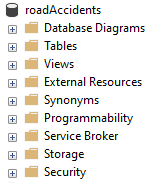
**SQL Analysis**

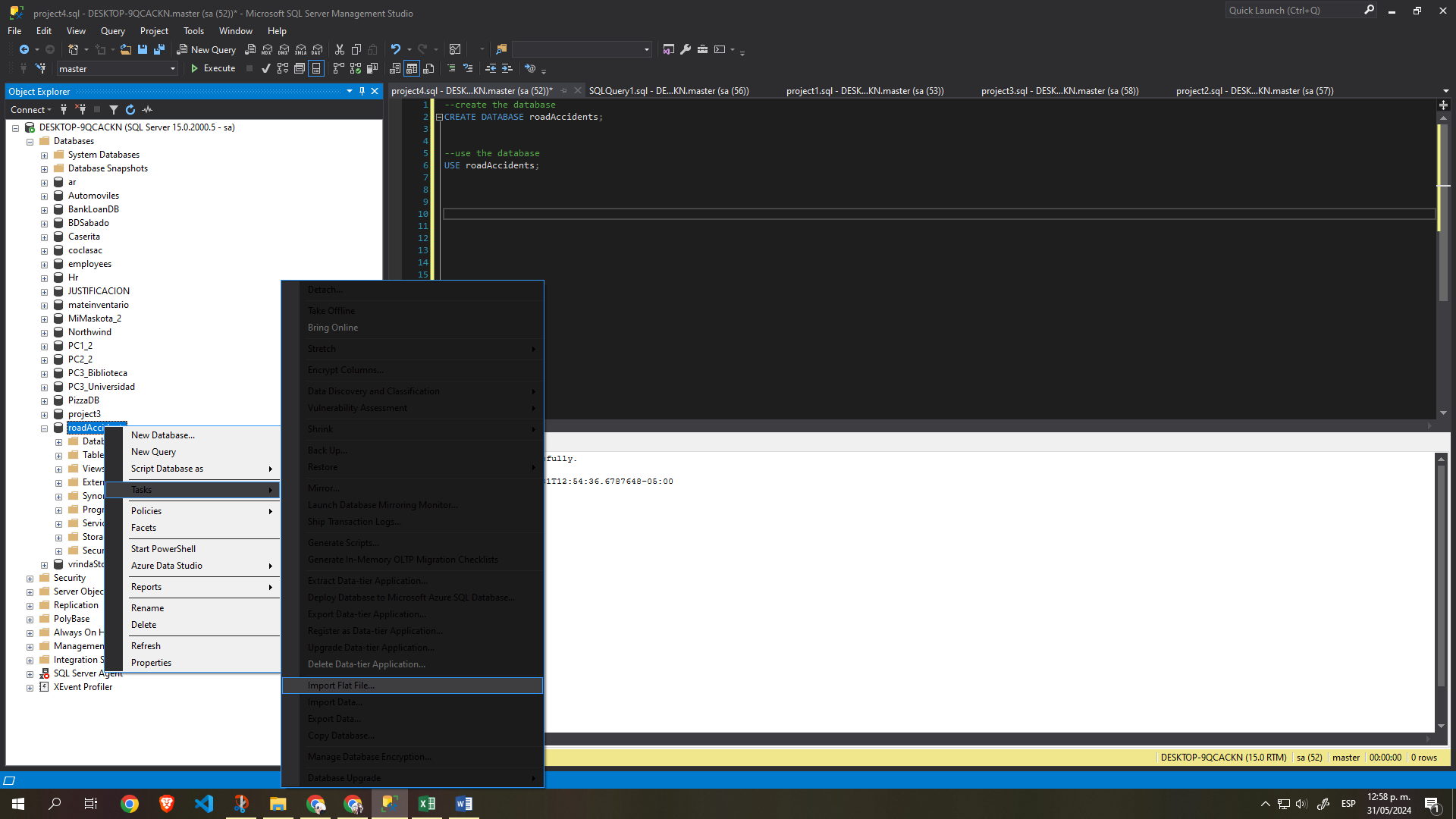
--create the database

CREATE DATABASE roadAccidents;



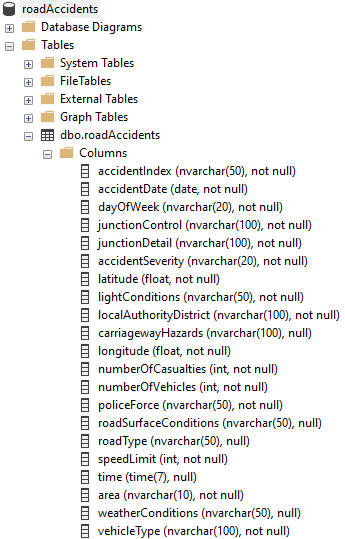
--use the database

USE roadAccidents;

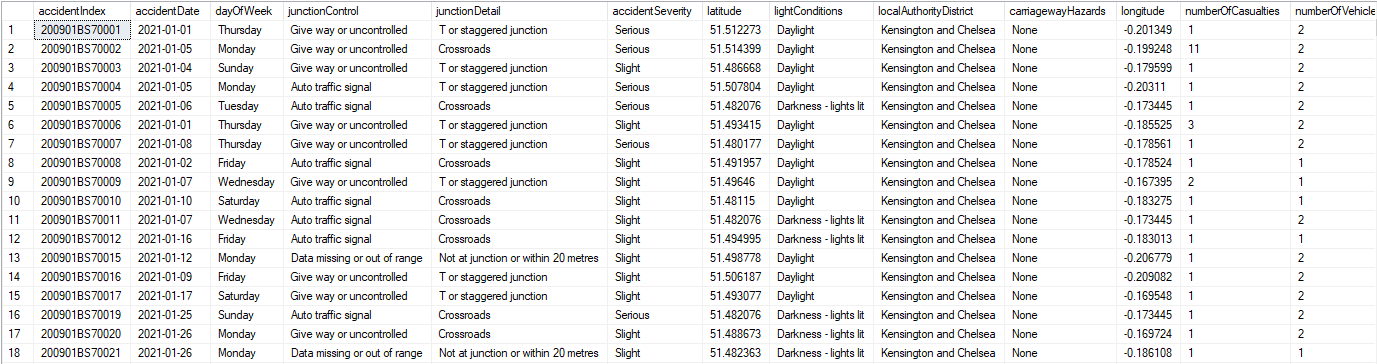


--we import the csv file into a new table in the database

--show the table



SELECT \* FROM roadAccidents;



--analizing the field accident severity

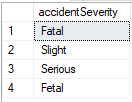
--we need to correct the misspelling

SELECT

DISTINCT accidentSeverity

FROM

roadAccidents



--put fatal instead of fetal

BEGIN TRAN

UPDATE

[dbo].[roadAccidents]

SET

accidentSeverity =

CASE

WHEN accidentSeverity = 'Fetal' THEN 'Fatal'

ELSE accidentSeverity

END

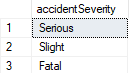
SELECT

DISTINCT accidentSeverity

FROM

roadAccidents

COMMIT TRAN



--adding new columns

BEGIN TRAN

ALTER TABLE

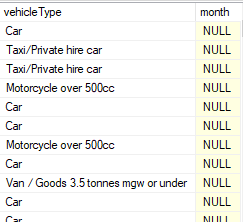
[dbo].[roadAccidents]

ADD

[month] NVARCHAR(10);

SELECT \* FROM roadAccidents;

COMMIT TRAN



--adding new columns

BEGIN TRAN

ALTER TABLE

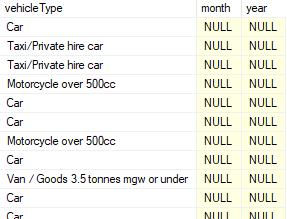
[dbo].[roadAccidents]

ADD

[year] INT;

SELECT \* FROM roadAccidents;

COMMIT TRAN



--filling the month column

BEGIN TRAN

UPDATE

roadAccidents

SET

month =

FORMAT(

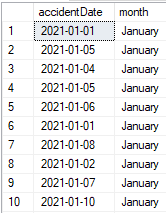
accidentDate,

'MMMM'

)

SELECT accidentDate, month FROM roadAccidents;

COMMIT TRAN



--filling the year column

BEGIN TRAN

UPDATE

roadAccidents

SET

year =

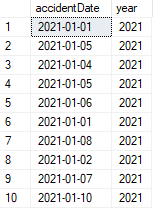
YEAR(

accidentDate

)

SELECT accidentDate, year FROM roadAccidents;

COMMIT TRAN



--base query

SELECT

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

--knowing the type and distinct values that can be send, we create a variable for each one of them-----------

--year variable:

--if you want the three years 2021, 2022, 2023, dont sent any year variable

--if you want just two years, sent the two years

--if you want one year, sent one year

--if you want more field to filter

--it is two cases (IF and ELSE) for a field aggregation like for month, etc.

CREATE PROCEDURE sumOfCasualties

@year1 INT = NULL,

@area1 NVARCHAR(10) = NULL

AS

BEGIN

IF (@year1 IS NULL)

BEGIN

IF (@area1 IS NULL)

BEGIN

SELECT

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

END

ELSE

BEGIN

SELECT

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

WHERE

area = @area1

END

END

ELSE

BEGIN

IF (@area1 IS NULL)

BEGIN

SELECT

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

WHERE

year = @year1

END

ELSE

BEGIN

SELECT

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

WHERE

year = @year1

AND

area = @area1

END

END

END

--combinations you can do

EXEC sumOfCasualties

EXEC sumOfCasualties @area1 = 'Rural'

EXEC sumOfCasualties @year1 = 2021

EXEC sumOfCasualties @year1 = 2022, @area1 = 'Rural'



--base query

DECLARE @sumTotalCasualties INT;

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents);

PRINT @sumTotalCasualties;

SELECT

accidentSeverity,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST((SUM( CAST(numberOfCasualties AS FLOAT) ) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidents

GROUP BY

accidentSeverity

--casualties by accident severity and percentages

CREATE PROCEDURE sumOfCasualtiesByAccidentSeverityAndPercentages

@year1 INT = NULL,

@area1 NVARCHAR(10) = NULL

AS

BEGIN

--we declare a variable for total casualties for later get the percentages

DECLARE @sumTotalCasualties INT;

IF (@year1 IS NULL)

BEGIN

IF (@area1 IS NULL)

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents);

SELECT

accidentSeverity,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidents

GROUP BY

accidentSeverity

END

ELSE

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE area = @area1);

SELECT

accidentSeverity,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidents

WHERE

area = @area1

GROUP BY

accidentSeverity

END

END

ELSE

BEGIN

IF (@area1 IS NULL)

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE year = @year1);

SELECT

accidentSeverity,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidents

WHERE

year = @year1

GROUP BY

accidentSeverity

END

ELSE

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE year = @year1 AND area = @area1);

SELECT

accidentSeverity,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidents

WHERE

year = @year1

AND

area = @area1

GROUP BY

accidentSeverity

END

END

END

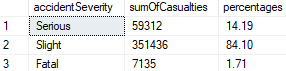
--combinations you can do

EXEC sumOfCasualtiesByAccidentSeverityAndPercentages

EXEC sumOfCasualtiesByAccidentSeverityAndPercentages @area1 = 'Rural'

EXEC sumOfCasualtiesByAccidentSeverityAndPercentages @year1 = 2021

EXEC sumOfCasualtiesByAccidentSeverityAndPercentages @year1 = 2022, @area1 = 'Rural'



--analyse the different types of vehicles, we can group some of them

--Cars: Car + Taxi/Private hire car, etc.

SELECT

DISTINCT vehicleType

FROM

roadAccidents

ORDER BY

vehicleType ASC

--base queries--------------

--getting the total for later

DECLARE @sumTotalCasualties INT;

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents);

PRINT @sumTotalCasualties;

--grouping

WITH roadAccidentsWithTypeGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN vehicleType = 'Car' THEN 'Cars'

WHEN vehicleType = 'Taxi/Private hire car' THEN 'Cars'

WHEN vehicleType = 'Bus or coach (17 or more pass seats)' THEN 'Bus'

WHEN vehicleType = 'Minibus (8 - 16 passenger seats)' THEN 'Bus'

WHEN vehicleType = 'Goods 7.5 tonnes mgw and over' THEN 'Van'

WHEN vehicleType = 'Goods over 3.5t. and under 7.5t' THEN 'Van'

WHEN vehicleType = 'Van / Goods 3.5 tonnes mgw or under' THEN 'Van'

WHEN vehicleType = 'Motorcycle 125cc and under' THEN 'Bike'

WHEN vehicleType = 'Motorcycle 50cc and under' THEN 'Bike'

WHEN vehicleType = 'Motorcycle over 125cc and up to 500cc' THEN 'Bike'

WHEN vehicleType = 'Motorcycle over 500cc' THEN 'Bike'

WHEN vehicleType = 'Other vehicle' THEN 'Others'

WHEN vehicleType = 'Pedal cycle' THEN 'Others'

WHEN vehicleType = 'Ridden horse' THEN 'Others'

ELSE vehicleType

END

AS newVehicleType

FROM

roadAccidents

)

SELECT

newVehicleType,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithTypeGroups

GROUP BY

newVehicleType

ORDER BY

sumOfCasualties DESC

--casualties by accident severity and percentages

CREATE PROCEDURE sumOfCasualtiesByVehicleTypeAndPercentages

@year1 INT = NULL,

@area1 NVARCHAR(10) = NULL

AS

BEGIN

--we declare a variable for total casualties for later get the percentages

DECLARE @sumTotalCasualties INT;

IF (@year1 IS NULL)

BEGIN

IF (@area1 IS NULL)

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents);

WITH roadAccidentsWithTypeGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN vehicleType = 'Car' THEN 'Cars'

WHEN vehicleType = 'Taxi/Private hire car' THEN 'Cars'

WHEN vehicleType = 'Bus or coach (17 or more pass seats)' THEN 'Bus'

WHEN vehicleType = 'Minibus (8 - 16 passenger seats)' THEN 'Bus'

WHEN vehicleType = 'Goods 7.5 tonnes mgw and over' THEN 'Van'

WHEN vehicleType = 'Goods over 3.5t. and under 7.5t' THEN 'Van'

WHEN vehicleType = 'Van / Goods 3.5 tonnes mgw or under' THEN 'Van'

WHEN vehicleType = 'Motorcycle 125cc and under' THEN 'Bike'

WHEN vehicleType = 'Motorcycle 50cc and under' THEN 'Bike'

WHEN vehicleType = 'Motorcycle over 125cc and up to 500cc' THEN 'Bike'

WHEN vehicleType = 'Motorcycle over 500cc' THEN 'Bike'

WHEN vehicleType = 'Other vehicle' THEN 'Others'

WHEN vehicleType = 'Pedal cycle' THEN 'Others'

WHEN vehicleType = 'Ridden horse' THEN 'Others'

ELSE vehicleType

END

AS newVehicleType

FROM

roadAccidents

)

SELECT

newVehicleType,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithTypeGroups

GROUP BY

newVehicleType

ORDER BY

sumOfCasualties DESC

END

ELSE

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE area = @area1);

WITH roadAccidentsWithTypeGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN vehicleType = 'Car' THEN 'Cars'

WHEN vehicleType = 'Taxi/Private hire car' THEN 'Cars'

WHEN vehicleType = 'Bus or coach (17 or more pass seats)' THEN 'Bus'

WHEN vehicleType = 'Minibus (8 - 16 passenger seats)' THEN 'Bus'

WHEN vehicleType = 'Goods 7.5 tonnes mgw and over' THEN 'Van'

WHEN vehicleType = 'Goods over 3.5t. and under 7.5t' THEN 'Van'

WHEN vehicleType = 'Van / Goods 3.5 tonnes mgw or under' THEN 'Van'

WHEN vehicleType = 'Motorcycle 125cc and under' THEN 'Bike'

WHEN vehicleType = 'Motorcycle 50cc and under' THEN 'Bike'

WHEN vehicleType = 'Motorcycle over 125cc and up to 500cc' THEN 'Bike'

WHEN vehicleType = 'Motorcycle over 500cc' THEN 'Bike'

WHEN vehicleType = 'Other vehicle' THEN 'Others'

WHEN vehicleType = 'Pedal cycle' THEN 'Others'

WHEN vehicleType = 'Ridden horse' THEN 'Others'

ELSE vehicleType

END

AS newVehicleType

FROM

roadAccidents

WHERE

area = @area1

)

SELECT

newVehicleType,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithTypeGroups

GROUP BY

newVehicleType

ORDER BY

sumOfCasualties DESC

END

END

ELSE

BEGIN

IF (@area1 IS NULL)

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE year = @year1);

WITH roadAccidentsWithTypeGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN vehicleType = 'Car' THEN 'Cars'

WHEN vehicleType = 'Taxi/Private hire car' THEN 'Cars'

WHEN vehicleType = 'Bus or coach (17 or more pass seats)' THEN 'Bus'

WHEN vehicleType = 'Minibus (8 - 16 passenger seats)' THEN 'Bus'

WHEN vehicleType = 'Goods 7.5 tonnes mgw and over' THEN 'Van'

WHEN vehicleType = 'Goods over 3.5t. and under 7.5t' THEN 'Van'

WHEN vehicleType = 'Van / Goods 3.5 tonnes mgw or under' THEN 'Van'

WHEN vehicleType = 'Motorcycle 125cc and under' THEN 'Bike'

WHEN vehicleType = 'Motorcycle 50cc and under' THEN 'Bike'

WHEN vehicleType = 'Motorcycle over 125cc and up to 500cc' THEN 'Bike'

WHEN vehicleType = 'Motorcycle over 500cc' THEN 'Bike'

WHEN vehicleType = 'Other vehicle' THEN 'Others'

WHEN vehicleType = 'Pedal cycle' THEN 'Others'

WHEN vehicleType = 'Ridden horse' THEN 'Others'

ELSE vehicleType

END

AS newVehicleType

FROM

roadAccidents

WHERE

year = @year1

)

SELECT

newVehicleType,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithTypeGroups

GROUP BY

newVehicleType

ORDER BY

sumOfCasualties DESC

END

ELSE

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE year = @year1 AND area = @area1);

WITH roadAccidentsWithTypeGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN vehicleType = 'Car' THEN 'Cars'

WHEN vehicleType = 'Taxi/Private hire car' THEN 'Cars'

WHEN vehicleType = 'Bus or coach (17 or more pass seats)' THEN 'Bus'

WHEN vehicleType = 'Minibus (8 - 16 passenger seats)' THEN 'Bus'

WHEN vehicleType = 'Goods 7.5 tonnes mgw and over' THEN 'Van'

WHEN vehicleType = 'Goods over 3.5t. and under 7.5t' THEN 'Van'

WHEN vehicleType = 'Van / Goods 3.5 tonnes mgw or under' THEN 'Van'

WHEN vehicleType = 'Motorcycle 125cc and under' THEN 'Bike'

WHEN vehicleType = 'Motorcycle 50cc and under' THEN 'Bike'

WHEN vehicleType = 'Motorcycle over 125cc and up to 500cc' THEN 'Bike'

WHEN vehicleType = 'Motorcycle over 500cc' THEN 'Bike'

WHEN vehicleType = 'Other vehicle' THEN 'Others'

WHEN vehicleType = 'Pedal cycle' THEN 'Others'

WHEN vehicleType = 'Ridden horse' THEN 'Others'

ELSE vehicleType

END

AS newVehicleType

FROM

roadAccidents

WHERE

year = @year1

AND

area = @area1

)

SELECT

newVehicleType,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithTypeGroups

GROUP BY

newVehicleType

ORDER BY

sumOfCasualties DESC

END

END

END

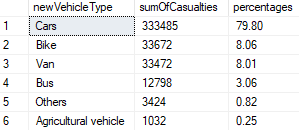
--combinations you can do

EXEC sumOfCasualtiesByVehicleTypeAndPercentages

EXEC sumOfCasualtiesByVehicleTypeAndPercentages @area1 = 'Rural'

EXEC sumOfCasualtiesByVehicleTypeAndPercentages @year1 = 2021

EXEC sumOfCasualtiesByVehicleTypeAndPercentages @year1 = 2022, @area1 = 'Rural'



--base query

SELECT

month,

year,

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

GROUP BY

month,

year

ORDER BY

month ASC,

year ASC

CREATE PROCEDURE sumOfCasualtiesByMonthAndYear

@year1 INT = NULL,

@area1 NVARCHAR(10) = NULL

AS

BEGIN

IF (@year1 IS NULL)

BEGIN

IF (@area1 IS NULL)

BEGIN

SELECT

month,

year,

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

GROUP BY

month,

year

ORDER BY

month ASC,

year ASC

END

ELSE

BEGIN

SELECT

month,

year,

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

WHERE

area = @area1

GROUP BY

month,

year

ORDER BY

month ASC,

year ASC

END

END

ELSE

BEGIN

IF (@area1 IS NULL)

BEGIN

SELECT

month,

year,

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

WHERE

year = @year1

GROUP BY

month,

year

ORDER BY

month ASC,

year ASC

END

ELSE

BEGIN

SELECT

month,

year,

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

WHERE

year = @year1

AND

area = @area1

GROUP BY

month,

year

ORDER BY

month ASC,

year ASC

END

END

END

--combinations you can do

EXEC sumOfCasualtiesByMonthAndYear

EXEC sumOfCasualtiesByMonthAndYear @area1 = 'Rural'

EXEC sumOfCasualtiesByMonthAndYear @year1 = 2021

EXEC sumOfCasualtiesByMonthAndYear @year1 = 2021, @area1 = 'Rural'



--base query

SELECT

roadType,

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

GROUP BY

roadType

ORDER BY

sumOfCasualties DESC

CREATE PROCEDURE sumOfCasualtiesByRoadType

@year1 INT = NULL,

@area1 NVARCHAR(10) = NULL

AS

BEGIN

IF (@year1 IS NULL)

BEGIN

IF (@area1 IS NULL)

BEGIN

SELECT

roadType,

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

GROUP BY

roadType

ORDER BY

sumOfCasualties DESC

END

ELSE

BEGIN

SELECT

roadType,

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

WHERE

area = @area1

GROUP BY

roadType

ORDER BY

sumOfCasualties DESC

END

END

ELSE

BEGIN

IF (@area1 IS NULL)

BEGIN

SELECT

roadType,

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

WHERE

year = @year1

GROUP BY

roadType

ORDER BY

sumOfCasualties DESC

END

ELSE

BEGIN

SELECT

roadType,

SUM(numberOfCasualties) AS sumOfCasualties

FROM

roadAccidents

WHERE

year = @year1

AND

area = @area1

GROUP BY

roadType

ORDER BY

sumOfCasualties DESC

END

END

END

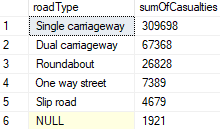
--combinations you can do

EXEC sumOfCasualtiesByRoadType

EXEC sumOfCasualtiesByRoadType @area1 = 'Rural'

EXEC sumOfCasualtiesByRoadType @year1 = 2021

EXEC sumOfCasualtiesByRoadType @year1 = 2021, @area1 = 'Rural'



------------------------------------------------------------------------------------

--analyse the different types of vehicles, we can group some of them

--Cars: Car + Taxi/Private hire car, etc.

SELECT

DISTINCT roadSurfaceConditions

FROM

roadAccidents

--base queries--------------

--getting the total for later

DECLARE @sumTotalCasualties INT;

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents);

PRINT @sumTotalCasualties;

--grouping

WITH roadAccidentsWithRoadSurfaceGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN roadSurfaceConditions = 'Flood over 3cm. deep' THEN 'Wet'

WHEN roadSurfaceConditions = 'Wet or damp' THEN 'Wet'

WHEN roadSurfaceConditions = 'Snow' THEN 'Snow - Ice'

WHEN roadSurfaceConditions = 'Frost or ice' THEN 'Snow - Ice'

ELSE roadSurfaceConditions

END

AS newRoadSurfaceConditions

FROM

roadAccidents

)

SELECT

newRoadSurfaceConditions,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithRoadSurfaceGroups

GROUP BY

newRoadSurfaceConditions

ORDER BY

sumOfCasualties DESC

--casualties by furface conditions severity and percentages

CREATE PROCEDURE sumOfCasualtiesByRoadSurfaceConditionsAndPercentages

@year1 INT = NULL,

@area1 NVARCHAR(10) = NULL

AS

BEGIN

--we declare a variable for total casualties for later get the percentages

DECLARE @sumTotalCasualties INT;

IF (@year1 IS NULL)

BEGIN

IF (@area1 IS NULL)

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents);

WITH roadAccidentsWithRoadSurfaceGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN roadSurfaceConditions = 'Flood over 3cm. deep' THEN 'Wet'

WHEN roadSurfaceConditions = 'Wet or damp' THEN 'Wet'

WHEN roadSurfaceConditions = 'Snow' THEN 'Snow - Ice'

WHEN roadSurfaceConditions = 'Frost or ice' THEN 'Snow - Ice'

ELSE roadSurfaceConditions

END

AS newRoadSurfaceConditions

FROM

roadAccidents

)

SELECT

newRoadSurfaceConditions,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithRoadSurfaceGroups

GROUP BY

newRoadSurfaceConditions

ORDER BY

sumOfCasualties DESC

END

ELSE

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE area = @area1);

WITH roadAccidentsWithRoadSurfaceGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN roadSurfaceConditions = 'Flood over 3cm. deep' THEN 'Wet'

WHEN roadSurfaceConditions = 'Wet or damp' THEN 'Wet'

WHEN roadSurfaceConditions = 'Snow' THEN 'Snow - Ice'

WHEN roadSurfaceConditions = 'Frost or ice' THEN 'Snow - Ice'

ELSE roadSurfaceConditions

END

AS newRoadSurfaceConditions

FROM

roadAccidents

WHERE

area = @area1

)

SELECT

newRoadSurfaceConditions,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithRoadSurfaceGroups

GROUP BY

newRoadSurfaceConditions

ORDER BY

sumOfCasualties DESC

END

END

ELSE

BEGIN

IF (@area1 IS NULL)

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE year = @year1);

WITH roadAccidentsWithRoadSurfaceGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN roadSurfaceConditions = 'Flood over 3cm. deep' THEN 'Wet'

WHEN roadSurfaceConditions = 'Wet or damp' THEN 'Wet'

WHEN roadSurfaceConditions = 'Snow' THEN 'Snow - Ice'

WHEN roadSurfaceConditions = 'Frost or ice' THEN 'Snow - Ice'

ELSE roadSurfaceConditions

END

AS newRoadSurfaceConditions

FROM

roadAccidents

WHERE

year = @year1

)

SELECT

newRoadSurfaceConditions,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithRoadSurfaceGroups

GROUP BY

newRoadSurfaceConditions

ORDER BY

sumOfCasualties DESC

END

ELSE

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE year = @year1 AND area = @area1);

WITH roadAccidentsWithRoadSurfaceGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN roadSurfaceConditions = 'Flood over 3cm. deep' THEN 'Wet'

WHEN roadSurfaceConditions = 'Wet or damp' THEN 'Wet'

WHEN roadSurfaceConditions = 'Snow' THEN 'Snow - Ice'

WHEN roadSurfaceConditions = 'Frost or ice' THEN 'Snow - Ice'

ELSE roadSurfaceConditions

END

AS newRoadSurfaceConditions

FROM

roadAccidents

WHERE

year = @year1

AND

area = @area1

)

SELECT

newRoadSurfaceConditions,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithRoadSurfaceGroups

GROUP BY

newRoadSurfaceConditions

ORDER BY

sumOfCasualties DESC

END

END

END

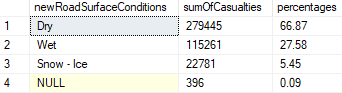
--combinations you can do

EXEC sumOfCasualtiesByRoadSurfaceConditionsAndPercentages

EXEC sumOfCasualtiesByRoadSurfaceConditionsAndPercentages @area1 = 'Rural'

EXEC sumOfCasualtiesByRoadSurfaceConditionsAndPercentages @year1 = 2021

EXEC sumOfCasualtiesByRoadSurfaceConditionsAndPercentages @year1 = 2021, @area1 = 'Rural'



--base query

DECLARE @sumTotalCasualties INT;

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents);

PRINT @sumTotalCasualties;

SELECT

area,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST((SUM( CAST(numberOfCasualties AS FLOAT) ) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidents

GROUP BY

area

--casualties by accident area and percentages

CREATE PROCEDURE sumOfCasualtiesByAreaAndPercentages

@year1 INT = NULL

AS

BEGIN

--we declare a variable for total casualties for later get the percentages

DECLARE @sumTotalCasualties INT;

IF (@year1 IS NULL)

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents);

SELECT

area,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidents

GROUP BY

area

END

ELSE

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE year = @year1);

SELECT

area,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidents

WHERE

year = @year1

GROUP BY

area

END

END

--combinations you can do

EXEC sumOfCasualtiesByAreaAndPercentages

EXEC sumOfCasualtiesByAreaAndPercentages @year1 = 2021



------------------------

SELECT

DISTINCT lightConditions

FROM

roadAccidents

--base queries--------------

--getting the total for later

DECLARE @sumTotalCasualties INT;

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents);

PRINT @sumTotalCasualties;

--grouping

WITH roadAccidentsWithLightGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN lightConditions = 'Darkness - lights lit' THEN 'Darkness'

WHEN lightConditions = 'Darkness - lighting unknown' THEN 'Darkness'

WHEN lightConditions = 'Darkness - lights unlit' THEN 'Darkness'

WHEN lightConditions = 'Darkness - no lighting' THEN 'Darkness'

ELSE lightConditions

END

AS newLightConditions

FROM

roadAccidents

)

SELECT

newLightConditions,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithLightGroups

GROUP BY

newLightConditions

ORDER BY

sumOfCasualties DESC

--casualties by light conditions and percentages

CREATE PROCEDURE sumOfCasualtiesByLightConditionsAndPercentages

@year1 INT = NULL,

@area1 NVARCHAR(10) = NULL

AS

BEGIN

--we declare a variable for total casualties for later get the percentages

DECLARE @sumTotalCasualties INT;

IF (@year1 IS NULL)

BEGIN

IF (@area1 IS NULL)

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents);

WITH roadAccidentsWithLightGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN lightConditions = 'Darkness - lights lit' THEN 'Darkness'

WHEN lightConditions = 'Darkness - lighting unknown' THEN 'Darkness'

WHEN lightConditions = 'Darkness - lights unlit' THEN 'Darkness'

WHEN lightConditions = 'Darkness - no lighting' THEN 'Darkness'

ELSE lightConditions

END

AS newLightConditions

FROM

roadAccidents

)

SELECT

newLightConditions,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithLightGroups

GROUP BY

newLightConditions

ORDER BY

sumOfCasualties DESC

END

ELSE

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE area = @area1);

WITH roadAccidentsWithLightGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN lightConditions = 'Darkness - lights lit' THEN 'Darkness'

WHEN lightConditions = 'Darkness - lighting unknown' THEN 'Darkness'

WHEN lightConditions = 'Darkness - lights unlit' THEN 'Darkness'

WHEN lightConditions = 'Darkness - no lighting' THEN 'Darkness'

ELSE lightConditions

END

AS newLightConditions

FROM

roadAccidents

WHERE

area = @area1

)

SELECT

newLightConditions,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithLightGroups

GROUP BY

newLightConditions

ORDER BY

sumOfCasualties DESC

END

END

ELSE

BEGIN

IF (@area1 IS NULL)

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE year = @year1);

WITH roadAccidentsWithLightGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN lightConditions = 'Darkness - lights lit' THEN 'Darkness'

WHEN lightConditions = 'Darkness - lighting unknown' THEN 'Darkness'

WHEN lightConditions = 'Darkness - lights unlit' THEN 'Darkness'

WHEN lightConditions = 'Darkness - no lighting' THEN 'Darkness'

ELSE lightConditions

END

AS newLightConditions

FROM

roadAccidents

WHERE

year = @year1

)

SELECT

newLightConditions,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithLightGroups

GROUP BY

newLightConditions

ORDER BY

sumOfCasualties DESC

END

ELSE

BEGIN

--set the variable

SET @sumTotalCasualties = (SELECT SUM(numberOfCasualties) FROM roadAccidents WHERE year = @year1 AND area = @area1);

WITH roadAccidentsWithLightGroups AS (

SELECT

numberOfCasualties,

CASE

WHEN lightConditions = 'Darkness - lights lit' THEN 'Darkness'

WHEN lightConditions = 'Darkness - lighting unknown' THEN 'Darkness'

WHEN lightConditions = 'Darkness - lights unlit' THEN 'Darkness'

WHEN lightConditions = 'Darkness - no lighting' THEN 'Darkness'

ELSE lightConditions

END

AS newLightConditions

FROM

roadAccidents

WHERE

year = @year1

AND

area = @area1

)

SELECT

newLightConditions,

SUM(numberOfCasualties) AS sumOfCasualties,

CAST( (CAST(SUM(numberOfCasualties) AS FLOAT) / @sumTotalCasualties \* 100) AS DECIMAL(10,2)) AS percentages

FROM

roadAccidentsWithLightGroups

GROUP BY

newLightConditions

ORDER BY

sumOfCasualties DESC

END

END

END

--combinations you can do

EXEC sumOfCasualtiesByLightConditionsAndPercentages

EXEC sumOfCasualtiesByLightConditionsAndPercentages @area1 = 'Rural'

EXEC sumOfCasualtiesByLightConditionsAndPercentages @year1 = 2021

EXEC sumOfCasualtiesByLightConditionsAndPercentages @year1 = 2021, @area1 = 'Rural'



