**TERM PROJECT DOCUMENTATION**

**Section :** Thursday

**Team Number :** 12

**Team Members :** Venkat Sai Phanindra Anagam, Vikranth Reddy Chapaala, Pavan Nalluri, Ritesh Venkata Sai Vesalapu

**TASK-1**

***Create Table Stored Procedure***

USE [ACCIDENTS]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[Create\_Tables] Script Date: 5/9/2024 10:13:35 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER PROCEDURE [dbo].[Create\_Tables] AS

BEGIN

CREATE TABLE Police\_Attendance

(

Did\_Police\_Officer\_Attend\_Scene\_Of\_Accident INT NOT NULL PRIMARY KEY,

Meaning nvarchar(100) NOT NULL

);

CREATE TABLE Dates

(

Date DATE NOT NULL PRIMARY KEY,

Day\_Of\_Week nvarchar(100) NOT NULL,

Year int

);

CREATE TABLE Location

(

Latitude FLOAT NOT NULL,

Longitude FLOAT NOT NULL,

PRIMARY KEY (Latitude, Longitude),

Location\_Easting\_OSGR nvarchar(100),

Location\_Northing\_OSGR nvarchar(100),

Police\_Force nvarchar(100),

LSOA\_Of\_Accident\_Location nvarchar(100),

Urban\_Or\_Rural\_Area nvarchar(100),

InScotland nvarchar(100)

);

CREATE TABLE Pedestrian\_Human\_Control

(

Pedestrian\_Crossing\_Human\_Control INT NOT NULL PRIMARY KEY,

Meaning nvarchar(100) NOT NULL

);

CREATE TABLE Pedestrian\_Physical\_Facilities

(

Pedestrian\_Crossing\_Physical\_Facilities INT NOT NULL PRIMARY KEY,

Meaning nvarchar(100) NOT NULL

);

CREATE TABLE Accidents

(

Accident\_Index nvarchar(100) NOT NULL PRIMARY KEY,

First\_Road\_Class nvarchar(100),

First\_Road\_Number INT,

Second\_Road\_Class nvarchar(100),

Second\_Road\_Number INT,

Accident\_Severity nvarchar(100),

Carriageway\_Hazards nvarchar(100),

Date DATE NOT NULL FOREIGN KEY REFERENCES Dates(Date) ON DELETE CASCADE,

Did\_Police\_Officer\_Attend\_Scene\_Of\_Accident INT FOREIGN KEY REFERENCES Police\_Attendance(Did\_Police\_Officer\_Attend\_Scene\_Of\_Accident) ON DELETE CASCADE,

Junction\_Control nvarchar(100),

Junction\_Detail nvarchar(100),

Latitude FLOAT ,

Longitude FLOAT ,

FOREIGN KEY (Latitude, Longitude) REFERENCES Location(Latitude, Longitude) ON DELETE CASCADE,

Light\_Conditions nvarchar(100),

Local\_District\_Authority nvarchar(100),

Local\_Highway\_Authority nvarchar(100),

Number\_Of\_Casualities INT,

Number\_Of\_Vehicles INT,

Pedestrian\_Crossing\_Human\_Control INT FOREIGN KEY REFERENCES Pedestrian\_Human\_Control(Pedestrian\_Crossing\_Human\_Control) ON DELETE CASCADE,

Pedestrian\_Crossing\_Physical\_Facilities INT FOREIGN KEY REFERENCES Pedestrian\_Physical\_Facilities(Pedestrian\_Crossing\_Physical\_Facilities) ON DELETE CASCADE,

Road\_Surface\_Conditions nvarchar(100),

Road\_Type nvarchar(100),

Special\_Conditions\_At\_Site nvarchar(100),

Speed\_Limit INT,

Time TIME,

Weather\_Conditions nvarchar(100)

) ;

CREATE TABLE Accident\_Location

(

Accident\_Index nvarchar(100) NOT NULL PRIMARY KEY,

Junction\_Location nvarchar(100),

Year INT

);

CREATE TABLE Restricted\_Lane

(

Vehicle\_Location\_Restricted\_Lane INT NOT NULL PRIMARY KEY,

Meaning nvarchar(100)

);

CREATE TABLE IMD\_Decile

(

Driver\_IMD\_Decile INT NOT NULL PRIMARY KEY,

Meaning nvarchar(100)

);

CREATE TABLE Vehicle

(

Accident\_Index nvarchar(100) NOT NULL,

Vehicle\_Reference INT NOT NULL,

PRIMARY KEY (Accident\_Index, Vehicle\_Reference),

Age\_Band\_of\_Driver nvarchar(100),

Age\_of\_Vehicle INT,

Driver\_Home\_Area\_Type nvarchar(100),

Driver\_IMD\_Decile INT FOREIGN KEY REFERENCES IMD\_Decile(Driver\_IMD\_Decile) ON DELETE CASCADE,

Engine\_Capacity\_CC INT,

Hit\_Object\_in\_Carriageway nvarchar(100),

Hit\_Object\_off\_Carriageway nvarchar(100),

Journey\_Purpose\_of\_Driver nvarchar(100),

make nvarchar(100),

model nvarchar(100),

Propulsion\_Code nvarchar(100),

Sex\_of\_Driver nvarchar(100),

Skidding\_and\_Overturning nvarchar(100),

Towing\_and\_Articulation nvarchar(100),

Vehicle\_Leaving\_Carriageway nvarchar(100),

Vehicle\_Location\_Restricted\_Lane INT FOREIGN KEY REFERENCES Restricted\_Lane(Vehicle\_Location\_Restricted\_Lane) ON DELETE CASCADE,

Vehicle\_Manoeuvre nvarchar(100),

Vehicle\_Type nvarchar(100),

Was\_Vehicle\_Left\_Hand\_Drive nvarchar(100),

X1st\_Point\_of\_Impact nvarchar(100)

);

ALTER TABLE [dbo].[Vehicle] ADD CONSTRAINT fk\_AcciI\_Vehivcletype FOREIGN KEY (Accident\_Index) REFERENCES [dbo].[Accidents](Accident\_Index);

ALTER TABLE [dbo].[Vehicle] ADD CONSTRAINT fk\_AcciI\_location FOREIGN KEY (Accident\_Index) REFERENCES [dbo].[Accident\_Location](Accident\_Index);

EXEC sp\_rename '[dbo].[Accidents].Number\_Of\_Casualities', 'Number\_of\_Casualities', 'COLUMN';

EXEC sp\_rename '[dbo].[Accidents].[Number\_Of\_Vehicles]', 'Number\_of\_Vehicles', 'COLUMN';

EXEC sp\_rename 'Accidents.[[Did\_Police\_Officer\_Attend\_Scene\_of\_Accident]]]', 'Did\_Police\_Officer\_Attend\_Scene\_of\_Accident', 'COLUMN';

END;

***Drop Table Stored Procedure***

USE [ACCIDENTS]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[Drop\_Tables] Script Date: 5/9/2024 10:14:08 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER PROCEDURE [dbo].[Drop\_Tables] AS

BEGIN

DROP TABLE If Exists [dbo].[Accidents];

DROP TABLE If Exists [dbo].[Vehicle];

DROP TABLE If Exists [dbo].[Accident\_Location];

DROP TABLE If Exists [dbo].[Dates];

DROP TABLE If Exists [dbo].[IMD\_Decile];

DROP TABLE If Exists [dbo].[Location];

DROP TABLE If Exists [dbo].[Pedestrian\_Human\_Control];

DROP TABLE If Exists [dbo].[Pedestrian\_Physical\_Facilities];

DROP TABLE If Exists [dbo].[Police\_Attendance];

DROP TABLE If Exists [dbo].[Restricted\_Lane];

END;

***Database Diagram***



***SQL Queries (In most queries top 10 rows of SQL Queries are displayed)***

1. Creating a view **(VIEW 1)** – We are creating a view ***temp1*** which contains columns from accident table along with car maker, age of vehicle and driver gender.

create or alter view temp1 as

select a.\*, v.make, v.Age\_of\_Vehicle, v.Sex\_of\_Driver

from [dbo].[Vehicle] v

join [dbo].[Accidents] a

on v.Accident\_Index = a.Accident\_Index;

1. **VIEW 2 –** Our view ***temp2***  contains all the accident table columns along with interpretation of whether police arrived at the incident or not.

create or alter view temp2 as

select a.\*,p.Meaning

from [dbo].[Accidents] a

join [dbo].[Police\_Attendance] p

on a.Did\_Police\_Officer\_Attend\_Scene\_Of\_Accident = p.Did\_Police\_Officer\_Attend\_Scene\_Of\_Accident

create or alter view temp2 as

select a.\*,p.Meaning

from [dbo].[Accidents] a

join [dbo].[Police\_Attendance] p

on a.Did\_Police\_Officer\_Attend\_Scene\_Of\_Accident = p.Did\_Police\_Officer\_Attend\_Scene\_Of\_Accident

1. This query calculates the total number of unique accidents for each vehicle make, sorting the results in descending order by the number of accidents**.(Query on view 1 , Aggregation)**

select make,count(distinct Accident\_Index) as No\_of\_accidents

from temp1

group by make

order by No\_of\_accidents desc;

A black and white chart with white text

Description automatically generated

1. This query calculates the total number of accidents for each year and local highway authority, sorting the results by year (**Join, Aggregation**)

select year,Local\_Highway\_Authority,count(\*) as No\_of\_accidents

from [dbo].[Accidents] a

join [dbo].[Dates] d

on a.Date=d.Date

group by year,Local\_Highway\_Authority

order by year

A black screen with white text

Description automatically generated

1. This query calculates the total number of casualties for each type of pedestrian human control, grouping the results by the meaning of each control type (**Join, Aggregation**)

select hc.Meaning, COUNT(a.Number\_Of\_Casualities) as Number\_Of\_Casualities

from Accidents a

join Pedestrian\_Human\_Control hc

on a.Pedestrian\_Crossing\_Human\_Control = hc.Pedestrian\_Crossing\_Human\_Control

group by hc.Meaning

A black screen with white text

Description automatically generated

1. This query calculates the total number of accidents for each combination of local district authority and police response type, grouping the results by these categories. (**Query on view 2**)

select Meaning as Did\_Police\_Arrive,Local\_District\_Authority,count(\*) as No\_of\_accidents

from temp2

group by Meaning,Local\_District\_Authority

A screenshot of a computer

Description automatically generated

1. This query calculates the total number of unique accidents for each IMD decile category, sorting the results in descending order by the number of accidents. (**Join,Aggregation**)

select i.Meaning,count(distinct v.Accident\_Index) as No\_of\_accidents

from [dbo].[Vehicle] v

join [dbo].[IMD\_Decile] i

on i.Driver\_IMD\_Decile=v.Driver\_IMD\_Decile

join [dbo].[Accidents] a

on v.Accident\_Index=a.Accident\_Index

group by i.Meaning

order by No\_of\_accidents desc

A black and white screen with white text

Description automatically generated

**TASK 2**

**Stored Procedure**

1. **Create table**

USE [OLAP]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[Create\_Tables] Script Date: 5/9/2024 11:17:40 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER PROCEDURE [dbo].[Create\_Tables] AS

BEGIN

CREATE TABLE Driver

(

Driver\_id INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

Age\_Band\_of\_Driver nvarchar(100),

Age\_of\_Vehicle INT,

Driver\_Home\_Area\_Type nvarchar(100),

Driver\_IMD\_Decile INT,

Journey\_Purpose\_of\_Driver nvarchar(100),

Sex\_of\_Driver nvarchar(100)

);

CREATE TABLE Automobile\_details

(

Automobile\_id INT NOT NULL IDENTITY(101,1) PRIMARY KEY,

Accident\_Index nvarchar(100) NOT NULL ,

Make nvarchar(100),

Model nvarchar(100),

Engine\_Capacity\_CC INT,

Propulsion\_Code nvarchar(100),

Vehicle\_Type nvarchar(100),

Was\_Vehicle\_Left\_Hand\_Drive nvarchar(100)

);

CREATE TABLE Accident\_details

(

Accident\_details\_id INT NOT NULL IDENTITY(1001,1) PRIMARY KEY,

Hit\_Object\_in\_Carriageway nvarchar(100),

Hit\_Object\_off\_Carriageway nvarchar(100),

Junction\_Location nvarchar(100),

Skidding\_and\_Overturning nvarchar(100),

Towing\_and\_Articulation nvarchar(100),

Vehicle\_Leaving\_Carriageway nvarchar(100),

Vehicle\_Location\_Restricted\_Lane INT,

Vehicle\_Manoeuvre nvarchar(100),

X1st\_Point\_of\_Impact nvarchar(100)

);

CREATE TABLE FT\_Vehicles\_Involved

(

Accident\_Index nvarchar(100) NOT NULL ,

Vehicle\_Reference INT NOT NULL,

PRIMARY KEY (Accident\_Index, Vehicle\_Reference),

Year INT,

Driver\_id INT FOREIGN KEY REFERENCES Driver(Driver\_id) ON DELETE CASCADE,

Automobile\_id INT FOREIGN KEY REFERENCES Automobile\_details(Automobile\_id) ON DELETE CASCADE,

Accident\_details\_id INT FOREIGN KEY REFERENCES Accident\_details(Accident\_details\_id) ON DELETE CASCADE

);

CREATE TABLE Date

(

Date\_id INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

Date DATE,

Day\_of\_Week nvarchar(100),

Time TIME,

Year INT

) ;

CREATE TABLE Location

(

Location\_id INT NOT NULL IDENTITY(101,1) PRIMARY KEY,

Latitude FLOAT ,

Longitude FLOAT,

Location\_Easting\_OSGR nvarchar(100),

Location\_Northing\_OSGR nvarchar(100),

Police\_Force nvarchar(100),

LSOA\_Of\_Accident\_Location nvarchar(100),

Urban\_Or\_Rural\_Area nvarchar(100),

InScotland nvarchar(100),

Local\_District\_Authority nvarchar(100),

Local\_Highway\_Authority nvarchar(100),

Pedestrian\_Crossing\_Human\_Control INT,

Pedestrian\_Crossing\_Physical\_Facilities INT

);

CREATE TABLE Road

(

Road\_id INT NOT NULL IDENTITY(1001,1) PRIMARY KEY,

First\_Road\_Class nvarchar(100),

First\_Road\_Number INT,

Second\_Road\_Class nvarchar(100),

Second\_Road\_Number INT,

Light\_Conditions nvarchar(100),

Road\_Surface\_Conditions nvarchar(100),

Road\_Type nvarchar(100),

Special\_Conditions\_At\_Site nvarchar(100),

Speed\_Limit INT

);

CREATE TABLE Acc\_Incidents

(

Acc\_Incidents\_id INT NOT NULL IDENTITY(101,1) PRIMARY KEY,

Accident\_Severity nvarchar(100),

Carriageway\_Hazards nvarchar(100),

Did\_Police\_Officer\_Attend\_Scene\_of\_Accident INT,

Junction\_Control nvarchar(100),

Junction\_Detail nvarchar(100)

);

CREATE TABLE FT\_Accident

(

Accident\_Index nvarchar(100) not null PRIMARY KEY,

Number\_of\_Casualties INT,

Number\_of\_Vehicles INT,

Date\_id INT FOREIGN KEY REFERENCES Date(Date\_id) ON DELETE CASCADE,

Location\_id INT FOREIGN KEY REFERENCES Location(Location\_id) ON DELETE CASCADE,

Road\_id INT FOREIGN KEY REFERENCES Road(Road\_id) ON DELETE CASCADE,

Acc\_Incidents\_id INT FOREIGN KEY REFERENCES Acc\_Incidents(Acc\_Incidents\_id) ON DELETE CASCADE,

Automobile\_id INT FOREIGN KEY REFERENCES Automobile\_details(Automobile\_id) ON DELETE CASCADE

);

CREATE TABLE DimDate

(

Date\_Key bigint NOT NULL PRIMARY KEY,

Date date NOT NULL,

Year bigint NOT NULL,

Month bigint NOT NULL,

MonthName varchar(max) NOT NULL,

Week bigint NOT NULL,

WeekDay bigint NOT NULL,

WeekDayName varchar(max) NOT NULL,

DayNumber bigint NOT NULL

);

EXEC sp\_rename '[dbo].[Automobile\_details].Make', 'make', 'COLUMN';

EXEC sp\_rename '[dbo].[Automobile\_details].Model', 'model', 'COLUMN';

EXEC sp\_rename '[dbo].[FT\_Accident].Number\_of\_Casualties', 'Number\_of\_Casualities', 'COLUMN';

ALTER TABLE [dbo].[FT\_Accident]

DROP CONSTRAINT [PK\_\_FT\_Accid\_\_ADCF401BBC50D76E];

ALTER TABLE [dbo].[FT\_Accident]

ADD FT\_Accident\_ID INT IDENTITY(1,1);

ALTER TABLE [dbo].[FT\_Accident]

ADD CONSTRAINT PK\_Accidents\_fact PRIMARY KEY (FT\_Accident\_ID);

ALTER TABLE [dbo].[FT\_Accident]

ADD

Accident\_Date\_key INT,

Accident\_date DATE;

END;

1. **DROP TABLE**

USE [OLAP]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[Drop\_Tables] Script Date: 5/9/2024 11:18:45 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER PROCEDURE [dbo].[Drop\_Tables] AS

BEGIN

DROP TABLE If Exists [dbo].[FT\_Accident];

DROP TABLE If Exists [dbo].[FT\_Vehicles\_Involved];

DROP TABLE If Exists [dbo].[Accident\_details];

DROP TABLE If Exists [dbo].[Automobile\_details];

DROP TABLE If Exists [dbo].[Driver];

DROP TABLE If Exists [dbo].[Date];

DROP TABLE If Exists [dbo].[Location];

DROP TABLE If Exists [dbo].[Road];

DROP TABLE If Exists [dbo].[Acc\_Incidents];

DROP TABLE If Exists [dbo].[DimDate]

END;

**DATABASE DIAGRAM**

****

**SQL Queries**

***Query 1***

**English Counterpart:**

Total Number of Accident Casualties in Scotland broken down by Urban and Rural areas

**SQL Query:**

SELECT coalesce([Urban\_Or\_Rural\_Area], 'All Area') AS Area\_Type,

SUM(A.Number\_of\_Casualities) AS Total\_Casualties

FROM FT\_Accident A join [dbo].[Location] l on A.Location\_id = l.Location\_id

where InScotland like '%Yes%'

GROUP BY CUBE (Urban\_Or\_Rural\_Area)

order by Total\_Casualties desc

**Results:**

+-------------+--------------------+

| Area\_Type | Total\_Casualties |

+=============+====================+

| All Area | 26605 |

+-------------+--------------------+

| Urban | 13653 |

+-------------+--------------------+

| Rural | 12950 |

+-------------+--------------------+

| Unallocated | 2 |

+-------------+--------------------+

***Query 2:***

**English Counterpart:**

How many accidents involving left-hand drive vehicles are there,

grouped by driver age bands, using the fact table that tracks event occurrences

**SQL Query:**

SELECT

Acc\_Incidents.Accident\_Severity,

FT\_Accident.Number\_of\_Vehicles,

COUNT(\*) AS Total\_Accidents

FROM

FT\_Accident

JOIN

Acc\_Incidents ON FT\_Accident.Acc\_Incidents\_id = Acc\_Incidents.Acc\_Incidents\_id

WHERE

FT\_Accident.Number\_of\_Vehicles > 6 AND

Acc\_Incidents.Did\_Police\_Officer\_Attend\_Scene\_of\_Accident = 1

GROUP BY

ROLLUP(Acc\_Incidents.Accident\_Severity, FT\_Accident.Number\_of\_Vehicles)

ORDER BY

Acc\_Incidents.Accident\_Severity DESC, FT\_Accident.Number\_of\_Vehicles DESC;

**Results:**

+---------------------+----------------------+-------------------+

| Accident\_Severity | Number\_of\_Vehicles | Total\_Accidents |

+=====================+======================+===================+

| Slight | 18 | 9 |

+---------------------+----------------------+-------------------+

| Slight | 16 | 12 |

+---------------------+----------------------+-------------------+

| Slight | 14 | 16 |

+---------------------+----------------------+-------------------+

| Slight | 12 | 36 |

+---------------------+----------------------+-------------------+

| Slight | 10 | 61 |

+---------------------+----------------------+-------------------+

| Slight | 9 | 75 |

+---------------------+----------------------+-------------------+

| Slight | 8 | 205 |

+---------------------+----------------------+-------------------+

| Slight | 7 | 443 |

+---------------------+----------------------+-------------------+

| Slight | | 857 |

+---------------------+----------------------+-------------------+

| Serious | 14 | 14 |

+---------------------+----------------------+-------------------+

| Serious | 13 | 11 |

+---------------------+----------------------+-------------------+

| Serious | 11 | 13 |

+---------------------+----------------------+-------------------+

| Serious | 10 | 10 |

+---------------------+----------------------+-------------------+

| Serious | 9 | 33 |

+---------------------+----------------------+-------------------+

| Serious | 8 | 87 |

+---------------------+----------------------+-------------------+

| Serious | 7 | 98 |

+---------------------+----------------------+-------------------+

| Serious | | 266 |

+---------------------+----------------------+-------------------+

| Fatal | 10 | 13 |

+---------------------+----------------------+-------------------+

| Fatal | 9 | 4 |

+---------------------+----------------------+-------------------+

| Fatal | 8 | 21 |

+---------------------+----------------------+-------------------+

| Fatal | 7 | 13 |

+---------------------+----------------------+-------------------+

| Fatal | | 51 |

+---------------------+----------------------+-------------------+

| | | 1174 |

+---------------------+----------------------+-------------------+

***Query 3:***

**English Counterpart:**

Total Number of Accident Casualties in UK categorized by Visibility and Vehicle Type

**SQL Query:**

select coalesce(r.Light\_Conditions,'All Light Conditions') as [Light Conditions],

coalesce(ad.Vehicle\_Type,'All vehicle type') as [Vehicle Type],

sum(a.Number\_of\_Casualities) as Number\_of\_Casualities

from [dbo].[FT\_Accident] a join [dbo].[Road] r

on r.Road\_id = a.Road\_id join [dbo].[Automobile\_details] ad on ad.Automobile\_id = a.Automobile\_id

Group by rollup(ad.Vehicle\_Type,r.Light\_Conditions)

order by [Light Conditions] asc,Number\_of\_Casualities desc

**Results:**

+------------------------------+---------------------------------------+-------------------------+

| Light Conditions | Vehicle Type | Number\_of\_Casualities |

+==============================+=======================================+=========================+

| All Light Conditions | All vehicle type | 435143 |

+------------------------------+---------------------------------------+-------------------------+

| All Light Conditions | Car | 333352 |

+------------------------------+---------------------------------------+-------------------------+

| All Light Conditions | Van / Goods 3.5 tonnes mgw or under | 23712 |

+------------------------------+---------------------------------------+-------------------------+

| All Light Conditions | Bus or coach (17 or more pass seats) | 14661 |

+------------------------------+---------------------------------------+-------------------------+

| All Light Conditions | Motorcycle over 500cc | 12479 |

+------------------------------+---------------------------------------+-------------------------+

| All Light Conditions | Goods 7.5 tonnes mgw and over | 10387 |

+------------------------------+---------------------------------------+-------------------------+

| All Light Conditions | Motorcycle 125cc and under | 9285 |

+------------------------------+---------------------------------------+-------------------------+

| All Light Conditions | Taxi/Private hire car | 9132 |

+------------------------------+---------------------------------------+-------------------------+

| All Light Conditions | Pedal cycle | 5816 |

+------------------------------+---------------------------------------+-------------------------+

| All Light Conditions | Goods over 3.5t. and under 7.5t |

3620 |

***Query 4:***

**English Counterpart:**

Total Number of Accident Casualties in UK categorized by the Police Force Jurisdiction

**SQL Query:**

select coalesce(l.[Police\_Force],'All Police force') as [Police Force],sum(a.Number\_of\_Casualities) as Number\_of\_Casualities

from [dbo].[FT\_Accident] a join [dbo].[Location] l

on a.Location\_id = l.Location\_id

Group by rollup(l.[Police\_Force])

**Results:**

+-----------------------+-------------------------+

| Police Force | Number\_of\_Casualities |

+=======================+=========================+

| Avon and Somerset | 10276 |

+-----------------------+-------------------------+

| Bedfordshire | 4502 |

+-----------------------+-------------------------+

| Cambridgeshire | 6649 |

+-----------------------+-------------------------+

| Central | 1375 |

+-----------------------+-------------------------+

| Cheshire | 9720 |

+-----------------------+-------------------------+

| City of London | 654 |

+-----------------------+-------------------------+

| Cleveland | 2940 |

+-----------------------+-------------------------+

| Cumbria | 4053 |

+-----------------------+-------------------------+

| Derbyshire | 7433 |

+-----------------------+-------------------------+

| Devon and Cornwall | 12358 |

+-----------------------+-------------------------+

| Dorset | 5694 |

+-----------------------+-------------------------+

| Dumfries and Galloway | 958 |

+-----------------------+-------------------------+

| Durham | 4336 |

+-----------------------+-------------------------+

| Dyfed-Powys | 4266 |

+-----------------------+-------------------------+

| Essex | 11776 |

+-----------------------+-------------------------+

| Fife | 1364 |

+-----------------------+-------------------------+

| Gloucestershire | 2960 |

+-----------------------+-------------------------+

| Grampian | 2369 |

+-----------------------+-------------------------+

| Greater Manchester | 15163 |

+-----------------------+-------------------------+

| Gwent | 2433 |

+-----------------------+-------------------------+

| Hampshire | 12433 |

+-----------------------+-------------------------+

| Hertfordshire | 8892 |

+-----------------------+-------------------------+

| Humberside | 7434 |

+-----------------------+-------------------------+

| Kent | 14911 |

+-----------------------+-------------------------+

| Lancashire | 12976 |

+-----------------------+-------------------------+

| Leicestershire | 7215 |

+-----------------------+-------------------------+

| Lincolnshire | 6347 |

+-----------------------+-------------------------+

| Lothian and Borders | 5593 |

+-----------------------+-------------------------+

| Merseyside | 9836 |

+-----------------------+-------------------------+

| Metropolitan Police | 56758 |

+-----------------------+-------------------------+

| Norfolk | 5822 |

+-----------------------+-------------------------+

| North Wales | 2387 |

+-----------------------+-------------------------+

| North Yorkshire | 6732 |

+-----------------------+-------------------------+

| Northamptonshire | 4033 |

+-----------------------+-------------------------+

| Northern | 1711 |

+-----------------------+-------------------------+

| Northumbria | 10290 |

+-----------------------+-------------------------+

| Nottinghamshire | 8538 |

+-----------------------+-------------------------+

| South Wales | 8100 |

+-----------------------+-------------------------+

| South Yorkshire | 10769 |

+-----------------------+-------------------------+

| Staffordshire | 9304 |

+-----------------------+-------------------------+

| Strathclyde | 11429 |

+-----------------------+-------------------------+

| Suffolk | 5209 |

+-----------------------+-------------------------+

| Surrey | 13128 |

+-----------------------+-------------------------+

| Sussex | 12397 |

+-----------------------+-------------------------+

| Tayside | 1807 |

+-----------------------+-------------------------+

| Thames Valley | 16539 |

+-----------------------+-------------------------+

| Warwickshire | 4773 |

+-----------------------+-------------------------+

| West Mercia | 7590 |

+-----------------------+-------------------------+

| West Midlands | 18144 |

+-----------------------+-------------------------+

| West Yorkshire | 17974 |

+-----------------------+-------------------------+

| Wiltshire | 4776 |

+-----------------------+-------------------------+

| All Police force | 435126 |

+-----------------------+-------------------------+

***Query 5:***

**English Counterpart:**

Progression of Total Casualties and corresponding driver demographics over the years in UK

**SQL Query:**

SELECT

coalesce(D.Sex\_of\_Driver, 'All Genders') AS Gender,

VI.Year AS Year,

SUM(A.Number\_of\_Casualities) AS Total\_Casualties

FROM [dbo].[Driver] D JOIN [dbo].[FT\_Vehicles\_Involved] VI ON D.Driver\_id = VI.Driver\_id JOIN FT\_Accident A on A.Automobile\_id = VI.Automobile\_id

where D.Sex\_of\_Driver IN ('Male','Female','Not known')

GROUP BY CUBE (D.Sex\_of\_Driver, VI.Year)

**Results:**

+-------------+--------+--------------------+

| Gender | Year | Total\_Casualties |

+=============+========+====================+

| Female | 2005 | 6518 |

+-------------+--------+--------------------+

| Male | 2005 | 17224 |

+-------------+--------+--------------------+

| Not known | 2005 | 532 |

+-------------+--------+--------------------+

| All Genders | 2005 | 24274 |

+-------------+--------+--------------------+

| Female | 2006 | 6531 |

+-------------+--------+--------------------+

| Male | 2006 | 17175 |

+-------------+--------+--------------------+

| Not known | 2006 | 449 |

+-------------+--------+--------------------+

| All Genders | 2006 | 24155 |

+-------------+--------+--------------------+

| Female | 2007 | 7326 |

+-------------+--------+--------------------+

| Male | 2007 | 19528 |

+-------------+--------+--------------------+

| Not known | 2007 | 548 |

+-------------+--------+--------------------+

| All Genders | 2007 | 27402 |

+-------------+--------+--------------------+

| Female | 2008 | 7524 |

+-------------+--------+--------------------+

| Male | 2008 | 18406 |

+-------------+--------+--------------------+

| Not known | 2008 | 555 |

+-------------+--------+--------------------+

| All Genders | 2008 | 26485 |

+-------------+--------+--------------------+

| Female | 2009 | 11986 |

+-------------+--------+--------------------+

| Male | 2009 | 26241 |

+-------------+--------+--------------------+

| Not known | 2009 | 773 |

+-------------+--------+--------------------+

| All Genders | 2009 | 39000 |

+-------------+--------+--------------------+

| Female | 2010 | 11926 |

+-------------+--------+--------------------+

| Male | 2010 | 26282 |

+-------------+--------+--------------------+

| Not known | 2010 | 755 |

+-------------+--------+--------------------+

| All Genders | 2010 | 38963 |

+-------------+--------+--------------------+

| Female | 2011 | 12320 |

+-------------+--------+--------------------+

| Male | 2011 | 25684 |

+-------------+--------+--------------------+

| Not known | 2011 | 815 |

+-------------+--------+--------------------+

| All Genders | 2011 | 38819 |

+-------------+--------+--------------------+

| Female | 2012 | 12395 |

+-------------+--------+--------------------+

| Male | 2012 | 24600 |

+-------------+--------+--------------------+

| Not known | 2012 | 797 |

+-------------+--------+--------------------+

| All Genders | 2012 | 37792 |

+-------------+--------+--------------------+

| Female | 2013 | 11423 |

+-------------+--------+--------------------+

| Male | 2013 | 24238 |

+-------------+--------+--------------------+

| Not known | 2013 | 837 |

+-------------+--------+--------------------+

| All Genders | 2013 | 36498 |

+-------------+--------+--------------------+

| Female | 2014 | 12423 |

+-------------+--------+--------------------+

| Male | 2014 | 26415 |

+-------------+--------+--------------------+

| Not known | 2014 | 978 |

+-------------+--------+--------------------+

| All Genders | 2014 | 39816 |

+-------------+--------+--------------------+

| Female | 2015 | 15413 |

+-------------+--------+--------------------+

| Male | 2015 | 34472 |

+-------------+--------+--------------------+

| Not known | 2015 | 2807 |

+-------------+--------+--------------------+

| All Genders | 2015 | 52692 |

+-------------+--------+--------------------+

| Female | 2016 | 14782 |

+-------------+--------+--------------------+

| Male | 2016 | 33264 |

+-------------+--------+--------------------+

| Not known | 2016 | 3191 |

+-------------+--------+--------------------+

| All Genders | 2016 | 51237 |

+-------------+--------+--------------------+

| All Genders | | 437133 |

+-------------+--------+--------------------+

| Female | | 130567 |

+-------------+--------+--------------------+

| Male | | 293529 |

+-------------+--------+--------------------+

| Not known | | 13037 |

+-------------+--------+--------------------+

***Query 6:***

**English Counterpart:**

Total Number of Accidents Involving Male Drivers, Categorized by Vehicle Age.

**SQL Query:**

SELECT

COALESCE(CAST(d.Age\_of\_Vehicle AS VARCHAR), 'All Vehicle Ages') AS [Vehicle Age],

COUNT(DISTINCT Accident\_Index) AS [Number of Accidents by Age]

FROM [dbo].[FT\_Vehicles\_Involved] vi

JOIN [dbo].[Driver] d

ON vi.Driver\_id = d.Driver\_id

WHERE d.Sex\_of\_Driver = 'Male' and Age\_of\_Vehicle is not null

GROUP BY rollup(Age\_of\_Vehicle)

Results:

+------------------+------------------------------+

| Vehicle Age | Number of Accidents by Age |

+==================+==============================+

| 1 | 16364 |

+------------------+------------------------------+

| 2 | 14318 |

+------------------+------------------------------+

| 3 | 13385 |

+------------------+------------------------------+

| 4 | 12794 |

+------------------+------------------------------+

| 5 | 12375 |

+------------------+------------------------------+

| 6 | 11756 |

+------------------+------------------------------+

| 7 | 11449 |

+------------------+------------------------------+

| 8 | 11329 |

+------------------+------------------------------+

| 9 | 10723 |

+------------------+------------------------------+

| 10 | 10014 |

+------------------+------------------------------+

| 11 | 8896 |

+------------------+------------------------------+

| 12 | 7893 |

+------------------+------------------------------+

| 13 | 6329 |

+------------------+------------------------------+

| 14 | 4845 |

+------------------+------------------------------+

| 15 | 3386 |

+------------------+------------------------------+

| 16 | 2406 |

+------------------+------------------------------+

| 17 | 1594 |

+------------------+------------------------------+

| 18 | 976 |

+------------------+------------------------------+

| 19 | 705 |

+------------------+------------------------------+

| 20 | 546 |

+------------------+------------------------------+

| 21 | 382 |

+------------------+------------------------------+

| 22 | 274 |

+------------------+------------------------------+

| 23 | 215 |

+------------------+------------------------------+

| 24 | 166 |

+------------------+------------------------------+

| 25 | 123 |

+------------------+------------------------------+

| 26 | 116 |

+------------------+------------------------------+

| 27 | 81 |

+------------------+------------------------------+

| 28 | 76 |

+------------------+------------------------------+

| 29 | 62 |

+------------------+------------------------------+

| 30 | 39 |

+------------------+------------------------------+

| 31 | 35 |

+------------------+------------------------------+

| 32 | 27 |

+------------------+------------------------------+

| 33 | 25 |

+------------------+------------------------------+

| 34 | 20 |

+------------------+------------------------------+

| 35 | 12 |

+------------------+------------------------------+

| 36 | 5 |

+------------------+------------------------------+

| 37 | 13 |

+------------------+------------------------------+

| 38 | 10 |

+------------------+------------------------------+

| 39 | 7 |

+------------------+------------------------------+

| 40 | 11 |

+------------------+------------------------------+

| 41 | 5 |

+------------------+------------------------------+

| 42 | 7 |

+------------------+------------------------------+

| 43 | 6 |

+------------------+------------------------------+

| 44 | 4 |

+------------------+------------------------------+

| 45 | 7 |

+------------------+------------------------------+

| 46 | 6 |

+------------------+------------------------------+

| 47 | 11 |

+------------------+------------------------------+

| 48 | 4 |

+------------------+------------------------------+

| 49 | 13 |

+------------------+------------------------------+

| 50 | 7 |

+------------------+------------------------------+

| 51 | 2 |

+------------------+------------------------------+

| 52 | 1 |

+------------------+------------------------------+

| 53 | 4 |

+------------------+------------------------------+

| 54 | 3 |

+------------------+------------------------------+

| 55 | 6 |

+------------------+------------------------------+

| 56 | 2 |

+------------------+------------------------------+

| 57 | 1 |

+------------------+------------------------------+

| 58 | 3 |

+------------------+------------------------------+

| 63 | 1 |

+------------------+------------------------------+

| 64 | 1 |

+------------------+------------------------------+

| 65 | 1 |

+------------------+------------------------------+

| 72 | 2 |

+------------------+------------------------------+

| 75 | 1 |

+------------------+------------------------------+

| 76 | 2 |

+------------------+------------------------------+

| 79 | 1 |

+------------------+------------------------------+

| 83 | 1 |

+------------------+------------------------------+

| 85 | 1 |

+------------------+------------------------------+

| All Vehicle Ages | 135651 |

+------------------+------------------------------+

***Query 7:***

**English Counterpart:**

How many accidents involving left-hand drive vehicles are there,

grouped by driver age bands, using the fact table that tracks event occurrences

**SQL Query:**

SELECT

D.Age\_Band\_of\_Driver,

COUNT(\*) AS Total\_Incidents

FROM

FT\_Vehicles\_Involved FV

JOIN

Automobile\_details AD ON FV.Automobile\_id = AD.Automobile\_id

JOIN

Driver D ON FV.Driver\_id = D.Driver\_id

WHERE

AD.Was\_Vehicle\_Left\_Hand\_Drive = 'Yes'

GROUP BY

D.Age\_Band\_of\_Driver

ORDER BY

D.Age\_Band\_of\_Driver;

**Results:**

+------------------------------+-------------------+

| Age\_Band\_of\_Driver | Total\_Incidents |

+==============================+===================+

| 16 - 20 | 27 |

+------------------------------+-------------------+

| 21 - 25 | 56 |

+------------------------------+-------------------+

| 26 - 35 | 119 |

+------------------------------+-------------------+

| 36 - 45 | 114 |

+------------------------------+-------------------+

| 46 - 55 | 110 |

+------------------------------+-------------------+

| 56 - 65 | 55 |

+------------------------------+-------------------+

| 66 - 75 | 25 |

+------------------------------+-------------------+

| Data missing or out of range | 75 |

+------------------------------+-------------------+

| Over 75 | 11 |

+------------------------------+-------------------+

**Task 3:**

*Query 1:*

**Nosql Query:**

db.Vehicles.find({'Age\_of\_Vehicle': {'$gt': 5}}, {'Accident\_Index': 1, 'Age\_of\_Vehicle': 1, 'model ': 1, 'Propulsion\_Code': 1, '\_id': 0}).sort({'Age\_of\_Vehicle': 1}).limit(5)

**English Counterpart:**

This query finds the first five vehicles details in the database that are older than 5 years and lists their accident index, age, model, and propulsion code, sorting the results by age in ascending order.

**Query results:**

+------------------+------------------+--------------------+-------------------+

| Accident\_Index | Age\_of\_Vehicle | model | Propulsion\_Code |

+==================+==================+====================+===================+

| 200501BS70049 | 6 | TX1 SILVER AUTO | Heavy oil |

+------------------+------------------+--------------------+-------------------+

| 200501BS00012 | 6 | MONDEO GHIA X AUTO | Petrol |

+------------------+------------------+--------------------+-------------------+

| 200501BS70249 | 6 | CITY FLY CLR 125 | Petrol |

+------------------+------------------+--------------------+-------------------+

| 200501BS70140 | 6 | VESPA ET4 | Petrol |

+------------------+------------------+--------------------+-------------------+

| 200501BS70007 | 6 | HR-V | Petrol |

+------------------+------------------+--------------------+-------------------+

*Query 2:*

**Nosql Aggregation Query:**

db.Vehicles.aggregate([{'$group': {'\_id': '$Sex\_of\_Driver', 'Gender': {'$first': '$Sex\_of\_Driver'}, 'totalAccidents': {'$sum': 1}}}, {'$project': {'\_id': 0, 'Gender': 1, 'totalAccidents': 1}}, {'$sort': {'totalAccidents': -1}}])

**English Counterpart:**

This query counts the total number of accidents for each driver gender, lists these totals along with the corresponding gender, and sorts the results by the number of accidents in descending order.

**Query results:**

+------------------------------+------------------+

| Gender | totalAccidents |

+==============================+==================+

| Male | 202811 |

+------------------------------+------------------+

| Female | 88728 |

+------------------------------+------------------+

| Not known | 10521 |

+------------------------------+------------------+

| Data missing or out of range | 19 |

+------------------------------+------------------+

*Query 3:*

**Nosql Query:**

db.Accidents.find({'Local\_Authority\_(District)': {'$regex': '^B'}}, {'Accident\_Index': 1, 'Local\_Authority\_(District)': 1, 'Accident\_Severity': 1, 'Speed\_limit': 1, '\_id': 0}).sort({'Speed\_limit': -1}).limit(5)

**English Counterpart:**

This query finds the first five accidents where the local authority district starts with the letter 'B' and lists their accident index, local authority district, accident severity, and speed limit, sorting the results by speed limit in descending order.

**Query results:**

+------------------+---------------------+------------------------------+---------------+

| Accident\_Index | Accident\_Severity | Local\_Authority\_(District) | Speed\_limit |

+==================+=====================+==============================+===============+

| 200504F018105 | Slight | Burnley | 70 |

+------------------+---------------------+------------------------------+---------------+

| 200501TC00454 | Slight | Barking and Dagenham | 70 |

+------------------+---------------------+------------------------------+---------------+

| 200504TA05022 | Slight | Blackburn with Darwen | 70 |

+------------------+---------------------+------------------------------+---------------+

| 200504F023205 | Slight | Burnley | 70 |

+------------------+---------------------+------------------------------+---------------+

| 200501SX20975 | Slight | Barnet | 70 |

+------------------+---------------------+------------------------------+---------------+

*Query 4:*

**Nosql Query:**

db.Accidents.find({'$and': [{'Number\_of\_Casualties': {'$gte': 30}}, {'Number\_of\_Casualties': {'$lte': 100}}]}, {'\_id': 0, 'Accident\_Index': 1, 'Accident\_Severity': 1, 'Date': 1, 'Road\_Surface\_Conditions': 1, 'Weather\_Conditions': 1})

**English Counterpart:**

This query finds all accidents where the number of casualties is between 30 and 100 and it lists accident's index, severity, date, road surface conditions, and weather conditions for each accident.

**Query results:**

+------------------+---------------------+------------+---------------------------+-----------------------+

| Accident\_Index | Accident\_Severity | Date | Road\_Surface\_Conditions | Weather\_Conditions |

+==================+=====================+============+===========================+=======================+

| 20074100T1174 | Fatal | 2007-10-11 | Wet or damp | Fog or mist |

+------------------+---------------------+------------+---------------------------+-----------------------+

| 200743N002017 | Fatal | 2007-01-03 | Wet or damp | Raining no high winds |

+------------------+---------------------+------------+---------------------------+-----------------------+

| 200805FF25077 | Slight | 2008-05-24 | Dry | Fine no high winds |

+------------------+---------------------+------------+---------------------------+-----------------------+

| 2010030001126 | Fatal | 2010-05-24 | Dry | Fine no high winds |

+------------------+---------------------+------------+---------------------------+-----------------------+

| 201097QC71203 | Fatal | 2010-03-31 | Snow | Snowing + high winds |

+------------------+---------------------+------------+---------------------------+-----------------------+

| 201291NM05840 | Fatal | 2012-06-08 | Dry | Fine no high winds |

+------------------+---------------------+------------+---------------------------+-----------------------+

| 2013353140013 | Serious | 2013-10-13 | Wet or damp | Raining + high winds |

+------------------+---------------------+------------+---------------------------+-----------------------+

| 201597LB05103 | Serious | 2015-03-26 | Dry | Fine + high winds |

+------------------+---------------------+------------+---------------------------+-----------------------+

*Query 5:*

**Nosql Aggregation Query:**

db.Accidents.aggregate([{'$match': {'InScotland': 'Yes'}}, {'$group': {'\_id': '$Urban\_or\_Rural\_Area', 'Urban\_or\_Rural\_Area': {'$first': '$Urban\_or\_Rural\_Area'}, 'Number\_of\_Accidents': {'$sum': 1}}}, {'$project': {'\_id': 0, 'Urban\_or\_Rural\_Area': 1, 'Number\_of\_Accidents': 1}}, {'$sort': {'Number\_of\_Accidents': -1}}])

**English Counterpart:**

This aggregation filters accidents that occurred in Scotland and groups them by 'Urban or Rural Area'.It calculates the total number of accidents for each area type and sorts the results by the number of accidents in descending order.

**Query results:**

+-----------------------+-----------------------+

| Urban\_or\_Rural\_Area | Number\_of\_Accidents |

+=======================+=======================+

| Urban | 7446 |

+-----------------------+-----------------------+

| Rural | 6018 |

+-----------------------+-----------------------+

| Unallocated | 1 |

+-----------------------+-----------------------+