TUS

THE TECHNOLOGICAL UNIVERSITY OF THE SHANNON: MIDLANDS MIDWEST

Student Name:	**Vinit Sharma**
Student Id Number:	**A00325758**
Course:	**Masters in Data Analytics**
Title of Assignment:	**Relation Database Project**
Date:	**11 December 2024**

A00325758
Vinit Sharma

Declaration

I hereby certify that the material, which is submitted in this report towards the award of MSc. in Data Analytics, is entirely my own work and has not been submitted for any academic assessment other than part fulfilment of the above named award.

Signed_____ Date 11-12-2024

Contents

1. Introduction:

When Prison building projects are put out to tender, a preferred bidder is chosen and a contract cost is written into the contract. Yet, in most cases, there are unforeseen challenges/problems (that couldn't be identified before the project starts – i.e. until the foundations for the Prison are dug) and the cost of addressing such problems has to be agreed as the project proceeds - the actual cost is typically higher than the contract cost. A database is required to store details of the Builders who build Prisons and the Prisons that they build. A Builder may build many Prisons and a Prison is built by a single Builder (building company).
This report aims to finalize the database design for managing prison building contracts. It addresses the relationship between builders and prison projects and provides solutions to a few database-related challenges using SQL analytic functions. These design and insights can be helpful to enhance project efficiency

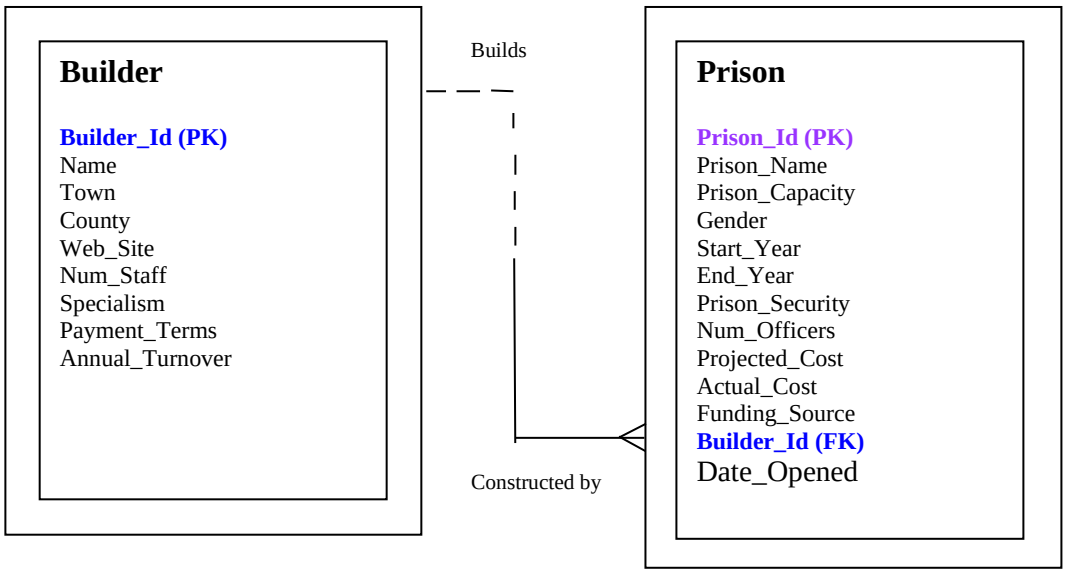
2. Entities and attributes:

The database comprises two key entities: Builder and Prison, structured in a parent-child relationship. The Builder table serves as the parent, with Builder_Id as its primary key, uniquely identifying each builder. The Prison table acts as the child, containing Builder_Id as a foreign key to establish a relationship with the Builder table. This linkage ensures that each prison is associated with a specific builder, while a single builder can be linked to multiple prisons. The relationship enforces referential integrity, streamlining data management and providing a clear structure for tracking prison construction projects.

Builder (**Builder_Id (PK)**, Name, Town, County, Web_Site, Num_Staff, Specialism, Payment_Terms, Annual_Turnover)
Prison (**Prison_Id (PK)**, Prison_Name, Prison_Capacity ,Gender, Start_Year, End_Year, Prison_Security, Num_Officers, Projected_Cost, Actual_Cost, Funding_Source, **Builder_Id (FK),** Date_Opened)

3. Entity Relationship Diagram (ERD)

With Attributes



Without Attributes

4. Create Queries

```
Create Table Builder      (
        Builder_Id          Number(2),
        Name                Varchar2(37),
        Town                Varchar2(13),
        County              Varchar2(9),
        Web_Site            Varchar2(28),
        Num_Staff           Number(4),
        Specialism          Varchar2(19),
        Payment_Terms       Varchar2(11),
        Annual_Turnover     Number(6),
Constraint Builder_Builder_Id_Pk Primary Key (Builder_Id),
Constraint Builder_Specialism_Ck Check (Specialism In ('Prisons', 'Office Blocks', 'Specialist
Premises', 'Underground Prisons')),
Constraint Builder_Payment_Terms_Ck Check (Payment_Terms In ('Weekly', 'Fortnightly', 'Monthly',
'Quarterly')));
```

```
Create Table Prison      (
        Prison_Id           Number(2),
        Prison_Name         Varchar2(24),
        Prison_Capacity     Number(3),
        Gender              Varchar2(6),
        Start_Year          Number(4),
        End_Year            Number(4),
        Prison_Security     Varchar2(6),
        Num_Officers        Number(3),
        Projected_Cost      Number(6),
        Actual_Cost         Number(6),
        Funding_Source      Varchar2(25),
        Builder_Id          Number(2),
        Date_Opened         Date,
Constraint Prison_Prison_Id_Pk Primary Key (Prison_Id),
Constraint Prison_Builder_Id_Fk Foreign Key (Builder_Id) References Builder (Builder_Id),
Constraint Prison_Gender_Ck Check (Gender In ('Female', 'Male', 'Mixed')),
Constraint Prison_Prison_Security_Ck Check (Prison_Security In ('Low', 'Medium', 'High')));
```

5. Insert Queries

BUILDER INSERTS

Insert Into Builder Values ('1', 'HBG Ascon', 'Crookstown', 'Cork', 'www.asconrohcon.com', '1100', 'Prisons', 'Weekly', '280');
Insert Into Builder Values ('2', 'Michael McNamara And Co', 'Carrigtwohill', 'Cork', 'www.mcnamaraconstruction.com', '500', 'Prisons', 'Fortnightly', '270.31');
Insert Into Builder Values ('3', 'PJ Hegarty And Sons', 'Blarney', 'Cork', 'www.pjhegarty.ie', '670', 'Specialist Premises', 'Monthly', '251.4');
Insert Into Builder Values ('4', 'Bowen Group', 'Buttevant', 'Cork', 'www.bowengroup.ie', '870', 'Underground Prisons', 'Monthly', '193');
Insert Into Builder Values ('5', 'G And T Crampton Group', 'Conna', 'Cork', 'www.gtcrampton.ie', '250', 'Office Blocks', 'Monthly', '160.51');
Insert Into Builder Values ('6', 'Menolly Homes', 'Castlefin', 'Donegal', 'www.menolly.ie', '15', 'Prisons', 'Weekly', '157.4');
Insert Into Builder Values ('7', 'John Fleming Construction', 'Ballyshannon', 'Donegal', 'www.jfConstruction.ie', '168', 'Underground Prisons', 'Monthly', '135');
Insert Into Builder Values ('8', 'Ballymore Properties', 'Greencastle', 'Donegal', 'www.ballymoreproperties.ie', '140', 'Prisons', 'Quarterly', '134');
Insert Into Builder Values ('9', 'P Elliott And Co', 'Brinlack', 'Donegal', 'www.pelliott.com', '250', 'Prisons', 'Weekly', '130');
Insert Into Builder Values ('10', 'Park Developments (Dublin)', 'Dunfanaghy', 'Donegal', 'www.parkDevelopments.com', '150', 'Prisons', 'Fortnightly', '111');
Insert Into Builder Values ('11', 'O"Flynn Construction', 'Dublin City', 'Dublin', 'www.oFlynnConstruction.ie', '110', 'Specialist Premises', 'Monthly', '74.6');
Insert Into Builder Values ('12', 'Manor Park Homebuilders', 'Brittas', 'Dublin', 'www.manorpark.ie', '120', 'Underground Prisons', 'Quarterly', '72.7');
Insert Into Builder Values ('13', 'Uniform Construction', 'Dublin', 'Dublin', 'www.uniformConstruction.ie', '250', 'Underground Prisons', 'Weekly', '68.3');
Insert Into Builder Values ('14', 'Murnane And O"Shea', 'Clifden', 'Galway', 'www.mando.ie', '200', 'Prisons', 'Weekly', '58');
Insert Into Builder Values ('15', 'Duggan Brothers (Contractors)', 'Ahascragh', 'Galway', 'www.duggans.ie', '150', 'Prisons', 'Monthly', '57.53');
Insert Into Builder Values ('16', 'TBD Building Contractors', 'Ballygar', 'Galway', 'www.tbd.ie', '45', 'Office Blocks', 'Quarterly', '54.67');
Insert Into Builder Values ('17', 'Coffey Construction', 'Glenamaddy', 'Galway', 'www.coffeyconstruction.com', '380', 'Specialist Premises', 'Weekly', '51.26');
Insert Into Builder Values ('18', 'Brian McCarthy Building Contractors', 'Galway City', 'Galway', 'www.bmcc.ie', '120', 'Specialist Premises', 'Monthly', '51');
Insert Into Builder Values ('19', 'John F Supple', 'Cloonboo', 'Galway', 'www.johnfsupple.ie', '100', 'Prisons', 'Monthly', '47.11');
Insert Into Builder Values ('20', 'Maplewood Development Group', 'Carraroe', 'Galway', 'www.maplewood.ie', '40', 'Office Blocks', 'Quarterly', '41.15');
Insert Into Builder Values ('21', 'Noonan Construction', 'Athenry', 'Galway', 'www.noonanconstruction.ie', '48', 'Specialist Premises', 'Weekly', '36.75');
Insert Into Builder Values ('22', 'CLG Builders', 'Gort', 'Galway', 'www.clg.ie', '300', 'Underground Prisons', 'Fortnightly', '35');
Insert Into Builder Values ('23', 'Collen Group', 'Headford', 'Galway', 'www.cpmarchitecture.com', '95', 'Prisons', 'Monthly', '33.82');
Insert Into Builder Values ('24', 'Dwyer Nolan Developments', 'Fenit', 'Kerry', 'www.dwyernolan.ie', '37', 'Prisons', 'Quarterly', '30');
Insert Into Builder Values ('25', 'McCabe Builders', 'Causeway', 'Kerry', 'www.mccabebuilders.com', '170', 'Prisons', 'Weekly', '30');
Insert Into Builder Values ('26', 'William Neville And Sons Construction', 'Castleisland', 'Kerry', 'www.nevilleconstruction.ie', '47', 'Office Blocks', 'Weekly', '29.82');
Insert Into Builder Values ('27', 'SMC Group', 'Ballyduff', 'Kerry', 'www.smcgroup.ie', '250', 'Underground Prisons', 'Monthly', '26.73');
Insert Into Builder Values ('28', 'Allen And Smyth Construction', 'Annascaul', 'Kerry', 'www.allenandsmyth.com', '75', 'Underground Prisons', 'Monthly', '24');
Insert Into Builder Values ('29', 'Cedar Building Co', 'Athlone', 'Westmeath', 'www.cedarbuildingco.ie', '70', 'Prisons', 'Weekly', '23.5');
Insert Into Builder Values ('30', 'Kelland Homes', 'Mullingar', 'Westmeath', 'www.kellandhomes.ie', '40', 'Underground Prisons', 'Weekly', '22');

PRISON INSERTS

Insert Into Prison Values (1,'Banagher Prison',356,'Mixed',1998,2001,'High',294,124000,129000,'KEB Ireland',12,'27-07-2001');
Insert Into Prison Values (2,'Crookstown Prison',270,'Male',1983,1984,'Medium',245,111000,116000,'ACT Venture Capital',25,'09-09-1984');
Insert Into Prison Values (3,'Ballyroe Prison',317,'Female',2011,2012,'Low',290,92000,95000,'Sanpaolo IMI Bank Ireland',25,'29-03-2012');
Insert Into Prison Values (4,'Arklow Prison',251,'Male',2000,2002,'Medium',215,101000,113000,'National Credit',14,'04-04-2002');
Insert Into Prison Values (5,'Fenit Prison',353,'Male',2003,2006,'High',315,93000,107000,'HFC Bank',24,'04-07-2006');
Insert Into Prison Values (6,'Ballysax Prison',387,'Male',1995,1996,'Medium',353,105000,105000,'ICC Venture Capital',15,'12-03-1996');
Insert Into Prison Values (7,'Gortnahoo Prison',264,'Male',2006,2008,'High',231,118000,118000,'KEB Ireland',19,'30-06-2008');
Insert Into Prison Values (8,'Causeway Prison',383,'Male',1999,2002,'Medium',346,93000,95000,'AIB Finance And Leasing',19,'19-12-2002');
Insert Into Prison Values (9,'Inistioge Prison',265,'Mixed',1992,1994,'High',224,112000,118000,'ACT Venture Capital',25,'28-09-1994');
Insert Into Prison Values (10,'Carrigtwohill Prison',255,'Mixed',1981,1982,'Low',189,90000,99000,'Sanpaolo IMI Bank Ireland',26,'20-12-1982');
Insert Into Prison Values (11,'Bellanode Prison',380,'Mixed',1999,2000,'Medium',360,120000,123000,'National Credit',12,'18-09-2000');
Insert Into Prison Values (12,'Clifden Prison',291,'Female',2002,2005,'Medium',255,91000,94000,'HFC Bank',11,'08-09-2005');
Insert Into Prison Values (13,'Graiguenamanagh Prison',365,'Male',1986,1989,'High',337,110000,119000,'ACT Venture Capital',10,'30-10-1989');
Insert Into Prison Values (14,'Castlefin Prison',266,'Male',1993,1995,'High',202,101000,113000,'HFC Bank',22,'18-05-1995');
Insert Into Prison Values (15,'Blarney Prison',332,'Mixed',2012,2015,'Medium',305,99000,113000,'AIB Finance And Leasing',27,'29-04-2015');
Insert Into Prison Values (16,'Coolgreany Prison',258,'Female',2014,2015,'Low',197,114000,115000,'ACT Venture Capital',4,'29-11-2015');
Insert Into Prison Values (17,'Delvin Prison',389,'Mixed',2009,2011,'Low',346,109000,115000,'Sanpaolo IMI Bank Ireland',2,'29-05-2011');
Insert Into Prison Values (18,'Ballyshannon Prison',276,'Female',2023,2024,'High',237,90000,95000,'National Credit',13,'05-04-2024');
Insert Into Prison Values (19,'Greencastle Prison',396,'Mixed',1988,1991,'High',337,127000,129000,'HFC Bank',7,'31-03-1991');
Insert Into Prison Values (20,'Clashmore Prison',330,'Female',1985,1986,'High',310,113000,128000,'ICC Venture Capital',9,'22-02-1986');
Insert Into Prison Values (21,'Clonegal Prison',300,'Female',1981,1983,'Low',231,108000,120000,'KEB Ireland',10,'12-08-1983');
Insert Into Prison Values (22,'Dublin City Prison',311,'Mixed',1982,1984,'Medium',280,116000,127000,'National Credit',23,'04-07-1984');

Insert Into Prison Values (23,'Garristown Prison',252,'Mixed',1993,1996,'Medium',187,127000,133000,'HFC Bank',10,'14-04-1996');
Insert Into Prison Values (24,'Broadford Prison',369,'Male',2006,2009,'Medium',316,127000,139000,'Sanpaolo IMI Bank Ireland',8,'03-12-2009');
Insert Into Prison Values (25,'Adare Prison',267,'Male',1982,1985,'Low',243,103000,113000,'National Credit',18,'26-06-1985');
Insert Into Prison Values (26,'Ballyjamesduff Prison',394,'Mixed',1991,1993,'Medium',346,117000,119000,'HFC Bank',25,'24-07-1993');
Insert Into Prison Values (27,'Buttevant Prison',320,'Female',1995,1997,'High',269,94000,108000,'ICC Venture Capital',17,'27-05-1997');
Insert Into Prison Values (28,'Brinlack Prison',393,'Female',1988,1991,'Medium',373,105000,107000,'KEB Ireland',6,'12-09-1991');
Insert Into Prison Values (29,'Ballybay Prison',276,'Mixed',1981,1983,'High',210,97000,101000,'ACT Venture Capital',6,'25-01-1983');
Insert Into Prison Values (30,'Conna Prison',352,'Female',2007,2009,'Low',292,93000,95000,'Sanpaolo IMI Bank Ireland',2,'15-07-2009');
Insert Into Prison Values (31,'Castleisland Prison',269,'Female',1997,1998,'High',202,113000,123000,'National Credit',26,'23-08-1998');
Insert Into Prison Values (32,'Dromiskin Prison',273,'Mixed',2009,2011,'Low',218,125000,135000,'HFC Bank',2,'20-03-2011');
Insert Into Prison Values (33,'Dunfanaghy Prison',277,'Male',2002,2003,'High',250,124000,131000,'ICC Venture Capital',8,'29-10-2003');
Insert Into Prison Values (34,'Ahascragh Prison',354,'Mixed',1992,1993,'Low',326,120000,131000,'KEB Ireland',27,'08-05-1993');
Insert Into Prison Values (35,'Ballyduff Prison',323,'Female',1983,1985,'Low',276,104000,108000,'HFC Bank',21,'02-11-1985');
Insert Into Prison Values (36,'Clones Prison',339,'Male',2001,2003,'High',306,120000,129000,'National Credit',11,'13-10-2003');
Insert Into Prison Values (37,'Balreask Prison',292,'Mixed',2013,2016,'High',238,117000,117000,'Sanpaolo IMI Bank Ireland',4,'15-09-2016');
Insert Into Prison Values (38,'Caherconlish Prison',358,'Mixed',1986,1988,'Medium',337,127000,133000,'National Credit',25,'23-08-1988');
Insert Into Prison Values (39,'Donegal Prison',268,'Female',1997,1998,'Low',229,107000,113000,'HFC Bank',26,'23-07-1998');
Insert Into Prison Values (40,'Ardagh Prison',335,'Female',2022,2023,'High',301,130000,141000,'ICC Venture Capital',5,'16-06-2023');
Insert Into Prison Values (41,'Brownstown Prison',313,'Male',1987,1988,'High',275,91000,92000,'KEB Ireland',7,'15-05-1988');
Insert Into Prison Values (42,'Glenealy Prison',276,'Female',1982,1983,'Medium',233,128000,140000,'AIB Finance And Leasing',17,'27-03-1983');
Insert Into Prison Values (43,'Ballinlough Prison',257,'Mixed',2023,2024,'Medium',226,112000,112000,'ACT Venture Capital',22,'03-09-2024');
Insert Into Prison Values (44,'Ballydesmond Prison',268,'Male',2009,2011,'Low',236,95000,106000,'Sanpaolo IMI Bank Ireland',20,'22-05-2011');
Insert Into Prison Values (45,'Ballygar Prison',366,'Mixed',1997,1999,'Medium',324,120000,135000,'National Credit',1,'01-03-1999');
Insert Into Prison Values (46,'Drimoleague Prison',251,'Mixed',1993,1996,'High',202,95000,102000,'KEB Ireland',9,'23-07-1996');
Insert Into Prison Values (47,'Ballycumber Prison',369,'Male',1989,1991,'High',329,96000,97000,'ICC Venture Capital',7,'22-05-1991');
Insert Into Prison Values (48,'Gowran Prison',373,'Female',2023,2024,'Medium',317,101000,112000,'KEB Ireland',21,'11-09-2024');
Insert Into Prison Values (49,'Ballyconnell Prison',396,'Mixed',1980,1983,'Low',335,114000,121000,'AIB Finance And Leasing',5,'29-01-1983');
Insert Into Prison Values (50,'Ballyhaise Prison',352,'Mixed',1989,1992,'Low',304,101000,116000,'ACT Venture Capital',11,'21-11-1992');
Insert Into Prison Values (51,'Castlemartyr Prison',305,'Mixed',1988,1990,'Low',272,93000,104000,'Sanpaolo IMI Bank Ireland',5,'29-08-1990');
Insert Into Prison Values (52,'Edenderry Prison',357,'Female',1981,1984,'Medium',294,96000,103000,'National Credit',25,'21-03-1984');
Insert Into Prison Values (53,'Johnstown Prison',307,'Female',1990,1992,'Medium',247,96000,102000,'HFC Bank',23,'30-09-1992');
Insert Into Prison Values (54,'Bridge End Prison',296,'Male',1995,1998,'Medium',268,116000,130000,'ICC Venture Capital',8,'02-12-1998');
Insert Into Prison Values (55,'Ballisodare Prison',289,'Female',2002,2005,'High',266,103000,110000,'AIB Finance And Leasing',25,'05-09-2005');
Insert Into Prison Values (56,'Glasslough Prison',388,'Male',2001,2003,'High',366,120000,130000,'KEB Ireland',7,'07-11-2003');
Insert Into Prison Values (57,'Glenamaddy Prison',390,'Mixed',2004,2007,'Medium',344,120000,128000,'Sanpaolo IMI Bank Ireland',23,'15-10-2007');
Insert Into Prison Values (58,'Ballytore Prison',359,'Male',1990,1991,'Medium',323,116000,119000,'National Credit',10,'01-03-1991');
Insert Into Prison Values (59,'Abbeyfeale Prison',308,'Female',1991,1993,'Low',251,102000,108000,'HFC Bank',23,'30-07-1993');
Insert Into Prison Values (60,'Emyvale Prison',319,'Male',2018,2021,'High',254,113000,123000,'ICC Venture Capital',12,'22-08-2021');
Insert Into Prison Values (61,'Baltimore Prison',359,'Male',2021,2022,'Low',311,115000,127000,'ACT Venture Capital',21,'28-12-2022');
Insert Into Prison Values (62,'Cootehill Prison',394,'Male',2003,2005,'High',362,95000,95000,'AIB Finance And Leasing',24,'05-11-2005');
Insert Into Prison Values (63,'Clonlara Prison',304,'Mixed',2012,2015,'High',279,108000,123000,'ACT Venture Capital',26,'06-08-2015');
Insert Into Prison Values (64,'Grangemore Prison',377,'Female',1996,1997,'Medium',312,94000,99000,'Sanpaolo IMI Bank Ireland',21,'14-06-1997');
Insert Into Prison Values (65,'Collon Prison',387,'Mixed',1991,1994,'Medium',358,119000,121000,'National Credit',25,'21-03-1994');
Insert Into Prison Values (66,'Galway City Prison',301,'Female',2016,2018,'Medium',232,109000,124000,'HFC Bank',19,'27-06-2018');
Insert Into Prison Values (67,'Cloonboo Prison',332,'Female',1985,1987,'High',273,116000,119000,'ICC Venture Capital',23,'10-01-1987');
Insert Into Prison Values (68,'Ballymore Eustace Prison',338,'Mixed',1984,1987,'High',314,98000,111000,'KEB Ireland',13,'21-03-1987');
Insert Into Prison Values (69,'Golden Prison',250,'Male',2009,2012,'Low',183,125000,130000,'AIB Finance And Leasing',1,'07-10-2012');
Insert Into Prison Values (70,'Burtonport Prison',356,'Male',2018,2021,'Medium',310,128000,137000,'KEB Ireland',12,'05-06-2021');
Insert Into Prison Values (71,'Coolaney Prison',352,'Female',1997,2000,'High',287,125000,138000,'Sanpaolo IMI Bank Ireland',1,'31-10-2000');
Insert Into Prison Values (72,'Carraroe Prison',303,'Mixed',2018,2020,'Medium',276,100000,115000,'Sanpaolo IMI Bank Ireland',17,'02-10-2020');
Insert Into Prison Values (73,'Drumlish Prison',319,'Male',2005,2007,'Low',269,123000,126000,'KEB Ireland',15,'23-01-2007');
Insert Into Prison Values (74,'Carrickmacross Prison',292,'Mixed',2004,2005,'High',254,103000,106000,'Sanpaolo IMI Bank Ireland',28,'19-12-2005');
Insert Into Prison Values (75,'Annascaul Prison',264,'Mixed',1984,1985,'High',201,102000,102000,'ACT Venture Capital',19,'12-12-1985');
Insert Into Prison Values (76,'Ballydehob Prison',275,'Male',2019,2021,'Medium',224,109000,119000,'AIB Finance And Leasing',26,'18-02-2021');
Insert Into Prison Values (77,'Clonaslee Prison',369,'Female',2012,2013,'Medium',333,126000,135000,'ACT Venture Capital',30,'24-11-2013');
Insert Into Prison Values (78,'Ballyclerahan Prison',340,'Mixed',2001,2002,'High',316,104000,110000,'Sanpaolo IMI Bank Ireland',1,'03-01-2002');
Insert Into Prison Values (79,'Cavan Prison',397,'Mixed',1987,1990,'Medium',351,126000,128000,'National Credit',25,'08-01-1990');
Insert Into Prison Values (80,'Belmullet Prison',311,'Female',2023,2024,'Low',272,125000,140000,'HFC Bank',28,'04-09-2024');
Insert Into Prison Values (81,'Cashel Prison',316,'Male',2020,2021,'Low',271,117000,130000,'ICC Venture Capital',27,'17-05-2021');
Insert Into Prison Values (82,'Ballinroad Prison',278,'Mixed',1981,1984,'Medium',249,124000,126000,'HFC Bank',12,'27-09-1984');
Insert Into Prison Values (83,'Glencullen Prison',297,'Male',2013,2016,'Medium',260,112000,124000,'ICC Venture Capital',25,'13-02-2016');
Insert Into Prison Values (84,'Ballynacargy Prison',377,'Female',1989,1990,'Low',340,118000,131000,'KEB Ireland',14,'26-12-1990');
Insert Into Prison Values (85,'Carrigans Prison',339,'Mixed',2010,2011,'Low',282,97000,105000,'National Credit',5,'30-01-2011');
Insert Into Prison Values (86,'Convoy Prison',354,'Mixed',1985,1986,'Medium',295,116000,131000,'National Credit',22,'12-11-1986');

```
Insert Into Prison Values (87,'Bray Prison',336,'Mixed',2022,2024,'High',283,119000,125000,'ACT Venture Capital',11,'22-09-2024');
Insert Into Prison Values (88,'Courtown Harbour Prison',303,'Male',2024,2024,'Low',236,93000,103000,'National Credit',10,'22-11-2024');
```

6. Analytic functions

Analytic functions are a powerful feature in SQL, combining aggregation capabilities with advanced row-by-row calculations using the Over clause. The Over clause often consists of analytic clauses like Partition By, Order By, and the windowing clause. It's important to distinguish between analytic and aggregate functions: if an aggregate function (like Sum, Avg, etc.) is followed by an Over clause, it becomes an analytic function.

Key Analytic Clauses

Partition By

The Partition By clause divides the result set into partitions (or groups) and applies the analytic function independently within each partition. This is akin to a Group By clause but allows the query to retain individual row details. For example, Sum(Salary) Over (Partition By Department) calculates the total salary within each department while keeping all employee records in the result set.

Order By

The Order By clause specifies the order in which rows are processed within each partition for the analytic function. This clause is essential for functions like Row_Number, Rank, or calculating running totals. For example, Rank() Over (Partition By Department Order By Salary Desc) assigns a rank to each employee within their department based on salary.

Windowing Clause

The windowing clause defines a subset of rows relative to the current row, within which the analytic function operates. It is primarily used with functions like Sum, Avg, and Lag. For instance, Sum(Salary) Over (Order By Hire_Date Rows Between 1 Preceding And Current Row) calculates a running total of salaries, including the current row and the previous one.

Highlighted questions are the ones used in Screencast explaination.

A00325758
Vinit Sharma

Avg() + Partition By + Order By

1) Select Builder id, prison name, prison capacity their average prison capacity of prison based on builder and the difference with average prison capacity. Show the following details only for builder id 1, 3, 8 and 10 . Sort the results by Builder id, Break the results by builder id.

```
cl scr
Break On Builder_Id Skip 1
Select Builder_Id, Prison_Name, Prison_Capacity,
        Avg(Prison_Capacity) Over(Partition By Builder_Id Order By Builder_Id) As
Avg_Prison_Cap,
        Avg(Prison_Capacity) Over(Partition By Builder_Id Order By Builder_Id) - Prison_Capacity
As Avg_Prison_Cap_Diff
From Prison
Where Builder_Id In (1, 3, 8, 10)
Order By Builder_Id;
Clear Breaks;
```

```
VS_SQL>Break On Builder_Id Skip 1
VS_SQL>Select Builder_Id, Prison_Name, Prison_Capacity,
  2     Avg(Prison_Capacity) Over(Partition By Builder_Id Order By Builder_Id) As Avg_Prison_Cap,
  3     Avg(Prison_Capacity) Over(Partition By Builder_Id Order By Builder_Id) - Prison_Capacity As Avg_Prison_Cap_Diff
  4  From Prison
  5  Where Builder_Id In (1, 3, 8, 10)
  6  Order By Builder_Id;

BUILDER_ID PRISON_NAME              PRISON_CAPACITY AVG_PRISON_CAP AVG_PRISON_CAP_DIFF
---------- ------------------------ --------------- -------------- -------------------
         1 Ballygar Prison                      366            327                 -39
           Ballyclerahan Prison                 340            327                 -13
           Coolaney Prison                      352            327                 -25
           Golden Prison                        250            327                  77

         8 Dunfanaghy Prison                    277            314                  37
           Bridge End Prison                    296            314                  18
           Broadford Prison                     369            314                 -55

        10 Courtown Harbour Prison              303          315.8                12.8
           Ballytore Prison                     359          315.8               -43.2
           Garristown Prison                    252          315.8                63.8
           Clonegal Prison                      300          315.8                15.8
           Graiguenamanagh Prison               365          315.8               -49.2


12 rows selected.

VS_SQL>Clear Breaks;
breaks cleared
VS_SQL>
```

This query will return aggregated output based on partition. That is the data is partitioned using Builder_Id, So the query will return Average Prison_Capacity Based on Builder_Id. Additionally, break setting is used in this query so output will skip a line after each Builder_Id block created by Partition. Remember to Clear Breaks after its application.

Max() + Min() + Partition By + Order By

2) Display Funding source, Prison construction that it funded, actual costing calculated for prison, Highest and Lowest funding done by the source. Get the data only for prisons with name starting with 'Bally' .Sort the data by Funding source and then by actual funding cost. Break the results by funding source.

```
cl scr
Break On Funding_Source Skip 1
Select Funding_Source, Prison_Name, Actual_Cost,
        Max(Actual_Cost) Over(Partition By Funding_Source Order By Funding_Source) As
Max_Amount_Funded,
        Min(Actual_Cost) Over(Partition By Funding_Source Order By Funding_Source) As
Min_Amount_Funded
From Prison
Where Prison_Name Like ('Bally%')
Order By Funding_Source, Actual_Cost;
Clear Breaks;
```

```
VS_SQL>Break On Funding_Source Skip 1
VS_SQL>Select Funding_Source, Prison_Name, Actual_Cost,
  2     Max(Actual_Cost) Over(Partition By Funding_Source Order By Funding_Source) As Max_Amount_Funded,
  3  Min(Actual_Cost) Over(Partition By Funding_Source Order By Funding_Source) As Min_Amount_Funded
  4  From Prison
  5  Where Prison_Name Like ('Bally%')
  6  Order By Funding_Source, Actual_Cost;

FUNDING_SOURCE             PRISON_NAME              ACTUAL_COST MAX_AMOUNT_FUNDED MIN_AMOUNT_FUNDED
-------------------------- ------------------------ ----------- ----------------- -----------------
ACT Venture Capital        Ballybay Prison               101000            116000            101000
                           Ballyhaise Prison             116000            116000            101000

AIB Finance And Leasing    Ballydehob Prison             119000            121000            119000
                           Ballyconnell Prison           121000            121000            119000

HFC Bank                   Ballyduff Prison              108000            119000            108000
                           Ballyjamesduff Prison         119000            119000            108000

ICC Venture Capital        Ballycumber Prison             97000            105000             97000
                           Ballysax Prison               105000            105000             97000

KEB Ireland                Ballymore Eustace Prison      111000            131000            111000
                           Ballynacargy Prison           131000            131000            111000

National Credit            Ballyshannon Prison            95000            135000             95000
                           Ballytore Prison              119000            135000             95000
                           Ballygar Prison               135000            135000             95000

Sanpaolo IMI Bank Ireland  Ballyroe Prison                95000            110000             95000
                           Ballydesmond Prison           106000            110000             95000
                           Ballyclerahan Prison          110000            110000             95000

Sanpaolo IMI Bank Ireland  Ballyroe Prison                95000            110000             95000
                           Ballydesmond Prison           106000            110000             95000
                           Ballyclerahan Prison          110000            110000             95000


16 rows selected.

VS_SQL>Clear Breaks;
breaks cleared
VS_SQL>
```

First_Value() + Order By + Windowing clause

3) Get Prison name, date prison opened, its actual cost, opening date and actual cost for first prison opened between current prison opening date and 1 year before. Show details for prison opened after 2020 only. Sort the output by opening date in reverse.

```
Cl Scr
Select Prison_Name, Date_Opened, Actual_Cost,
        First_Value(Date_Opened) Over(Order By Date_Opened
                Range 365 Preceding) As Prev_Year_Opening_Date,
        First_Value(Actual_Cost) Over(Order By Date_Opened
                Range 365 Preceding) As Prev_Year_Actual_Cost
From Prison
Where Date_Opened > To_Date('31-12-2020', 'DD-MM-YYYY')
Order By Date_Opened Desc;
```

```
VS_SQL>Select Prison_Name, Date_Opened, Actual_Cost,
  2     First_Value(Date_Opened) Over(Order By Date_Opened
  3  Range 365 Preceding) As Prev_Year_Opening_Date,
  4     First_Value(Actual_Cost) Over(Order By Date_Opened
  5  Range 365 Preceding) As Prev_Year_Actual_Cost
  6  From Prison
  7  Where Date_Opened > To_Date('31-12-2020', 'DD-MM-YYYY')
  8  Order By Date_Opened Desc;

PRISON_NAME              DATE_OPENE ACTUAL_COST PREV_YEAR_ PREV_YEAR_ACTUAL_COST
------------------------ ---------- ----------- ---------- --------------------
Courtown Harbour Prison  22-11-2024      103000 05-04-2024                95000
Bray Prison              22-09-2024      125000 05-04-2024                95000
Gowran Prison            11-09-2024      112000 05-04-2024                95000
Belmullet Prison         04-09-2024      140000 05-04-2024                95000
Ballinlough Prison       03-09-2024      112000 05-04-2024                95000
Ballyshannon Prison      05-04-2024       95000 16-06-2023               141000
Ardagh Prison            16-06-2023      141000 28-12-2022               127000
Baltimore Prison         28-12-2022      127000 28-12-2022               127000
Emyvale Prison           22-08-2021      123000 18-02-2021               119000
Burtonport Prison        05-06-2021      137000 18-02-2021               119000
Cashel Prison            17-05-2021      130000 18-02-2021               119000
Ballydehob Prison        18-02-2021      119000 18-02-2021               119000

12 rows selected.

VS_SQL>
```

In this query a subset will be created by windowing clause based on Date_Opened. This subset will consist of all the data on prison opened in last 365 days with respect to a particular prison, this subset will be sorted by Date_Opened using Order By clause and First_Value() will Find out the first record(Earliest Date_Opened record) from the subset created.

Dense_Rank() + Order By

4) Display data for prison(ID), Capacity, security, Number of staffs at prison, and rank the prison based on number of staffs high to low. Number of staffs in prison should be between 200 and 220. Order by number of staffs in the prison in reverse.

```
Cl Scr
Select Prison_Id, Prison_Capacity, Prison_Security, Num_Officers,
        Dense_Rank() Over(Order By Num_Officers Desc) As Num_Staffs_Ranked
From Prison
Where Num_Officers Between 200 And 220
Order By Num_Officers Desc;
```

```
VS_SQL>Select Prison_Id, Prison_Capacity, Prison_Security, Num_Officers,
  2     Dense_Rank() Over(Order By Num_Officers Desc) As Num_Staffs_Ranked
  3  From Prison
  4  Where Num_Officers Between 200 And 220
  5  Order By Num_Officers Desc;

 PRISON_ID PRISON_CAPACITY PRISON NUM_OFFICERS NUM_STAFFS_RANKED
---------- --------------- ------ ------------ -----------------
        32             273 Low             218                 1
         4             251 Medium          215                 2
        29             276 High            210                 3
        14             266 High            202                 4
        31             269 High            202                 4
        46             251 High            202                 4
        75             264 High            201                 5

7 rows selected.

VS_SQL>
```

Dense_Rank() Function gives out rank based on attribute mentioned in Order By clause. When two or more of the values are same in the Order By attribute, it ranks them the same, and then increament the rank by 1 for the next different value. I use Dense_Rank() As I find it more appropriate than Rank() Function. As in similar conditions Rank() function would have skip some ranks if some of the values are repeated.

Dense_Rank() + Partition By + Order By

5) Display Builder id, prison name, prison security level and its Actual cost. Also rank the prison based on their actual cost grouped in prison security level. Limit the data where actual cost is between 95000 And 100000. Order the results by prison security and then by rank.

```
Cl Scr
Select Builder_Id, Prison_Name, Prison_Security, Actual_Cost,
        Dense_Rank() Over(Partition By Prison_Security Order By Actual_Cost) As
Annual_Cost_Ranking
From Prison
Where Actual_Cost Between 95000 And 100000
Order By Prison_Security;
```

```
VS_SQL>Select Builder_Id, Prison_Name, Prison_Security, Actual_Cost,
  2     Dense_Rank() Over(Partition By Prison_Security Order By Actual_Cost) As Annual_Cost_Ranking
  3  From Prison
  4  Where Actual_Cost Between 95000 And 100000
  5  Order By Prison_Security;

BUILDER_ID PRISON_NAME              PRISON ACTUAL_COST ANNUAL_COST_RANKING
---------- ------------------------ ------ ----------- -------------------
        13 Ballyshannon Prison      High         95000                   1
        24 Cootehill Prison         High         95000                   1
         7 Ballycumber Prison       High         97000                   2
         2 Conna Prison             Low          95000                   1
        25 Ballyroe Prison          Low          95000                   1
        26 Carrigtwohill Prison     Low          99000                   2
        19 Causeway Prison          Medium       95000                   1
        21 Grangemore Prison        Medium       99000                   2

8 rows selected.

VS_SQL>
```

Dense_Rank() + Partition By + Order By

6) Select Prison name, Gender, Actual cost required from prisons table. Rank this data from most expensive to cheap prison based on gender. Display only those prison which cost more than 133000. Sort by Gender and Actual cost in reverse.

```
Cl scr
Select Prison_Name, Gender, Actual_Cost,
        Dense_Rank() Over(Partition By Gender Order By Actual_Cost Desc) As
Most_Expensive_Prison
From Prison
Where Actual_Cost > 133000
Order By Gender, Actual_Cost Desc ;
```

```
VS_SQL>Select Prison_Name, Gender, Actual_Cost,
  2     Dense_Rank() Over(Partition By Gender Order By Actual_Cost Desc) As Most_Expensive_Prison
  3  From Prison
  4  Where Actual_Cost > 133000
  5  Order By Gender, Actual_Cost Desc ;

PRISON_NAME              GENDER ACTUAL_COST MOST_EXPENSIVE_PRISON
------------------------ ------ ----------- ---------------------
Ardagh Prison            Female      141000                     1
Glenealy Prison          Female      140000                     2
Belmullet Prison         Female      140000                     2
Coolaney Prison          Female      138000                     3
Clonaslee Prison         Female      135000                     4
Broadford Prison         Male        139000                     1
Burtonport Prison        Male        137000                     2
Dromiskin Prison         Mixed       135000                     1
Ballygar Prison          Mixed       135000                     1

9 rows selected.

VS_SQL>
```

Min() + Partition By

7)Display Prison id, name, security and number of officers working there. Also Mention the lowest number of officers working for each security type. Look for Prisons having less than 215 Officers on premises. Order data by security type.

Cl Scr

```
Select Prison_ID, Prison_Name, Prison_Security, Num_Officers,
        Min(Num_Officers) Over(Partition By Prison_Security) As Min_Guards
From Prison
Where Num_Officers < 215
Order By Prison_Security;
```

```
VS_SQL>
VS_SQL>Select Prison_ID, Prison_Name, Prison_Security, Num_Officers,
  2     Min(Num_Officers) Over(Partition By Prison_Security) As Min_Guards
  3  From Prison
  4  Where Num_Officers < 215
  5  Order By Prison_Security;

 PRISON_ID PRISON_NAME              PRISON NUM_OFFICERS MIN_GUARDS
---------- ------------------------ ------ ------------ ----------
        29 Ballybay Prison          High            210        201
        46 Drimoleague Prison       High            202        201
        31 Castleisland Prison      High            202        201
        75 Annascaul Prison         High            201        201
        14 Castlefin Prison         High            202        201
        16 Coolgreany Prison        Low             197        183
        69 Golden Prison            Low             183        183
        10 Carrigtwohill Prison     Low             189        183
        23 Garristown Prison        Medium          187        187

9 rows selected.

VS_SQL>
```

Dense_Rank() + Partition By + Order By

8)Select Prison details like id, name, gender from prison table. Mention the difference between actual and projected cost in these prisons and rank them by the most difference based on gender. Filter the data where difference of cost is more than 13000. Sort the output by gender and then Difference of costs in reverse.

```
Cl scr
Select Prison_Id, Prison_Name, Gender, Actual_Cost – Projected_Cost As Costing_Difference,
        Dense_Rank() Over(Partition By Gender Order By Actual_Cost – Projected_Cost Desc) As
Cost_Exceeded_Rank
From Prison
Where Actual_Cost – Projected_Cost > 13000
Order By Gender, Costing_Difference Desc;
```

```
VS_SQL>Select Prison_Id, Prison_Name, Gender, Actual_Cost - Projected_Cost As Costing_Difference,
  2     Dense_Rank() Over(Partition By Gender Order By Actual_Cost - Projected_Cost Desc) As Cost_Exceeded_Rank
  3  From Prison
  4  Where Actual_Cost - Projected_Cost > 13000
  5  Order By Gender, Costing_Difference Desc;

 PRISON_ID PRISON_NAME             GENDER COSTING_DIFFERENCE COST_EXCEEDED_RANK
---------- ----------------------- ------ ------------------ ------------------
        66 Galway City Prison      Female              15000                  1
        20 Clashmore Prison        Female              15000                  1
        80 Belmullet Prison        Female              15000                  1
        27 Buttevant Prison        Female              14000                  2
        54 Bridge End Prison       Male                14000                  1
         5 Fenit Prison            Male                14000                  1
        72 Carraroe Prison         Mixed               15000                  1
        63 Clonlara Prison         Mixed               15000                  1
        86 Convoy Prison           Mixed               15000                  1
        45 Ballygar Prison         Mixed               15000                  1
        50 Ballyhaise Prison       Mixed               15000                  1
        15 Blarney Prison          Mixed               14000                  2

12 rows selected.

VS_SQL>
```

Avg() + Partition By

9) Display Prison name, security level, actual cost, average cost according to security level, and difference between actual cost and average cost from prison table. Filter the result where actual costing is less than 98000. Order by security level in reverse.

Cl scr

```
Select Prison_Name, Prison_Security, Actual_Cost,
        Avg(Actual_Cost) Over(Partition By Prison_Security) As Avg_Actual_Cost,
        Actual_Cost - Avg(Actual_Cost) Over(Partition By Prison_Security) As Difference_Avg_Cost
From Prison
Where Actual_Cost < 98000
Order By Prison_Security;
```

```
VS_SQL>Select Prison_Name, Prison_Security, Actual_Cost,
  2     Avg(Actual_Cost) Over(Partition By Prison_Security) As Avg_Actual_Cost,
  3     Actual_Cost - Avg(Actual_Cost) Over(Partition By Prison_Security) As Difference_Avg_Cost
  4  From Prison
  5  Where Actual_Cost < 98000
  6  Order By Prison_Security;

PRISON_NAME              PRISON ACTUAL_COST AVG_ACTUAL_COST DIFFERENCE_AVG_COST
------------------------ ------ ----------- --------------- -------------------
Ballyshannon Prison      High         95000           94750                 250
Ballycumber Prison       High         97000           94750                2250
Brownstown Prison        High         92000           94750               -2750
Cootehill Prison         High         95000           94750                 250
Conna Prison             Low          95000           95000                   0
Ballyroe Prison          Low          95000           95000                   0
Clifden Prison           Medium       94000           94500                -500
Causeway Prison          Medium       95000           94500                 500

8 rows selected.

VS_SQL>
```

Dense_Rank() + Order By

10) Rank the prisons based on their projected cost, use prison id for tie situation. Display columns like id, name, projected cost and ranking for prisons with name starting with letter ‘G’. Order the output by projected cost in reverse order.

```
Cl scr
Select Prison_Id, Prison_Name, Projected_Cost,
        Dense_Rank() Over(Order By Projected_Cost Desc) As Highest_Proj_Cost
From Prison
Where Prison_Name Like 'G%'
Order By Projected_Cost Desc;
```

```
VS_SQL>Select Prison_Id, Prison_Name, Projected_Cost,
  2     Dense_Rank() Over(Order By Projected_Cost Desc) As Highest_Proj_Cost
  3  From Prison
  4  Where Prison_Name Like 'G%'
  5  Order By Projected_Cost Desc;

 PRISON_ID PRISON_NAME              PROJECTED_COST HIGHEST_PROJ_COST
---------- ------------------------ -------------- -----------------
        42 Glenealy Prison                  128000                 1
        19 Greencastle Prison               127000                 2
        23 Garristown Prison                127000                 2
        69 Golden Prison                    125000                 3
        57 Glenamaddy Prison                120000                 4
        56 Glasslough Prison                120000                 4
         7 Gortnahoo Prison                 118000                 5
        83 Glencullen Prison                112000                 6
        13 Graiguenamanagh Prison           110000                 7
        66 Galway City Prison               109000                 8
        48 Gowran Prison                    101000                 9
        64 Grangemore Prison                 94000                10

12 rows selected.

VS_SQL>
```

Max() + Min() + Partition By

11) State the funding source, prison id, actual cost and the difference between their maximum and minimum funding(actual cost). Break the results by funding source.

Cl scr

```
Break On Funding_Source Skip 1
Select Funding_Source, Prison_Id, Actual_Cost,
        Max(Actual_Cost) Over(Partition By Funding_Source)  –
        Min(Actual_Cost) Over(Partition By Funding_Source) As Cost_Difference
From Prison
Where Actual_Cost Between 120000 And 125000
Order By Funding_Source;
Clear Breaks;
```

```
VS_SQL>
VS_SQL>Break On Funding_Source Skip 1
VS_SQL>Select Funding_Source, Prison_Id, Actual_Cost,
  2     Max(Actual_Cost) Over(Partition By Funding_Source)  -
  3     Min(Actual_Cost) Over(Partition By Funding_Source) As Cost_Difference
  4  From Prison
  5  Where Actual_Cost Between 120000 And 125000
  6  Order By Funding_Source;

FUNDING_SOURCE             PRISON_ID ACTUAL_COST COST_DIFFERENCE
------------------------- ---------- ----------- ---------------
ACT Venture Capital               63      123000            2000
                                  87      125000            2000

AIB Finance And Leasing           49      121000               0

HFC Bank                          66      124000               0

ICC Venture Capital               60      123000            1000
                                  83      124000            1000

KEB Ireland                       21      120000               0

National Credit                   65      121000            2000
                                  11      123000            2000
                                  31      123000            2000


10 rows selected.

VS_SQL>Clear Breaks;
breaks cleared
VS_SQL>
```

Count() + Partition By

12) Get prison id, Construction start year and end year, opening date and count of prison opened in end year. Filter data to get details related prisons which construction ended between 1988 and 1993. Order By end year.

```
Cl Scr
Select Prison_Id, Start_Year, End_Year, Date_Opened,
        Count(*) Over(Partition By End_Year) As Num_Prisons_Opened
From Prison
Where End_Year Between 1988 And 1993
Order By End_Year;
```

```
VS_SQL>Select Prison_Id, Start_Year, End_Year, Date_Opened,
  2     Count(*) Over(Partition By End_Year) As Num_Prisons_Opened
  3  From Prison
  4  Where End_Year Between 1988 And 1993
  5  Order By End_Year;

 PRISON_ID START_YEAR   END_YEAR DATE_OPENE NUM_PRISONS_OPENED
---------- ---------- ---------- ---------- ------------------
        41       1987       1988 15-05-1988                  2
        38       1986       1988 23-08-1988                  2
        13       1986       1989 30-10-1989                  1
        84       1989       1990 26-12-1990                  3
        79       1987       1990 08-01-1990                  3
        51       1988       1990 29-08-1990                  3
        47       1989       1991 22-05-1991                  4
        28       1988       1991 12-09-1991                  4
        19       1988       1991 31-03-1991                  4
        58       1990       1991 01-03-1991                  4
        53       1990       1992 30-09-1992                  2
        50       1989       1992 21-11-1992                  2
        26       1991       1993 24-07-1993                  3
        59       1991       1993 30-07-1993                  3
        34       1992       1993 08-05-1993                  3

15 rows selected.

VS_SQL>
```

Count function will return count of records per partition.

Lead() + Partition By + Order By

13) Select Prison id, name, security type, capacity of inmates. Also find the next higher capacity prison within the same security level. Filter the data to get prison info for names starting with ‘D’. Order By Security.

Cl scr

```
Select Prison_Id, Prison_Name, Prison_Security, Prison_Capacity,
        Lead(Prison_Capacity) Over(Partition By Prison_Security Order By Prison_Capacity) As High_Inmates_Count
From Prison
Where Prison_Name Like 'D%'
Order By Prison_Security;
```

```
VS_SQL>
VS_SQL>Select Prison_Id, Prison_Name, Prison_Security, Prison_Capacity,
  2     Lead(Prison_Capacity) Over(Partition By Prison_Security Order By Prison_Capacity) As High_Inmates_Count
  3  From Prison
  4  Where Prison_Name Like 'D%'
  5  Order By Prison_Security;

 PRISON_ID PRISON_NAME              PRISON PRISON_CAPACITY HIGH_INMATES_COUNT
---------- ------------------------ ------ --------------- ------------------
        46 Drimoleague Prison       High               251                277
        33 Dunfanaghy Prison        High               277
        39 Donegal Prison           Low                268                273
        32 Dromiskin Prison         Low                273                319
        73 Drumlish Prison          Low                319                389
        17 Delvin Prison            Low                389
        22 Dublin City Prison       Medium             311

7 rows selected.

VS_SQL>
```

Lead() will return the immediate next value of attribute mentioned inside of it, based on partition level.

Avg() + Partition By

14) Select prison id, prison name, prison security gender, actual cost, and running cost for last 3 rows. Find this data for low security level and prison name starting with ‘C’. Sort By Gender, prison id and than actual cost.

Cl scr

```
Select Prison_Id, Prison_Name, Prison_Security, Gender, Actual_Cost,
        Avg(Actual_Cost) Over(Partition By Gender Order By Prison_Id
        Rows Between 2 Preceding And Current Row) As Running_Avg
From Prison
Where Prison_Security = 'Low'
And Prison_Name Like 'C%'
Order By Gender, Prison_Id, Actual_Cost;
```

```
VS_SQL>Select Prison_Id, Prison_Name, Prison_Security, Gender, Actual_Cost,
  2     Avg(Actual_Cost) Over(Partition By Gender Order By Prison_Id
  3     Rows Between 2 Preceding And Current Row) As Running_Avg
  4  From Prison
  5  Where Prison_Security = 'Low'
  6  And Prison_Name Like 'C%'
  7  Order By Gender, Prison_Id, Actual_Cost;

 PRISON_ID PRISON_NAME              PRISON GENDER ACTUAL_COST RUNNING_AVG
---------- ------------------------ ------ ------ ----------- -----------
        16 Coolgreany Prison        Low    Female      115000      115000
        21 Clonegal Prison          Low    Female      120000      117500
        30 Conna Prison             Low    Female       95000      110000
        81 Cashel Prison            Low    Male        130000      130000
        88 Courtown Harbour Prison  Low    Male        103000      116500
        10 Carrigtwohill Prison     Low    Mixed        99000       99000
        51 Castlemartyr Prison      Low    Mixed       104000      101500
        85 Carrigans Prison         Low    Mixed       105000  102666.667

8 rows selected.

VS_SQL>
```

Avg() + Partition By

15) Give prison details like id, name, security level, number of staffs from prison table. Also display Difference between prison's number of staffs and average staffs based on prison security. Filter the data where Number of officers in a prison is between 230 and 240. Order the data by prison security level.

Cl scr

```
Select Prison_Id, Prison_Name, Prison_Security, Num_Officers,
        Num_Officers  - Avg(Num_Officers) Over(Partition By Prison_Security) As
Difference_Avg_Staffs
From Prison
Where Num_Officers Between 230 And 240
Order By Prison_Security;
```

```
VS_SQL>
VS_SQL>Select Prison_Id, Prison_Name, Prison_Security, Num_Officers,
  2     Num_Officers  - Avg(Num_Officers) Over(Partition By Prison_Security) As Difference_Avg_Staffs
  3  From Prison
  4  Where Num_Officers Between 230 And 240
  5  Order By Prison_Security;

 PRISON_ID PRISON_NAME              PRISON NUM_OFFICERS DIFFERENCE_AVG_STAFFS
---------- ------------------------ ------ ------------ --------------------
         7 Gortnahoo Prison         High            231            -4.3333333
        18 Ballyshannon Prison      High            237            1.66666667
        37 Balreask Prison          High            238            2.66666667
        88 Courtown Harbour Prison  Low             236            1.66666667
        44 Ballydesmond Prison      Low             236            1.66666667
        21 Clonegal Prison          Low             231            -3.3333333
        42 Glenealy Prison          Medium          233                    .5
        66 Galway City Prison       Medium          232                   -.5

8 rows selected.

VS_SQL>
```

Lead() + Partition By + Order By

16) Select Prison id, name, gender, construction end year, opening date and the date when the next prison is opened with same gender. Show the results only for prisons which was completed in construction between 1997 and 2005. Sort result by Gender and Opening date. Break the results by gender.

```
Cl scr
Break On Gender Skip 1
Select Prison_Id, Prison_Name, Gender, End_Year, Date_Opened,
        Lead(Date_Opened) Over(Partition By Gender Order By Date_Opened) As
Next_Prison_Opened
From Prison
Where End_Year Between 1997 And 2005
Order By Gender, Date_Opened;
Clear Breaks;
```

```
VS_SQL>Break On Gender Skip 1
VS_SQL>Select Prison_Id, Prison_Name, Gender, End_Year, Date_Opened,
  2     Lead(Date_Opened) Over(Partition By Gender Order By Date_Opened) As Next_Prison_Opened
  3  From Prison
  4  Where End_Year Between 1997 And 2005
  5  Order By Gender, Date_Opened;

 PRISON_ID PRISON_NAME              GENDER   END_YEAR DATE_OPENE NEXT_PRISO
---------- ------------------------ ------ ---------- ---------- ----------
        27 Buttevant Prison         Female       1997 27-05-1997 14-06-1997
        64 Grangemore Prison                     1997 14-06-1997 23-07-1998
        39 Donegal Prison                        1998 23-07-1998 23-08-1998
        31 Castleisland Prison                   1998 23-08-1998 31-10-2000
        71 Coolaney Prison                       2000 31-10-2000 05-09-2005
        55 Ballisodare Prison                    2005 05-09-2005 08-09-2005
        12 Clifden Prison                        2005 08-09-2005

        54 Bridge End Prison        Male         1998 02-12-1998 04-04-2002
         4 Arklow Prison                         2002 04-04-2002 19-12-2002
         8 Causeway Prison                       2002 19-12-2002 13-10-2003
        36 Clones Prison                         2003 13-10-2003 29-10-2003
        33 Dunfanaghy Prison                     2003 29-10-2003 07-11-2003
        56 Glasslough Prison                     2003 07-11-2003 05-11-2005
        62 Cootehill Prison                      2005 05-11-2005

        45 Ballygar Prison          Mixed        1999 01-03-1999 18-09-2000
        11 Bellanode Prison                      2000 18-09-2000 27-07-2001
         1 Banagher Prison                       2001 27-07-2001 03-01-2002
        78 Ballyclerahan Prison                  2002 03-01-2002 19-12-2005
        74 Carrickmacross Prison                 2005 19-12-2005

19 rows selected.

VS_SQL>Clear Breaks;
breaks cleared
VS_SQL>
```

Sum() + Partition By + Order By

17) Display the prison id, name, security type, number of officers, and running total of officers count. Filter the data by number of officers between 270 and 281. Sort the result by prison security and then by number of officers.

```
Cl scr
Select Prison_Id, Prison_Name, Prison_Security, Num_Officers,
        Sum(Num_Officers) Over(Partition By Prison_Security Order By Num_Officers) As
Running_Total_Officers
From Prison
Where Num_Officers Between 270 And 281
Order By Prison_Security, Num_Officers;
```

```
VS_SQL>Select Prison_Id, Prison_Name, Prison_Security, Num_Officers,
  2     Sum(Num_Officers) Over(Partition By Prison_Security Order By Num_Officers) As Running_Total_Officers
  3  From Prison
  4  Where Num_Officers Between 270 And 281
  5  Order By Prison_Security, Num_Officers;

 PRISON_ID PRISON_NAME              PRISON NUM_OFFICERS RUNNING_TOTAL_OFFICERS
---------- ------------------------ ------ ------------ ----------------------
        67 Cloonboo Prison          High            273                    273
        41 Brownstown Prison        High            275                    548
        63 Clonlara Prison          High            279                    827
        81 Cashel Prison            Low             271                    271
        51 Castlemartyr Prison      Low             272                    815
        80 Belmullet Prison         Low             272                    815
        35 Ballyduff Prison         Low             276                   1091
        72 Carraroe Prison          Medium          276                    276
        22 Dublin City Prison       Medium          280                    556

9 rows selected.

VS_SQL>
```

Lead() + Partition By + Order By

18) Find out name, gender of prisons with low security based on gender. Sort the data by gender and than by number of officers in reverse order. Also show that by how much extra count of officers the prison have compared to their immediate next. Break the results by gender and lastly display the actual cost of these prisons.

```
Cl scr
Break On Gender Skip 1
Select Prison_Name, Gender , Prison_Security, Num_Officers, Actual_Cost,
        Num_Officers – Lead(Num_Officers) Over(Partition By Gender Order By Num_Officers Desc)
As OfficerDiff
From Prison
Where Prison_Security = 'Low'
Order By Gender, Num_Officers Desc;
Clear Breaks;
```

```
VS_SQL>Break On Gender Skip 1
VS_SQL>Select Prison_Name, Gender , Prison_Security, Num_Officers, Actual_Cost,
  2     Num_Officers - Lead(Num_Officers) Over(Partition By Gender Order By Num_Officers Desc) As OfficerDiff
  3  From Prison
  4  Where Prison_Security = 'Low'
  5  Order By Gender, Num_Officers Desc;

PRISON_NAME              GENDER PRISON NUM_OFFICERS ACTUAL_COST OFFICERDIFF
------------------------ ------ ------ ------------ ----------- -----------
Ballynacargy Prison      Female Low             340      131000          48
Conna Prison                    Low             292       95000           2
Ballyroe Prison                 Low             290       95000          14
Ballyduff Prison                Low             276      108000           4
Belmullet Prison                Low             272      140000          21
Abbeyfeale Prison               Low             251      108000          20
Clonegal Prison                 Low             231      120000           2
Donegal Prison                  Low             229      113000          32
Coolgreany Prison               Low             197      115000

Baltimore Prison         Male   Low             311      127000          40
Cashel Prison                   Low             271      130000           2
Drumlish Prison                 Low             269      126000          26
Adare Prison                    Low             243      113000           7
Courtown Harbour Prison         Low             236      103000           0
Ballydesmond Prison             Low             236      106000          53
Golden Prison                   Low             183      130000

Delvin Prison            Mixed  Low             346      115000          11
Ballyconnell Prison             Low             335      121000           9
Ahascragh Prison                Low             326      131000          22
Ballyhaise Prison               Low             304      116000          22
Carrigans Prison                Low             282      105000          10
```

```
Delvin Prison            Mixed  Low             346      115000          11
Ballyconnell Prison             Low             335      121000           9
Ahascragh Prison                Low             326      131000          22
Ballyhaise Prison               Low             304      116000          22
Carrigans Prison                Low             282      105000          10
Castlemartyr Prison             Low             272      104000          54
Dromiskin Prison                Low             218      135000          29
Carrigtwohill Prison            Low             189       99000


24 rows selected.

VS_SQL>Clear Breaks;
breaks cleared
VS_SQL>
```

Sum() + Order By + Windowing clause

19) Display id, opening date, capacity of prison and its actual cost of prisons which opened last 10 years(on or after 2014). Additionally find out how much prisoner capacity has been increased in Ireland in last one year along with how much has been spent on these prisons in last one year. Sort the output by opening date.

```
Cl scr
Select Prison_Id, Date_Opened, Prison_Capacity, Actual_Cost,
        Sum(Prison_Capacity) Over(Order By Date_Opened Range 365 Preceding) As priorYearCap,
        Sum(Actual_Cost) Over(Order By Date_Opened Range 365 Preceding) As priorYearCost
From Prison
Where Date_Opened >= To_Date('2014-01-01', 'YYYY-MM-DD')
Order By Date_Opened;
```

```
VS_SQL>Select Prison_Id, Date_Opened, Prison_Capacity, Actual_Cost,
  2     Sum(Prison_Capacity) Over(Order By Date_Opened Range 365 Preceding) As priorYearCap,
  3     Sum(Actual_Cost) Over(Order By Date_Opened Range 365 Preceding) As priorYearCost
  4  From Prison
  5  Where Date_Opened >= To_Date('2014-01-01', 'YYYY-MM-DD')
  6  Order By Date_Opened;

 PRISON_ID DATE_OPENE PRISON_CAPACITY ACTUAL_COST PRIORYEARCAP PRIORYEARCOST
---------- ---------- --------------- ----------- ------------ -------------
        15 29-04-2015             332      113000          332        113000
        63 06-08-2015             304      123000          636        236000
        16 29-11-2015             258      115000          894        351000
        83 13-02-2016             297      124000         1191        475000
        37 15-09-2016             292      117000          847        356000
        66 27-06-2018             301      124000          301        124000
        72 02-10-2020             303      115000          303        115000
        76 18-02-2021             275      119000          578        234000
        81 17-05-2021             316      130000          894        364000
        70 05-06-2021             356      137000         1250        501000
        60 22-08-2021             319      123000         1569        624000
        61 28-12-2022             359      127000          359        127000
        40 16-06-2023             335      141000          694        268000
        18 05-04-2024             276       95000          611        236000
        43 03-09-2024             257      112000          533        207000
        80 04-09-2024             311      140000          844        347000
        48 11-09-2024             373      112000         1217        459000
        87 22-09-2024             336      125000         1553        584000
        88 22-11-2024             303      103000         1856        687000

19 rows selected.

VS_SQL>
```

Ratio_To_Report() + Partition By

20) Get data for prisons funded by AIB Finance And Leasing, ACT Venture Capital and KEB Ireland. Display funding sorce, builder id, prison id, actual cost and ratio of funding a source has done on a particular prison. Order data by actual cost. Break the data by funding source

```
Cl scr
Break On Funding_Source Skip 1
Select Funding_Source, Builder_Id, Prison_Id, Actual_Cost,
        Ratio_To_Report(Actual_Cost) Over(Partition By Funding_Source) As Funded_Ratio
From Prison
Where Funding_Source In ('AIB Finance And Leasing', 'ACT Venture Capital', 'KEB Ireland')
Order By Funding_Source;
Clear Breaks;
```

```
VS_SQL>Break On Funding_Source Skip 1
VS_SQL>Select Funding_Source, Builder_Id, Prison_Id, Actual_Cost,
  2     Ratio_To_Report(Actual_Cost) Over(Partition By Funding_Source) As Funded_Ratio
  3  From Prison
  4  Where Funding_Source In ('AIB Finance And Leasing', 'ACT Venture Capital', 'KEB Ireland')
  5  Order By Funding_Source;

FUNDING_SOURCE            BUILDER_ID  PRISON_ID ACTUAL_COST FUNDED_RATIO
------------------------- ---------- ---------- ----------- ------------
ACT Venture Capital               25          9      118000   .083747339
                                  30         77      135000   .095812633
                                  11         50      116000   .082327892
                                  11         87      125000   .088715401
                                  22         43      112000   .079488999
                                  10         13      119000   .084457062
                                   4         16      115000   .081618169
                                  21         61      127000   .090134847
                                  26         63      123000   .087295955
                                  19         75      102000   .072391767
                                   6         29      101000   .071682044
                                  25          2      116000   .082327892

AIB Finance And Leasing           27         15      113000   .122426869
                                   5         49      121000   .131094258
                                   1         69      130000    .14084507
                                  19          8       95000   .102925244
                                  17         42      140000   .151679307
                                  24         62       95000   .102925244
                                  26         76      119000   .128927411
                                  25         55      110000   .119176598

KEB Ireland                       14         84      131000   .084734799
```

```
KEB Ireland                       14         84      131000   .084734799
                                  15         73      126000   .081500647
                                  12         70      137000   .088615783
                                  13         68      111000   .071798189
                                   7         56      130000   .084087969
                                   9         46      102000   .065976714
                                   7         41       92000   .059508409
                                  27         34      131000   .084734799
                                   6         28      107000   .069210867
                                  10         21      120000   .077619664
                                  19          7      118000   .076326003
                                  12          1      129000   .083441138
                                  21         48      112000   .072445019


33 rows selected.

VS_SQL>Clear Breaks;
breaks cleared
VS_SQL>
```

Listagg () + Within Group

21) List down all the prison names built by a builder with builder id 1 to 10.

```
Cl Scr
Select Builder_Id, Listagg(Prison_Name, ',') Within Group (Order By Prison_Name) As Prisons
From Prison
Where Builder_Id Between 1 And 10
Group By Builder_Id
Order By Builder_Id;
```

```
VS_SQL>Select Builder_Id, Listagg(Prison_Name, ',') Within Group (Order By Prison_Name) As Prisons
  2  From Prison
  3  Where Builder_Id Between 1 And 10
  4  Group By Builder_Id
  5  Order By Builder_Id;

BUILDER_ID
----------
PRISONS
--------------------------------------------------------------------------------------------------------
         1
Ballyclerahan Prison,Ballygar Prison,Coolaney Prison,Golden Prison

         2
Conna Prison,Delvin Prison,Dromiskin Prison

         4
Balreask Prison,Coolgreany Prison

         5
Ardagh Prison,Ballyconnell Prison,Carrigans Prison,Castlemartyr Prison

         6
Ballybay Prison,Brinlack Prison

         7
Ballycumber Prison,Brownstown Prison,Glasslough Prison,Greencastle Prison

         8
Bridge End Prison,Broadford Prison,Dunfanaghy Prison
```

```
         9
Clashmore Prison,Drimoleague Prison

        10
Ballytore Prison,Clonegal Prison,Courtown Harbour Prison,Garristown Prison,Graiguenamanagh Prison


9 rows selected.

VS_SQL>
```

List_Agg function gives you output with the attribute separated by a separator of your choice. It kind of concate the values of attribute based on grouped by attribute.

Cume_Dist() + Partition By + Order By

22) Display construction end year, name, gender and capacity of the prison. Also display the cumulative distribution of capacity based on construction end year.Filter the data to get details for prison for which construction ended on or after 2015. Order the output by end year than prison's capacity.

```
Cl scr
Select End_Year, Prison_Name, Gender, Prison_Capacity,
        Cume_Dist() Over(Partition By End_Year Order By Prison_Capacity) As CD_Capacity
From Prison
Where End_Year >= 2015
Order By End_Year, Prison_Capacity;
```

```
VS_SQL>Select End_Year, Prison_Name, Gender, Prison_Capacity,
  2     Cume_Dist() Over(Partition By End_Year Order By Prison_Capacity) As CD_Capacity
  3  From Prison
  4  Where End_Year >= 2015
  5  Order By End_Year, Prison_Capacity;

  END_YEAR PRISON_NAME              GENDER PRISON_CAPACITY CD_CAPACITY
---------- ------------------------ ------ --------------- -----------
      2015 Coolgreany Prison        Female             258  .333333333
      2015 Clonlara Prison          Mixed              304  .666666667
      2015 Blarney Prison           Mixed              332           1
      2016 Balreask Prison          Mixed              292          .5
      2016 Glencullen Prison        Male               297           1
      2018 Galway City Prison       Female             301           1
      2020 Carraroe Prison          Mixed              303           1
      2021 Ballydehob Prison        Male               275         .25
      2021 Cashel Prison            Male               316          .5
      2021 Emyvale Prison           Male               319         .75
      2021 Burtonport Prison        Male               356           1
      2022 Baltimore Prison         Male               359           1
      2023 Ardagh Prison            Female             335           1
      2024 Ballinlough Prison       Mixed              257  .166666667
      2024 Ballyshannon Prison      Female             276  .333333333
      2024 Courtown Harbour Prison  Male               303          .5
      2024 Belmullet Prison         Female             311  .666666667
      2024 Bray Prison              Mixed              336  .833333333
      2024 Gowran Prison            Female             373           1

19 rows selected.

VS_SQL>
```

Cume_Dist function returns cumulative distribution based on partitioned attribute. That is consider if you have 5 values 3, 5, 7, 8, 9. If you want cumulative distribution of these values it will be like: 0.2, 0.4, 0.6, 0.8, 1. It basically gives out a value that show how much proportion of records are equal to or less than that particular value.

Round () + Partition By

23) For the prisons built between 1993 and 2007 find out percentage of number of officers based on prison security level. Display prison name, security level, construction end year and number of officers in the prison. Sort the output by security level, than by end year and lastly by number of officers.

```
Cl scr
Select Prison_Name, Prison_Security, End_Year, Num_Officers,
        Round((Num_Officers/ Sum(Num_Officers) Over (Partition By Prison_Security)) * 100, 2) AS
Percentage_Of_Officers
From Prison
Where End_Year Between 1993 And 2001
Order By Prison_Security, End_Year, Prison_Capacity;
```

```
VS_SQL>Select Prison_Name, Prison_Security, End_Year, Num_Officers,
  2     Round((Num_Officers/ Sum(Num_Officers) Over (Partition By Prison_Security)) * 100, 2) AS Percentage_Of_Officers
  3  From Prison
  4  Where End_Year Between 1993 And 2001
  5  Order By Prison_Security, End_Year, Prison_Capacity;

PRISON_NAME              PRISON   END_YEAR NUM_OFFICERS PERCENTAGE_OF_OFFICERS
------------------------ ------ ---------- ------------ ----------------------
Inistioge Prison         High         1994          224                  13.33
Castlefin Prison         High         1995          202                  12.02
Drimoleague Prison       High         1996          202                  12.02
Buttevant Prison         High         1997          269                  16.01
Castleisland Prison      High         1998          202                  12.02
Coolaney Prison          High         2000          287                  17.08
Banagher Prison          High         2001          294                   17.5
Abbeyfeale Prison        Low          1993          251                  31.14
Ahascragh Prison         Low          1993          326                  40.45
Donegal Prison           Low          1998          229                  28.41
Ballyjamesduff Prison    Medium       1993          346                   13.8
Collon Prison            Medium       1994          358                  14.27
Garristown Prison        Medium       1996          187                   7.46
Ballysax Prison          Medium       1996          353                  14.07
Grangemore Prison        Medium       1997          312                  12.44
Bridge End Prison        Medium       1998          268                  10.69
Ballygar Prison          Medium       1999          324                  12.92
Bellanode Prison         Medium       2000          360                  14.35

18 rows selected.

VS_SQL>
```

Dense_Rank() + Order By

24) For builders who projected cost of more than 120000 find out how much percentage of extra costing they required for a single project and rank these builder according to there efficiency(less extra costing) in that particular project. Display Builder's name, prison's name, the projected cost they decarled for the project, actual cost required by the end of project, percentage of extra amount required and ranking based on extra amount required(Lowest to Highest). Sort these results by Percentage of extra costing required for a project.

```
Cl Scr
Select Name, Prison_Name, Projected_Cost, Actual_Cost, ((Actual_Cost  - Projected_Cost) /
Projected_Cost) * 100 As Extra_Cost,
        Dense_Rank() Over(Order By ((Actual_Cost  - Projected_Cost) / Projected_Cost) * 100) As
Efficiency
From Builder, (Select Prison.Builder_Id, Prison_Name, Projected_Cost, Actual_Cost
                From Prison) PrisonData
Where Builder.Builder_Id = PrisonData.Builder_Id
And Projected_Cost > 120000
Order By Extra_Cost;
```

```
VS_SQL>Select Name, Prison_Name, Projected_Cost, Actual_Cost, ((Actual_Cost  - Projected_Cost) / Projected_Cost) * 100 As Extra_Cost,

  2     Dense_Rank() Over(Order By ((Actual_Cost  - Projected_Cost) / Projected_Cost) * 100) As Efficiency
  3  From Builder, (Select Prison.Builder_Id, Prison_Name, Projected_Cost, Actual_Cost
  4              From Prison) PrisonData
  5  Where Builder.Builder_Id = PrisonData.Builder_Id
  6  And Projected_Cost > 120000
  7  Order By Extra_Cost;

NAME                                PRISON_NAME              PROJECTED_COST ACTUAL_COST EXTRA_COST EFFICIENCY
----------------------------------- ------------------------ -------------- ----------- ---------- ----------
John Fleming Construction           Greencastle Prison               127000      129000 1.57480315          1
McCabe Builders                     Cavan Prison                     126000      128000 1.58730159          2
Manor Park Homebuilders             Ballinroad Prison                124000      126000 1.61290323          3
Duggan Brothers (Contractors)       Drumlish Prison                  123000      126000 2.43902439          4
HBG Ascon                           Golden Prison                    125000      130000          4          5
Manor Park Homebuilders             Banagher Prison                  124000      129000 4.03225806          6
Park Developments (Dublin)          Garristown Prison                127000      133000 4.72440945          7
McCabe Builders                     Caherconlish Prison              127000      133000 4.72440945          7
Ballymore Properties                Dunfanaghy Prison                124000      131000 5.64516129          8
Manor Park Homebuilders             Burtonport Prison                128000      137000    7.03125          9
Kelland Homes                       Clonaslee Prison                 126000      135000 7.14285714         10
Michael McNamara And Co             Dromiskin Prison                 125000      135000          8         11
G And T Crampton Group              Ardagh Prison                    130000      141000 8.46153846         12
Coffey Construction                 Glenealy Prison                  128000      140000      9.375         13
Ballymore Properties                Broadford Prison                 127000      139000  9.4488189         14
HBG Ascon                           Coolaney Prison                  125000      138000       10.4         15
Allen And Smyth Construction        Belmullet Prison                 125000      140000         12         16

17 rows selected.

VS_SQL>
```

7. Youtube Links

Channel Link : https://www.youtube.com/@Msc_VS
Screencast 1 : https://youtu.be/n1a-JNb58H8
Screencast 2 : https://youtu.be/3jFFpVgmjF0