @WorkFlowStepTableRowCount INT;
    -- SET @WorkFlowStepTableRowCount = 0;
    -- DECLARE @EndingDateTime DATETIME2 = SYSDATETIME();
    -- EXEC [Process].[usp_TrackWorkFlow] 'Drop Foreign Keys',
    --     @WorkFlowStepTableRowCount,
    --     @UserAuthorization = @GroupMemberUserAuthorizationKey,
    --     @startTime = @start,
    --     @endTime = @end;
END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:                Group 3 - Team Effort
-- Create date: 04-12-24
-- Description: Populates the Fact.Data table
-- =====
SET ANSI_NULLS ON

```

```

GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[Load_Data]
    @GroupMemberUserAuthorizationKey INT
AS
BEGIN

    DECLARE @start AS datetime2, @end AS datetime2;
    SET @start = SYSDATETIME();
    SET NOCOUNT ON;

    INSERT INTO [CH01-01-Fact].[Data] (
        SalesKey, SalesManagerKey, OccupationKey, TerritoryKey, ProductKey, CustomerKey,
        ProductCategory, SalesManager, ProductSubcategory, ProductCode,
        ProductName, Color, ModelName, OrderQuantity, UnitPrice, ProductStandardCost,
        SalesAmount, OrderDate, MonthName, MonthNumber, Year, CustomerName, MaritalStatus,
        Gender, Education, Occupation, TerritoryRegion, TerritoryCountry, TerritoryGroup,
        UserAuthorizationKey, DateAdded, DateOfLastUpdate
    )
    SELECT
        NEXT VALUE FOR [PkSequence].[DataSequenceObject], -- Assuming a sequence for SalesManagers.SalesManagerKey,
        SalesManagers.SalesManagerKey,
        occ.OccupationKey,
        terr.TerritoryKey,
        prod.ProductKey,
        cust.CustomerKey,
        Original.ProductCategory,
        Original.SalesManager,
        Original.ProductSubcategory,
        Original.ProductCode,
        Original.ProductName,
        Original.Color,
        Original.ModelName,
        Original.OrderQuantity,
        Original.UnitPrice,
        Original.ProductStandardCost,
        Original.SalesAmount,
        Original.OrderDate,
        Original.MonthName,
        Original.MonthNumber,
        Original.Year,
        Original.CustomerName,
        Original.MaritalStatus,
        Original.Gender,
        Original.Education,
        Original.Occupation,
        Original.TerritoryRegion,
        Original.TerritoryCountry,
        Original.TerritoryGroup,
        @GroupMemberUserAuthorizationKey,
        SYSDATETIME(),
        SYSDATETIME()
    FROM FileUpload.OriginallyLoadedData AS Original
    LEFT JOIN [CH01-01-Dimension].DimTerritory AS terr ON Original.TerritoryCountry = terr.TerritoryCountry
        AND Original.TerritoryGroup = terr.TerritoryGroup
        AND Original.TerritoryRegion = terr.TerritoryRegion
    LEFT JOIN [CH01-01-Dimension].DimProduct AS prod ON prod.ProductName = Original.ProductName
    LEFT JOIN [CH01-01-Dimension].DimCustomer AS cust ON cust.CustomerName = Original.CustomerName
    LEFT JOIN [CH01-01-Dimension].DimOccupation AS occ ON occ.Occupation = Original.Occupation
    LEFT JOIN [CH01-01-Dimension].SalesManagers ON Original.SalesManager = SalesManagers.SalesManagerKey

    DECLARE @rowcount AS int;
    SET @rowcount = (SELECT COUNT(*) FROM [CH01-01-Fact].[Data]);
    SET @end = SYSDATETIME();

```

```

EXEC [Process].[usp_TrackWorkflow]
    @WorkflowDescription = N'[Project2].[Load_Data] loads data into [CH01-01-Fact].[
    @WorkflowStepTableRowCount = @rowcount,
    @UserAuthorization = @GroupMemberUserAuthorizationKey,
    @startTime = @start,
    @endTime = @end;

END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Damian Rozpedowski
-- Create date: 04-12-24
-- Description: Populates the DimCustomer table
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[Load_DimCustomer]
    @GroupMemberUserAuthorizationKey int
AS
BEGIN
    DECLARE @DateAdded DATETIME2;
    SET @DateAdded = SYSDATETIME();

    DECLARE @DateOfLastUpdate DATETIME2;
    SET @DateOfLastUpdate = SYSDATETIME();

    declare @start as datetime2, @end as datetime2;
    set @start = SYSDATETIME()
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;
    INSERT INTO [CH01-01-Dimension].[DimCustomer]
        (CustomerName, UserAuthorizationKey,
        DateAdded,
        DateOfLastUpdate)
    SELECT distinct Original.CustomerName, @GroupMemberUserAuthorizationKey,
        @DateAdded,
        @DateOfLastUpdate
    FROM FileUpload.OriginallyLoadedData AS Original left JOIN
        [Ch01-01-Dimension].DimCustomer AS New
        ON New.CustomerName = Original.CustomerName;

    declare @rowcount as int
    set @rowcount = (select count(*)
    from [CH01-01-Dimension].[DimCustomer]);
    set @end = SYSDATETIME()
    EXEC [Process].[usp_TrackWorkflow]
        @WorkflowDescription = N'[Project2].[Load_DimCustomer] loads data into [
        @WorkflowStepTableRowCount = @rowcount,
        @UserAuthorization = @GroupMemberUserAuthorizationKey,
        @startTime = @start,
        @endTime = @end

END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Hannah Kurian
-- Create date: 04-12-24
-- Description: Populates the DimGender table
-- =====
SET ANSI_NULLS ON

```

```

GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[Load_DimGender]
    @GroupMemberUserAuthorizationKey INT
AS
BEGIN
    DECLARE @DateAdded DATETIME2;
    SET @DateAdded = SYSDATETIME();

    DECLARE @DateOfLastUpdate DATETIME2;
    SET @DateOfLastUpdate = SYSDATETIME();

    DECLARE @start AS datetime2, @end AS datetime2;
    SET @start = SYSDATETIME();
    SET NOCOUNT ON;

    INSERT INTO [CH01-01-Dimension].[DimGender] (Gender, GenderDescription, UserAuthorizationKey,
        DateAdded,
        DateOfLastUpdate)
    SELECT DISTINCT
        Original.Gender,
        CASE Original.Gender
            WHEN 'M' THEN 'MALE'
            WHEN 'F' THEN 'FEMALE'
            ELSE 'UNKNOWN' -- Handles unexpected values
        END, @GroupMemberUserAuthorizationKey,
        @DateAdded,
        @DateOfLastUpdate
    FROM FileUpload.OriginallyLoadedData AS Original
    LEFT JOIN [CH01-01-Dimension].[DimGender] AS g ON g.Gender = Original.Gender
    WHERE g.Gender IS NULL; -- Avoid duplicates

    DECLARE @rowcount AS int;
    SET @rowcount = (SELECT COUNT(*) FROM [CH01-01-Dimension].[DimGender]);
    SET @end = SYSDATETIME();

    EXEC [Process].[usp_TrackWorkFlow]
        @WorkFlowDescription = N'[Project2].[Load_DimGender] loads data into [CH01-01-Di
        @WorkFlowStepTableRowCount = @rowcount,
        @UserAuthorization = @GroupMemberUserAuthorizationKey,
        @startTime = @start,
        @endTime = @end;

END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Hannah Kurian
-- Create date: 04-12-24
-- Description: Populates the DimMaritalStatus table
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[Load_DimMaritalStatus]
    @GroupMemberUserAuthorizationKey INT
AS
BEGIN
    DECLARE @DateAdded DATETIME2;
    SET @DateAdded = SYSDATETIME();

    DECLARE @DateOfLastUpdate DATETIME2;
    SET @DateOfLastUpdate = SYSDATETIME();

```

```

declare @start as datetime2, @end as datetime2;
set @start = SYSDATETIME()

-- SET NOCOUNT ON added to prevent extra result sets from
-- interfering with SELECT statements.
SET NOCOUNT ON;
INSERT INTO [CH01-01-Dimension].[DimMaritalStatus]
    (MaritalStatus, MaritalStatusDescription, UserAuthorizationKey,
    DateAdded,
    DateOfLastUpdate)
SELECT DISTINCT Orig.MaritalStatus, CASE Orig.MaritalStatus
    WHEN 'M' THEN 'Married'
    WHEN 'S' THEN 'Single'
    ELSE 'Unknown'
END AS MaritalStatusDescription,
    @GroupMemberUserAuthorizationKey,
    @DateAdded,
    @DateOfLastUpdate
FROM FileUpload.OriginallyLoadedData AS Orig LEFT JOIN
    [CH01-01-Dimension].[DimMaritalStatus] AS ms ON
    ms.MaritalStatus = Orig.MaritalStatus

declare @rowcount as int
set @rowcount = (select count(*)
from [CH01-01-Dimension].[DimMaritalStatus]);
set @end = SYSDATETIME()
EXEC [Process].[usp_TrackWorkFlow]
    @WorkFlowDescription = N'[Project2].[Load_DimMaritalStatus] loads data i
    @WorkFlowStepTableRowCount = @rowcount,
    @UserAuthorization = @GroupMemberUserAuthorizationKey,
    @startTime = @start,
    @endTime = @end

END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Hasnatul Hosna
-- Create date: 04-12-24
-- Description: Populates the DimOccupation table
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[Load_DimOccupation]
    @GroupMemberUserAuthorizationKey INT
AS
BEGIN
    DECLARE @DateAdded DATETIME2;
    SET @DateAdded = SYSDATETIME();

    DECLARE @DateOfLastUpdate DATETIME2;
    SET @DateOfLastUpdate = SYSDATETIME();

    declare @start as datetime2, @end as datetime2;
    set @start = SYSDATETIME()
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Inserting occupations with the correct OccupationKey based on the provided mappin
    INSERT INTO [CH01-01-Dimension].[DimOccupation] ([OccupationKey], [Occupation], User

```



```

        DateAdded,
        DateOfLastUpdate)
SELECT DISTINCT
    CASE A.Occupation
        WHEN 'Clerical' THEN 1
        WHEN 'Management' THEN 2
        WHEN 'Manual' THEN 3
        WHEN 'Professional' THEN 4
        WHEN 'Skilled Manual' THEN 5
        ELSE NULL -- Handling of unexpected occupations can be decided
    END,
    A.Occupation, @GroupMemberUserAuthorizationKey,
    @DateAdded,
    @DateOfLastUpdate
FROM FileUpload.OriginallyLoadedData AS A
LEFT JOIN [CH01-01-Dimension].DimOccupation AS oc ON oc.Occupation = A.Occupation
WHERE oc.Occupation IS NULL; -- Ensures no duplicates are inserted

declare @rowcount as int
set @rowcount = (SELECT COUNT(*) FROM [CH01-01-Dimension].[DimOccupation]);
set @end = SYSDATETIME()

-- Tracking the workflow
EXEC [Process].[usp_TrackWorkFlow]
    @WorkFlowDescription = N'[Project2].[Load_DimOccupation] loads data into [CH01-0
    @WorkFlowStepTableRowCount = @rowcount,
    @UserAuthorization = @GroupMemberUserAuthorizationKey,
    @startTime = @start,
    @endTime = @end
END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Hasnatul Hosna
-- Create date: 04-12-24
-- Description: Populates the DimOrderDate table
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[Load_DimOrderDate]
    @GroupMemberUserAuthorizationKey INT
AS
BEGIN
    DECLARE @DateAdded DATETIME2;
    SET @DateAdded = SYSDATETIME();

    DECLARE @DateOfLastUpdate DATETIME2;
    SET @DateOfLastUpdate = SYSDATETIME();

    declare @start as datetime2, @end as datetime2;
    set @start = SYSDATETIME()
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    insert into [CH01-01-Dimension].DimOrderDate
        (OrderDate, [MonthName], MonthNumber, [Year], UserAuthorizationKey,
        DateAdded,
        DateOfLastUpdate)
    select DISTINCT A.OrderDate, A.[MonthName], A.MonthNumber, A.[Year], @GroupMemb
        @DateAdded,
        @DateOfLastUpdate

```



```

from FileUpload.OriginallyLoadedData as A
    left join [CH01-01-Dimension].DimOrderDate as od
    ON od.OrderDate = A.OrderDate
        AND od.MonthNumber = A.MonthNumber
        AND od.[MonthName] = A.[MonthName]
        AND od.[Year] = A.[Year];

declare @rowcount as int
set @rowcount = (select count(*)
from [CH01-01-Dimension].DimOrderDate);
set @end = SYSDATETIME()
EXEC [Process].[usp_TrackWorkFlow]
    @WorkFlowDescription = N'[Project2].[Load_DimOrderDate] loads data into
    @WorkFlowStepTableRowCount = @rowcount,
    @UserAuthorization = @GroupMemberUserAuthorizationKey,
    @startTime = @start,
    @endTime = @end

END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Essmer Sanchez
-- Create date: 04-12-24
-- Description: Populates the DimProduct table
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[Load_DimProduct]
    @GroupMemberUserAuthorizationKey INT
AS
BEGIN
    DECLARE @DateAdded DATETIME2;
    SET @DateAdded = SYSDATETIME();

    DECLARE @DateOfLastUpdate DATETIME2;
    SET @DateOfLastUpdate = SYSDATETIME();

    DECLARE @start AS DATETIME2, @end AS DATETIME2;
    SET @start = SYSDATETIME();
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Inserting distinct ProductName and ProductCode from FileUpload.OriginallyLoadedDa
    INSERT INTO [CH01-01-Dimension].[DimProduct]
        (ProductKey, ProductCategory, ProductName, ProductSubcategory, ProductSubCategor
        DateAdded,
        DateOfLastUpdate)
    SELECT
        NEXT VALUE FOR [PkSequence].[DimProductSequenceObject],
        Original.ProductCategory,
        Original.ProductName,
        Original.ProductSubcategory,
        sub.ProductSubcategoryKey,
        Original.ModelName,
        Original.ProductCode,
        Original.Color, @GroupMemberUserAuthorizationKey,
        @DateAdded,
        @DateOfLastUpdate
    FROM
        (SELECT DISTINCT
            ProductCategory,

```

```

        ProductName,
        ProductSubcategory,
        ProductCode,
        Color,
        ModelName
    FROM FileUpload.OriginallyLoadedData) AS Original
LEFT JOIN [CH01-01-Dimension].DimProductSubcategory AS sub
    ON Original.ProductSubcategory = sub.ProductSubcategory;

DECLARE @rowcount AS INT;
SET @rowcount = (SELECT COUNT(*)
    FROM [CH01-01-Dimension].[DimProduct]);
SET @end = SYSDATETIME();

EXEC [Process].[usp_TrackWorkFlow]
    @WorkFlowDescription = N'[Project2].[Load_DimProduct] loads data into [CH01-01-D
    @WorkFlowStepTableRowCount = @rowcount,
    @UserAuthorization = @GroupMemberUserAuthorizationKey,
    @startTime = @start,
    @endTime = @end;

END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Essmer Sanchez
-- Create date: 04-12-24
-- Description: Populates the DimProductCategory table
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[Load_DimProductCategory]
    @GroupMemberUserAuthorizationKey INT
AS
BEGIN
    DECLARE @DateAdded DATETIME2;
    SET @DateAdded = SYSDATETIME();

    DECLARE @DateOfLastUpdate DATETIME2;
    SET @DateOfLastUpdate = SYSDATETIME();

    DECLARE @start AS datetime2, @end AS datetime2;
    SET @start = SYSDATETIME();
    SET NOCOUNT ON;

    -- Temporary table to store new ProductCategoryKeys
    IF OBJECT_ID('tempdb..#NewCategories') IS NOT NULL DROP TABLE #NewCategories;
    CREATE TABLE #NewCategories (
        ProductCategory NVARCHAR(50),
        ProductCategoryKey INT
    );

    -- Insert distinct categories with new keys into the temporary table
    INSERT INTO #NewCategories (ProductCategory, ProductCategoryKey)
    SELECT
        OLD.ProductCategory,
        NEXT VALUE FOR [PkSequence].[DimProductCategorySequenceObject]
    FROM (
        SELECT DISTINCT ProductCategory
        FROM FileUpload.OriginallyLoadedData
    ) AS OLD
    LEFT JOIN [CH01-01-Dimension].DimProductCategory AS NEW
        ON OLD.ProductCategory = NEW.ProductCategory

```

```

WHERE NEW.ProductCategory IS NULL;

-- Insert new distinct categories into DimProductCategory
INSERT INTO [CH01-01-Dimension].DimProductCategory
    ([ProductCategoryKey], [ProductCategory], UserAuthorizationKey,
    DateAdded,
    DateOfLastUpdate)
SELECT
    nc.ProductCategoryKey,
    nc.ProductCategory, @GroupMemberUserAuthorizationKey,
    @DateAdded,
    @DateOfLastUpdate
FROM #NewCategories AS nc;

DECLARE @rowcount AS int;
SET @rowcount = (SELECT COUNT(*) FROM [CH01-01-Dimension].DimProductCategory);
SET @end = SYSDATETIME();

EXEC [Process].[usp_TrackWorkFlow]
    @WorkFlowDescription = N'[Project2].[Load_DimProductCategory] loads data into [C
    @WorkFlowStepTableRowCount = @rowcount,
    @UserAuthorization = @GroupMemberUserAuthorizationKey,
    @startTime = @start,
    @endTime = @end;

-- Clean up temporary table
DROP TABLE #NewCategories;
END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Essmer Sanchez
-- Create date: 04-12-24
-- Description: Populates the DimProductSubcategory table
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[Load_DimProductSubcategory]
    @GroupMemberUserAuthorizationKey INT
AS
BEGIN
    DECLARE @DateAdded DATETIME2;
    SET @DateAdded = SYSDATETIME();

    DECLARE @DateOfLastUpdate DATETIME2;
    SET @DateOfLastUpdate = SYSDATETIME();

    DECLARE @start AS datetime2, @end AS datetime2;
    SET @start = SYSDATETIME();
    SET NOCOUNT ON;

    ;WITH DistinctSubcategories AS (
        SELECT DISTINCT
            OLD.ProductSubcategory,
            PC.ProductCategoryKey
        FROM FileUpload.OriginallyLoadedData AS OLD
        INNER JOIN [CH01-01-Dimension].DimProductCategory AS PC
            ON OLD.ProductCategory = PC.ProductCategory
        WHERE PC.ProductCategoryKey IS NOT NULL
    )
    -- Inserting product subcategories using the correct sequence for primary key genera
    INSERT INTO [CH01-01-Dimension].DimProductSubcategory

```

```

([ProductSubcategoryKey], [ProductSubcategory], [ProductCategoryKey], UserAuthor
DateAdded,
DateOfLastUpdate)
SELECT
    NEXT VALUE FOR PkSequence.DimProductSubcategorySequenceObject,
    sc.ProductSubcategory,
    sc.ProductCategoryKey, @GroupMemberUserAuthorizationKey,
        @DateAdded,
        @DateOfLastUpdate
FROM DistinctSubcategories AS sc;

DECLARE @rowcount AS int;
SET @rowcount = (SELECT COUNT(*) FROM [CH01-01-Dimension].DimProductSubcategory);
SET @end = SYSDATETIME();

EXEC [Process].[usp_TrackWorkFlow]
    @WorkFlowDescription = N'[Project2].[Load_DimProductSubcategory] loads data into
    @WorkFlowStepTableRowCount = @rowcount,
    @UserAuthorization = @GroupMemberUserAuthorizationKey,
    @startTime = @start,
    @endTime = @end;

END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Damian Rozpedowski
-- Create date: 04-12-24
-- Description: Populates the DimTerritory table
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[Load_DimTerritory]
    @GroupMemberUserAuthorizationKey INT
AS
BEGIN
    DECLARE @DateAdded DATETIME2;
    SET @DateAdded = SYSDATETIME();

    DECLARE @DateOfLastUpdate DATETIME2;
    SET @DateOfLastUpdate = SYSDATETIME();

    declare @start as datetime2, @end as datetime2;
    set @start = SYSDATETIME()
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;
    INSERT INTO [CH01-01-Dimension].[DimTerritory]
        (TerritoryGroup, TerritoryCountry, TerritoryRegion, UserAuthorizationKey,
        DateAdded,
        DateOfLastUpdate)
    SELECT distinct Orig.TerritoryGroup, Orig.TerritoryCountry, Orig.TerritoryRegion
        @DateAdded,
        @DateOfLastUpdate
    FROM FileUpload.OriginallyLoadedData AS Orig left JOIN
        [CH01-01-Dimension].[DimTerritory] AS ter
        ON ter.TerritoryCountry = Orig.TerritoryCountry
            AND ter.TerritoryRegion = Orig.TerritoryRegion
            AND ter.TerritoryGroup = Orig.TerritoryGroup;

    declare @rowcount as int
    set @rowcount = (select count(*)

```

```

        from [CH01-01-Dimension].[DimTerritory]);
    set @end = SYSDATETIME()
    EXEC [Process].[usp_TrackWorkFlow]
        @WorkFlowDescription = N'[Project2].[Load_DimTerritory] loads data into
        @WorkFlowStepTableRowCount = @rowcount,
        @UserAuthorization = @GroupMemberUserAuthorizationKey,
        @startTime = @start,
        @endTime = @end
END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Hannah Kurian
-- Create date: 04-12-24
-- Description: Populates the SalesManagers table
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[Load_SalesManagers]
    @GroupMemberUserAuthorizationKey INT
AS
BEGIN
    DECLARE @DateAdded DATETIME2;
    SET @DateAdded = SYSDATETIME();

    DECLARE @DateOfLastUpdate DATETIME2;
    SET @DateOfLastUpdate = SYSDATETIME();

    SET NOCOUNT ON;

    DECLARE @start AS DATETIME2 = SYSDATETIME();
    DECLARE @end AS DATETIME2;

    DECLARE @CategoryOrder TABLE (Category NVARCHAR(255), OrderKey INT);
    INSERT INTO @CategoryOrder (Category, OrderKey)
    VALUES
        ('Bike Racks', 1),
        ('Bike Stands', 2),
        ('Bottles and Cages', 3),
        ('Caps', 4),
        ('Cleaners', 5),
        ('Fenders', 6),
        ('Gloves', 7),
        ('Helmets', 8),
        ('Hydration Packs', 9),
        ('Jerseys', 10),
        ('Mountain Bikes', 11),
        ('Road Bikes', 12),
        ('Shorts', 13),
        ('Socks', 14),
        ('Tires and Tubes', 15),
        ('Touring Bikes', 16),
        ('Vests', 17);

    -- Inserting distinct new SalesManagers not currently in the database
    INSERT INTO [CH01-01-Dimension].[SalesManagers]
        ([SalesManagerKey], [SalesManager], [Category], [Office], UserAuthorizationKey,
        DateAdded,
        DateOfLastUpdate)
    SELECT
        C.OrderKey AS SalesManagerKey,

```

```

O.SalesManager,
O.ProductSubcategory AS Category,
CASE
    WHEN O.SalesManager IN ('Maurizio Macagno', 'Marco Russo') THEN 'Redmond'
    WHEN O.SalesManager IN ('Alberto Ferrari', 'Luis Bonifaz') THEN 'Seattle'
    ELSE 'Default Office'
END AS Office, @GroupMemberUserAuthorizationKey,
@DateAdded,
@DateOfLastUpdate
FROM (
    SELECT DISTINCT SalesManager, ProductSubcategory
    FROM FileUpload.OriginallyLoadedData
) AS O
JOIN @CategoryOrder C ON C.Category = O.ProductSubcategory
LEFT JOIN [CH01-01-Dimension].[SalesManagers] AS M
    ON M.SalesManager = O.SalesManager AND M.Category = O.ProductSubcategory
WHERE M.SalesManager IS NULL;

DECLARE @rowcount AS int;
SET @rowcount = (SELECT COUNT(*) FROM [CH01-01-Dimension].[SalesManagers]);
SET @end = SYSDATETIME();

EXEC [Process].[usp_TrackWorkFlow]
    @WorkFlowDescription = N'[Project2].[Load_SalesManagers] loads data into [CH01-01-Dimension].[SalesManagers]',
    @WorkFlowStepTableRowCount = @rowcount,
    @UserAuthorization = @GroupMemberUserAuthorizationKey,
    @startTime = @start,
    @endTime = @end;

END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Group 3 - Team Effort
-- Create date: 04-12-24
-- Description: Runs most of the procedures in order
--              to drop foreign keys, truncate the tables,
--              execute each tables load procedure, and adds
--              foreign keys to the newly populated tables.
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[LoadStarSchemaData]
-- Add the parameters for the stored procedure here
AS
BEGIN
    SET NOCOUNT ON;
    declare @start as datetime2, @end as datetime2;
    set @start = SYSDATETIME()
    --
    -- Drop All of the foreign keys prior to truncating tables in the star sche
    --
    EXEC [Project2].[DropForeignKeysFromStarSchemaData] @GroupMemberUserAuthorizati
    --
    -- Check row count before truncation
    EXEC [Project2].[ShowTableStatusRowCount]
        @GroupMemberUserAuthorizationKey = 2, -- Change -1 to the appropriate U
        @TableStatus = N'''Pre truncate of tables'''
    --
    -- Always truncate the Star Schema Data
    --
    EXEC [Project2].[TruncateStarSchemaData] @GroupMemberUserAuthorizationKey = 2;

```

```

--      Load the star schema
--
EXEC [Project2].[Load_DimProductCategory] @GroupMemberUserAuthorizationKey = 3;

EXEC [Project2].[Load_DimProductSubcategory] @GroupMemberUserAuthorizationKey =

EXEC [Project2].[Load_DimProduct] @GroupMemberUserAuthorizationKey = 3;

EXEC [Project2].[Load_SalesManagers] @GroupMemberUserAuthorizationKey = 4;

EXEC [Project2].[Load_DimGender] @GroupMemberUserAuthorizationKey = 4;

EXEC [Project2].[Load_DimMaritalStatus] @GroupMemberUserAuthorizationKey = 4;

EXEC [Project2].[Load_DimOccupation] @GroupMemberUserAuthorizationKey = 5;

EXEC [Project2].[Load_DimOrderDate] @GroupMemberUserAuthorizationKey = 5;

EXEC [Project2].[Load_DimTerritory] @GroupMemberUserAuthorizationKey = 2;

EXEC [Project2].[Load_DimCustomer] @GroupMemberUserAuthorizationKey = 2;

EXEC [Project2].[Load_Data] @GroupMemberUserAuthorizationKey = 1;
--      Recreate all of the foreign keys prior after loading the star schema
--
--
--      Check row count before truncation
EXEC [Project2].[ShowTableStatusRowCount]
    @GroupMemberUserAuthorizationKey = 1, -- Change -1 to the appropriate U
    @TableStatus = N'''Row Count after loading the star schema'''
--

EXEC [Project2].[AddForeignKeysToStarSchemaData] @GroupMemberUserAuthorizationKe
-- Change -1 to the appropriate UserAuthorizationKey

declare @rowcount as int
set @rowcount = 0
set @end = SYSDATETIME()
EXEC [Process].[usp_TrackWorkFlow]
    @WorkFlowDescription = N'[Project2].[DropForeignKeysFromStarSchemaData]
    @WorkFlowStepTableRowCount = @rowcount,
    @UserAuthorization = 1,
    @startTime = @start,
    @endTime = @end

--
END;

```

```

In [ ]: USE [BIClass]
GO
-- =====
-- Author:          Group 3 - Team Effort
-- Create date: 04-12-24
-- Description: Used to show how many rows have been
--              filled in the tables, displayed when
--              executed in LoadStarSchemaData
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[ShowTableStatusRowCount]
    @GroupMemberUserAuthorizationKey int,
    @TableStatus NVARCHAR(30)

```



```
AS
BEGIN
```

```
SET NOCOUNT ON;
select TableStatus = @TableStatus, TableName = 'CH01-01-Dimension.DimCustomer', C
FROM [CH01-01-Dimension].DimCustomer
select TableStatus = @TableStatus, TableName = 'CH01-01-Dimension.DimGender', COU
FROM [CH01-01-Dimension].DimGender
select TableStatus = @TableStatus, TableName = 'CH01-01-Dimension.DimMaritalStatu
FROM [CH01-01-Dimension].DimMaritalStatus
select TableStatus = @TableStatus, TableName = 'CH01-01-Dimension.DimOccupation',
FROM [CH01-01-Dimension].DimOccupation
select TableStatus = @TableStatus, TableName = 'CH01-01-Dimension.DimOrderDate',
FROM [CH01-01-Dimension].DimOrderDate
select TableStatus = @TableStatus, TableName = 'CH01-01-Dimension.DimProduct', CO
FROM [CH01-01-Dimension].DimProduct
select TableStatus = @TableStatus, TableName = 'CH01-01-Dimension.DimProductCateg
FROM [CH01-01-Dimension].DimProductCategory
select TableStatus = @TableStatus, TableName = 'CH01-01-Dimension.DimProductSubca
FROM [CH01-01-Dimension].DimProductSubcategory
select TableStatus = @TableStatus, TableName = 'CH01-01-Dimension.DimTerritory',
FROM [CH01-01-Dimension].DimTerritory
select TableStatus = @TableStatus, TableName = 'CH01-01-Dimension.SalesManagers',
FROM [CH01-01-Dimension].SalesManagers
select TableStatus = @TableStatus, TableName = 'CH01-01-Fact.Data', COUNT(*) as n
FROM [CH01-01-Fact].Data
```

```
END;
```

In []:

```
USE [BIClass]
GO
-- =====
-- Author:          Damian Rozpedowski
-- Create date: 04-12-24
-- Description: Truncates the data, in order to do that
--              foreign keys must have already been dropped.
--              Drops each sequence object and recreates them,
--              then truncates the tables and restarts sequences.
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [Project2].[TruncateStarSchemaData]
    @GroupMemberUserAuthorizationKey int

AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.

    SET NOCOUNT ON;
    declare @start1 as datetime2, @end1 as datetime2, @start as datetime2, @end as d
    set @start1 = SYSDATETIME()
    set @start = SYSDATETIME()

    DROP SEQUENCE [PkSequence].[DataSequenceObject]
    DROP SEQUENCE [PkSequence].[DimCustomerSequenceObject]
    DROP SEQUENCE [PkSequence].[DimGenderSequenceObject]
    DROP SEQUENCE [PkSequence].[DimMartialStatusSequenceObject]
    DROP SEQUENCE [PkSequence].[DimOccupationSequenceObject]
    DROP SEQUENCE [PkSequence].[DimOrderDateSequenceObject]
    DROP SEQUENCE [PkSequence].[DimProductCategorySequenceObject]
    DROP SEQUENCE [PkSequence].[DimProductSequenceObject]
```

```
DROP SEQUENCE [PkSequence].[DimProductSubCategorySequenceObject]
DROP SEQUENCE [PkSequence].[DimTerritorySequenceObject]
DROP SEQUENCE [PkSequence].[SalesManagersSequenceObject]
DROP SEQUENCE [PkSequence].[UserAuthorizationSequenceObject]
DROP SEQUENCE [PkSequence].[WorkFlowStepsSequenceObject]
```

```
CREATE SEQUENCE [PkSequence].[DimCustomerSequenceObject]
  AS [int]
  START WITH 1
  INCREMENT BY 1
  MINVALUE 1
  MAXVALUE 2147483647
  CACHE
```

```
CREATE SEQUENCE [PkSequence].[DimGenderSequenceObject]
  AS [int]
  START WITH 1
  INCREMENT BY 1
  MINVALUE 1
  MAXVALUE 2147483647
  CACHE
```

```
CREATE SEQUENCE [PkSequence].[DimMaritalStatusSequenceObject]
  AS [int]
  START WITH 1
  INCREMENT BY 1
  MINVALUE 1
  MAXVALUE 2147483647
  CACHE
```

```
CREATE SEQUENCE [PkSequence].[DimOccupationSequenceObject]
  AS [int]
  START WITH 1
  INCREMENT BY 1
  MINVALUE 1
  MAXVALUE 2147483647
  CACHE
```

```
CREATE SEQUENCE [PkSequence].[DimOrderDateSequenceObject]
  AS [int]
  START WITH 1
  INCREMENT BY 1
  MINVALUE 1
  MAXVALUE 2147483647
  CACHE
```

```
CREATE SEQUENCE [PkSequence].[DimProductSequenceObject]
  AS [int]
  START WITH 1
  INCREMENT BY 1
  MINVALUE 1
  MAXVALUE 2147483647
  CACHE
```

```
CREATE SEQUENCE [PkSequence].[DimProductCategorySequenceObject]
  AS [int]
  START WITH 1
  INCREMENT BY 1
  MINVALUE 1
  MAXVALUE 2147483647
  CACHE
```

```
CREATE SEQUENCE [PkSequence].[DimProductSubCategorySequenceObject]
  AS [int]
  START WITH 1
  INCREMENT BY 1
```

```
MINVALUE 1
MAXVALUE 2147483647
CACHE
```

```
CREATE SEQUENCE [PkSequence].[DimTerritorySequenceObject]
AS [int]
START WITH 1
INCREMENT BY 1
MINVALUE 1
MAXVALUE 2147483647
CACHE
```

```
CREATE SEQUENCE [PkSequence].[SalesManagersSequenceObject]
AS [int]
START WITH 1
INCREMENT BY 1
MINVALUE 1
MAXVALUE 2147483647
CACHE
```

```
CREATE SEQUENCE [PkSequence].[DataSequenceObject]
AS [int]
START WITH 1
INCREMENT BY 1
MINVALUE 1
MAXVALUE 2147483647
CACHE
```

```
CREATE SEQUENCE [PkSequence].[UserAuthorizationSequenceObject]
AS [int]
START WITH 1
INCREMENT BY 1
MINVALUE 1
MAXVALUE 2147483647
CACHE
```

```
CREATE SEQUENCE [PkSequence].[WorkflowStepsSequenceObject]
AS [int]
START WITH 1
INCREMENT BY 1
MINVALUE 1
MAXVALUE 2147483647
CACHE
```

```
truncate table [CH01-01-Fact].data;
ALTER SEQUENCE [PkSequence].[DataSequenceObject]
    RESTART WITH 1;
truncate table [CH01-01-Dimension].SalesManagers;
ALTER SEQUENCE [PkSequence].[SalesManagersSequenceObject]
    RESTART WITH 1;
truncate table [CH01-01-Dimension].DimProductSubcategory;
ALTER SEQUENCE [PkSequence].[DimProductSubCategorySequenceObject]
    RESTART WITH 1;
truncate table [CH01-01-Dimension].DimProductCategory;
ALTER SEQUENCE [PkSequence].[DimProductCategorySequenceObject]
    RESTART WITH 1;
truncate table [CH01-01-Dimension].DimGender;
ALTER SEQUENCE [PkSequence].[DimGenderSequenceObject]
    RESTART WITH 1;
truncate table [CH01-01-Dimension].DimMaritalStatus;
ALTER SEQUENCE [PkSequence].[DimMaritalStatusSequenceObject]
    RESTART WITH 1;
truncate table [CH01-01-Dimension].DimOccupation;
ALTER SEQUENCE [PkSequence].[DimOccupationSequenceObject]
    RESTART WITH 1;
```

```

truncate table [CH01-01-Dimension].DimOrderDate;
ALTER SEQUENCE [PkSequence].[DimOrderDateSequenceObject]
    RESTART WITH 1;
truncate table [CH01-01-Dimension].DimTerritory;
ALTER SEQUENCE [PkSequence].[DimTerritorySequenceObject]
    RESTART WITH 1;
truncate table [CH01-01-Dimension].DimProduct;
ALTER SEQUENCE [PkSequence].[DimProductSequenceObject]
    RESTART WITH 1;
truncate table [CH01-01-Dimension].DimCustomer;
ALTER SEQUENCE [PkSequence].[DimCustomerSequenceObject]
    RESTART WITH 1;
truncate table [Process].[WorkflowSteps];
ALTER SEQUENCE [PkSequence].[WorkflowStepsSequenceObject]
    RESTART WITH 1;

declare @rowcount as int
set @rowcount = 0
set @end1 = SYSDATETIME()
EXEC [Process].[usp_TrackWorkFlow]
    @WorkFlowDescription = N'[Project2].[DropForeignKeysFromStarSchemaData]
    @WorkFlowStepTableRowCount = @rowcount,
    @UserAuthorization = @GroupMemberUserAuthorizationKey,
    @startTime = @start1,
    @endTime = @end1

set @rowcount = 0
set @end = SYSDATETIME()
EXEC [Process].[usp_TrackWorkFlow]
    @WorkFlowDescription = N'[Project2].[TruncateStarSchemaData] Truncated a
    @WorkFlowStepTableRowCount = @rowcount,
    @UserAuthorization = @GroupMemberUserAuthorizationKey,
    @startTime = @start,
    @endTime = @end
end;

```