

|  |
| --- |
|  |
| Related image  PUB LEASE CONTRACTS  Relational Databases  Project Report  By Yash Sinojia [A00268852] |
| MASTER OF SCIENCE IN DATA ANALYTICS  2019 - 2020  FACULTY OF BUSINESS AND HOSPITALITY  SEMESTER 1 |

CONTENTS

[1. INTRODUCTION 3](#_Toc26891315)

[2. ORIGINAL PROJECT OUTLINE 4](#_Toc26891316)

[3. ENTITY RELATIONSHIP DIAGRAM 6](#_Toc26891317)

[4. BUILDING DATABASE 7](#_Toc26891318)

[4.1 Creating Tables 7](#_Toc26891319)

[4.2 Inserting Values 15](#_Toc26891320)

[5. TABLES 21](#_Toc26891321)

[6. QUERIES 24](#_Toc26891322)

[6.1 Queries involving Joins 24](#_Toc26891323)

[6.1.1 Self Join 24](#_Toc26891324)

[6.1.2 Outer Join 25](#_Toc26891325)

[6.2 Subqueries 26](#_Toc26891326)

[6.2.1 Using Comparison Operators 26](#_Toc26891327)

[6.2.2 Using Any and All 27](#_Toc26891328)

[6.2.3 Using Exists 28](#_Toc26891329)

[6.2.4 Multi-legged Subquery 29](#_Toc26891330)

[6.3 Update, Delete and Alter Queries 31](#_Toc26891331)

[6.3.1 Update Query 31](#_Toc26891332)

[6.3.2 Delete and Alter Queries 33](#_Toc26891333)

[6.4 Views 37](#_Toc26891334)

[6.4.1 With Check Option 37](#_Toc26891335)

[6.5 Grouping Queries 40](#_Toc26891336)

[6.5.1 Rollup with Grouping Sets and Union Alternatives 40](#_Toc26891337)

[6.5.2 Cube with Grouping Sets and Union Alternatives 44](#_Toc26891338)

[6.6 Partitioning Queries 49](#_Toc26891339)

[6.6.1 Aggregates with Running Total And Row Number 49](#_Toc26891340)

[6.6.2 Dense Rank 50](#_Toc26891341)

[6.6.3 Lead and Lag 52](#_Toc26891342)

[6.6.4 Windowing Aggregates 53](#_Toc26891343)

[6.6.5 First Value and Last Value 54](#_Toc26891344)

[6.7 Case Expressions 55](#_Toc26891345)

[6.7.1 Case Labels 55](#_Toc26891346)

[6.7.2 Labels Within Groups 56](#_Toc26891347)

[6.7.3 Pivoting into Multiple Rows 58](#_Toc26891348)

# INTRODUCTION

This project is an ambitious attempt to utilize all the knowledge assimilated from Relational Databases module, taught by Prof. Noel Tierney. A given case scenario of Pub Lease Contracts has been analyzed to be implemented in Oracle SQL. For that, the data has been populated according to column headers provided and with all the restrictions it imposed. This data was further refined according to the relationships among the tables. Those relationships were modelled on an Entity Relationship Diagram.

Then, this organized data was taken to the next level, to be implemented in Oracle SQL. For that, the headers and relations from ERD where converted into create table statements subjected with various integrity constraints. These tables were populated by data with insert statements.

Once, the tables are fully functional in Oracle SQL, various kinds of queries are designed. This section of implementation manages to summarize the concepts that have been covered in the module. Beyond that, it also explores various advanced functionalities of SQL beyond the syllabus. This report documents the author’s understanding of SQL and the author’s feasibility in adapting this knowledge for utilizing complex SQL functionalities.

This document does not demonstrate all the varieties of SQL queries. But instead explores the deeper conceptual aspects within 19 functionalities that this report describes as inventive query solutions for varied questions based on the given scenario.

# ASSUMPTIONS

It has been assumed that each Landlord leases a Pub at least for him to be actually a landlord.

It has been also assumed that each pub has a single distinct lease agreement, to be realistic. And hence Pub and Lease\_Agreement have a one-to-one relationship without any holes.

Other than these assumptions, no other ambiguity, inconsistency or flaw in the project scenario has been triggered.

# ORIGINAL PROJECT OUTLINE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 21 | Yash | Sinojia | Pub Lease Contracts | PLC |

Historically pubs were owner operated. However, leasing pubs has become very common in recent years. Some years ago, the Vintners association commissioned a project (that remains unfinished) to design a database application that would capture details of pub leasing and the level of activity in the sector. Key details captured by systems analysts include:

* A landlord may own many pubs and each pub is owned by a single landlord
* A publican may participate in many lease agreements and a lease agreement involves a single publican
* A letting agent (auctioneer) may arrange many lease agreements and a lease agreement involves a single letting agent
* A solicitor may draw up many lease agreements and a lease agreement is drawn up by a single solicitor

----- ORIGINAL -----

Landlord (Landlord\_Id, FName, Sname, Address, Town, County, Tel\_No, Email, Specialism (*Large Pubs, Medium Sized Pubs, Small Pubs*)

Pub (Pub\_Id, Name, Address, City, Country, Tel\_No, Email, Capacity (i.e. how many people can it hold), Year\_Built, Sq\_Metres, Customer\_Profile (*Students, Professionals, Sports Fans, Pensioners*), Current\_Value, Specialism (*Food, Music etc*), Location (*Poor, Good, Excellent*), Landlord\_Id)

Publican (Publican\_Id, Fname, Sname, Address, County, Gender, Date\_Of\_Birth, Mobile\_No, Email\_Address, Qualifications (*Bar Manager, Chef, Bar Supervisor*), Specialism (*Music, Food, Sport, Parties),*

Letting\_Agent (Letting\_Agent\_Id, Name, City, County, Specialism (*Long Term Contracts, Medium Term Leases, Long Term Agreements),* Web\_Site, Email, Year\_Founded, Annual\_Turnover, Reputation (*Fair, Good, Excellent*)

Lease\_Agreement (Lease\_Agreement\_Id, Pub\_Id, Publican\_Id, Start\_Date, End\_Date, Deposit, Instalment\_Frequency (*Monthly, Quarterly, Annually*), Payment\_Method (*Cheque, Direct Debit, Standing Order*), Instalment\_Amount, Letting\_Agent\_Id, Solicitor\_Id)

Solicitor (Solicitor\_Id, Fname, Sname, Address, County, Gender, Date\_Of\_Birth, Mobile\_No, Email\_Address, Qualifications)

----- MODIFIED -----

Landlord Landlord\_Id, FName, Sname, Address, Town, County, Tel\_No, Email, Specialism

Pub Pub\_Id, Name, Address, City, Country, Tel\_No, Email, Capacity, Year\_Built, Sq\_Metres, Customer\_Profile, Current\_Value, Specialism, Location, Landlord\_Id

Publican Publican\_Id, Fname, Sname, Address, County, Gender, Date\_Of\_Birth, Mobile\_No, Email\_Address, Qualifications, Specialism

Letting\_Agent Letting\_Agent\_Id, Name, City, County, Specialism*,* Web\_Site, Email, Year\_Founded, Annual\_Turnover, Reputation

Solicitor Solicitor\_Id, Fname, Sname, Address, County, Gender, Date\_Of\_Birth, Mobile\_No, Email\_Address, Qualifications

Lease\_Agreement Lease\_Agreement\_Id, Pub\_Id, Publican\_Id, Start\_Date, End\_Date, Deposit, Instalment\_Frequency, Payment\_Method, Installment\_Amount, Letting\_Agent\_Id, Solicitor\_Id

Lease\_Agreement

Lease\_Agreement\_Id

Pub\_Id

Publican\_Id

Start\_Date

End\_Date

Deposit

Instalment\_Frequency

Payment\_Method

Installment\_Amount

Letting\_Agent\_Id

Solicitor\_Id

Solicitor

Solicitor\_Id

Fname

Sname

Address

County

Gender

Date\_Of\_Birth

Mobile\_No

Email\_Address

Qualifications

Letting\_Agent

Letting\_Agent\_Id

Name

City

County

Specialism

Web\_Site

Email

Year\_Founded

Annual\_Turnover

Reputation

Publican

Publican\_Id

Fname

Sname

Address

County

Gender

Date\_Of\_Birth

Mobile\_No

Email\_Address

Qualifications

Specialism

Landlord

Landlord\_Id

FName

Sname

Address

Town

County

Tel\_No

Email

Specialism

Pub

Pub\_Id

Name

Address

City

Country

Tel\_No

Email

Capacity

Year\_Built

Sq\_Metres

Customer\_Profile

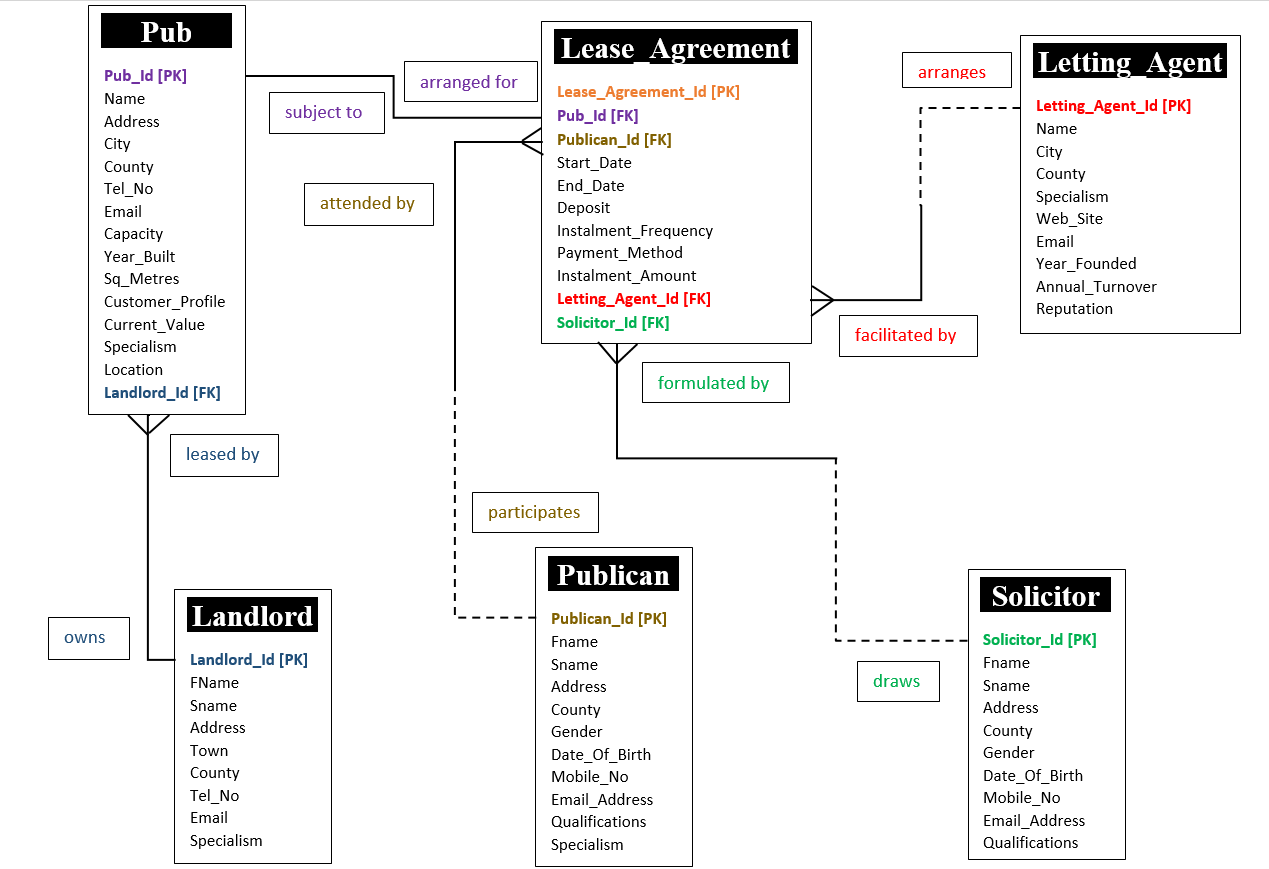
Current\_Value

Specialism

Location

Landlord\_Id

# ENTITY RELATIONSHIP DIAGRAM



# BUILDING DATABASE

## **4.1 Creating Tables**

host echo Building Oracle Pub\_Lease\_Contracts tables. Please wait.

Clear Screen

Set Termout On

Set Echo On

Set Feedback On

Set Verify On

Set Heading on

Set Pagesize 100

Set Linesize 180

Drop Table Lease\_Agreement;

Drop Table Solicitor;

Drop Table Letting\_Agent;

Drop Table Publican;

Drop Table Pub;

Drop Table Landlord;

/\* Table Name: Landlord \*/

Drop Table Landlord;

Create Table Landlord (

Landlord\_Id Number(2),

Fname Varchar2(8),

Sname Varchar2(10) Constraint Landlord\_Sname\_Nn Not Null,

Address Varchar2(19),

Town Varchar2(11),

County Varchar2(9),

Tel\_No Varchar2(6),

Email Varchar2(18),

Specialism Varchar2(11),

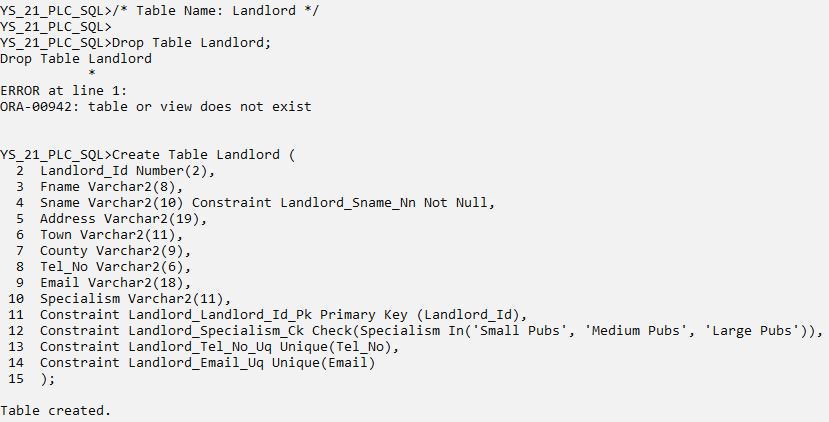
Constraint Landlord\_Landlord\_Id\_Pk Primary Key (Landlord\_Id),

Constraint Landlord\_Specialism\_Ck Check(Specialism In('Small Pubs', 'Medium Pubs', 'Large Pubs')),

Constraint Landlord\_Tel\_No\_Uq Unique(Tel\_No),

Constraint Landlord\_Email\_Uq Unique(Email)

);



/\* Table Name: Pub \*/

Drop Table Pub;

Create Table Pub(

Pub\_Id Number(3),

Name Varchar2(19) Constraint Pub\_Name\_Nn Not Null,

Address Varchar2(21),

City Varchar2(11),

County Varchar2(9),

Tel\_No Number(6),

Email Varchar2(22),

Capacity Number(4),

Year\_Built Number(4),

Sq\_Meters Number(5),

Customer\_Profile Varchar2(13),

Current\_Value Number(7),

Specialism Varchar2(8),

Location Varchar2(9),

Landlord\_Id Number(2),

Constraint Pub\_Pub\_Id\_Pk Primary Key(Pub\_Id),

Constraint Pub\_Customer\_Profile\_Ck Check(Customer\_Profile In('Students', 'Professionals', 'Sports Fans', 'Pensioners')),

Constraint Pub\_Specialism\_Ck Check(Specialism In('Food', 'Music', 'Sports', 'Ambience','Parties')),

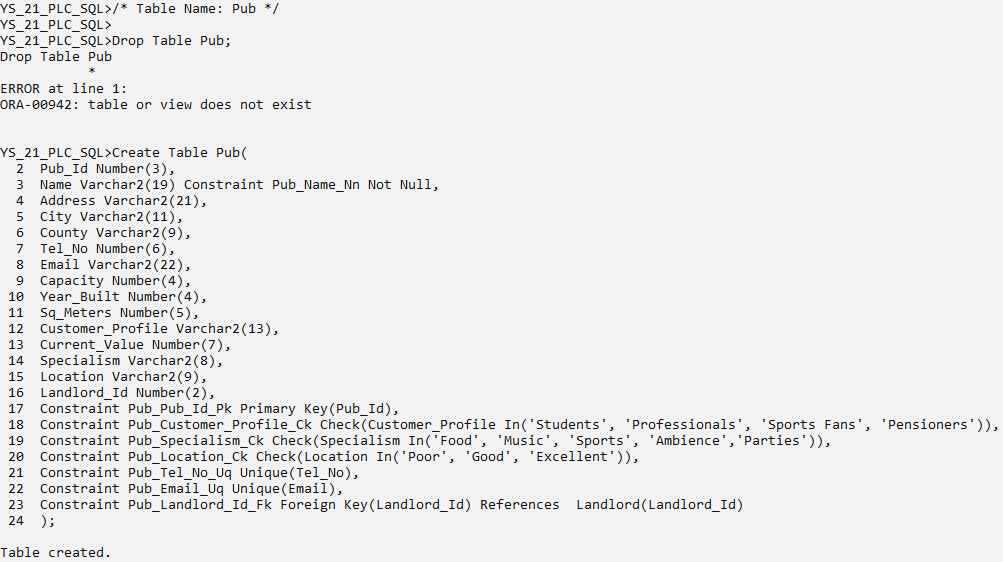
Constraint Pub\_Location\_Ck Check(Location In('Poor', 'Good', 'Excellent')),

Constraint Pub\_Tel\_No\_Uq Unique(Tel\_No),

Constraint Pub\_Email\_Uq Unique(Email),

Constraint Pub\_Landlord\_Id\_Fk Foreign Key(Landlord\_Id) References Landlord(Landlord\_Id)

);



/\* Table Name: Publican \*/

Drop Table Publican;

Create Table Publican(

Publican\_Id Number(3),

Fname Varchar2(7),

Sname Varchar2(11) Constraint Publican\_Sname\_Nn Not Null,

Address Varchar2(22),

County Varchar2(9),

Gender Varchar2(1),

Date\_of\_Birth Date,

Mobile\_No Number(6),

Email\_Address Varchar2(18),

Qualifications Varchar2(14),

Specialism Varchar2(7),

Constraint Publican\_Publican\_Id\_Pk Primary Key(Publican\_Id),

Constraint Publican\_Qualifications\_Ck Check(Qualifications in('Bar Manager', 'Chef', 'Bar Supervisor')),

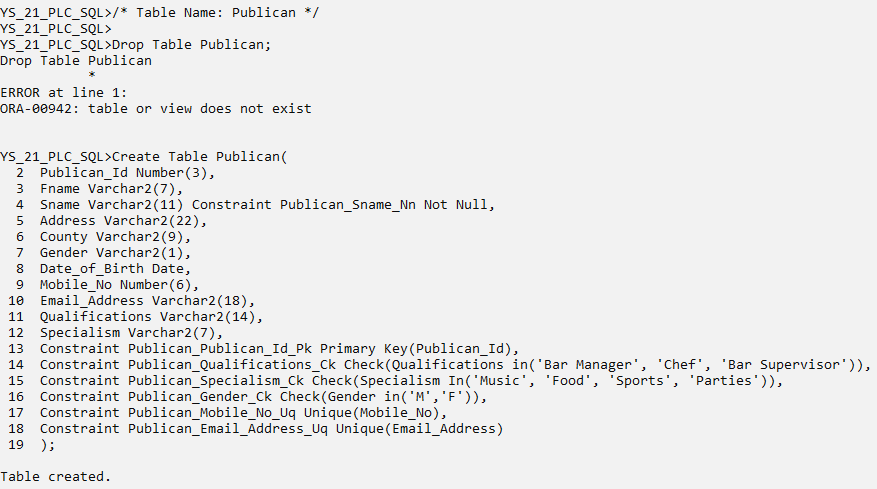
Constraint Publican\_Specialism\_Ck Check(Specialism In('Music', 'Food', 'Sports', 'Parties')),

Constraint Publican\_Gender\_Ck Check(Gender in('M','F')),

Constraint Publican\_Mobile\_No\_Uq Unique(Mobile\_No),

Constraint Publican\_Email\_Address\_Uq Unique(Email\_Address)

);



/\* Table Name: Letting Agent \*/

Drop Table Letting\_Agent;

Create Table Letting\_Agent(

Letting\_Agent\_Id Number(3),

Name Varchar2(14) Constraint Letting\_Agent\_Name\_Nn Not Null,

City Varchar2(11),

County Varchar2(9),

Specialism Varchar2(20),

Web\_Site Varchar2(26),

Email Varchar2(26),

Year\_Founded Number(4),

Annual\_Turnover Number(5),

Reputation Varchar2(9),

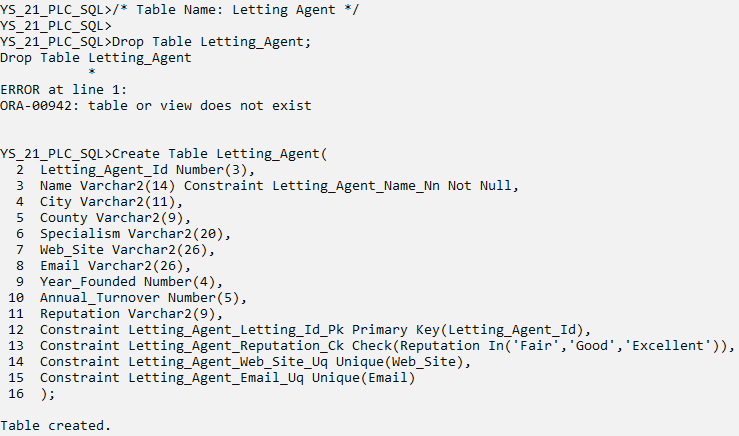
Constraint Letting\_Agent\_Letting\_Id\_Pk Primary Key(Letting\_Agent\_Id),

Constraint Letting\_Agent\_Reputation\_Ck Check(Reputation In('Fair','Good','Excellent')),

Constraint Letting\_Agent\_Web\_Site\_Uq Unique(Web\_Site),

Constraint Letting\_Agent\_Email\_Uq Unique(Email)

);



/\* Table Name: Solicitor \*/

Drop Table Solicitor;

Create Table Solicitor(

Solicitor\_Id Number(3),

Fname Varchar2(6),

Sname Varchar2(8) Constraint Solicitor\_Sname\_Nn Not Null,

Address Varchar2(17),

County Varchar2(9),

Gender Varchar2(1),

Date\_of\_Birth Date,

Mobile\_No Number(6),

Email\_Address Varchar2(17),

Qualifications Varchar2(8),

Constraint Solicitor\_Solicitor\_Id\_Pk Primary Key(Solicitor\_Id),

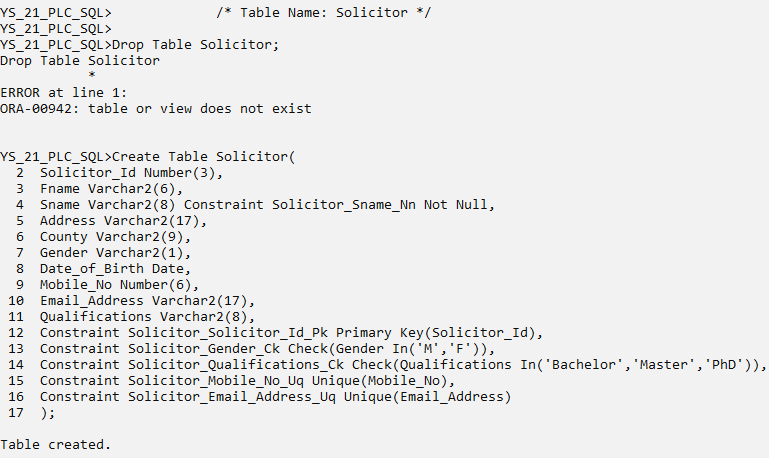
Constraint Solicitor\_Gender\_Ck Check(Gender In('M','F')),

Constraint Solicitor\_Qualifications\_Ck Check(Qualifications In('Bachelor','Master','PhD')),

Constraint Solicitor\_Mobile\_No\_Uq Unique(Mobile\_No),

Constraint Solicitor\_Email\_Address\_Uq Unique(Email\_Address)

);



/\* Table Name: Lease Agreement \*/

Drop Table Lease\_Agreement;

Create Table Lease\_Agreement(

Lease\_Agreement\_Id Number(4),

Pub\_Id Number(3),

Publican\_Id Number(3),

Start\_Date Date,

End\_Date Date,

Deposit Number(5),

Installment\_frequency Varchar2(9),

Payment\_Method Varchar2(14),

Installment\_Amount Number(6),

Letting\_Agent\_Id Number(3),

Solicitor\_Id Number(3),

Constraint Lease\_Agreement\_Lease\_Id\_Pk Primary Key(Lease\_Agreement\_Id),

Constraint Lease\_Agreement\_Inst\_freq\_Ck Check(Installment\_frequency In('Annually','Quarterly','Monthly')),

Constraint Lease\_Agreement\_Pay\_Method\_Ck Check(Payment\_Method In('Cheque','Direct Debit','Standing Order')),

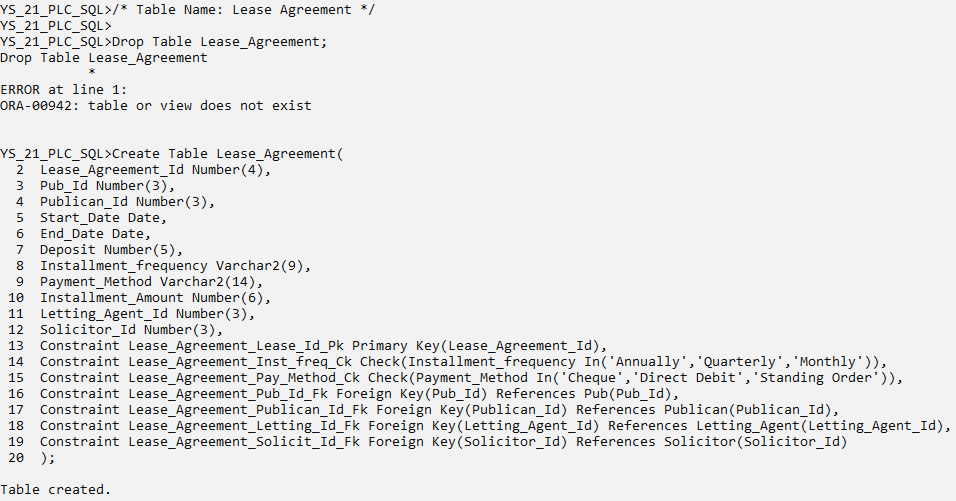
Constraint Lease\_Agreement\_Pub\_Id\_Fk Foreign Key(Pub\_Id) References Pub(Pub\_Id),

Constraint Lease\_Agreement\_Publican\_Id\_Fk Foreign Key(Publican\_Id) References Publican(Publican\_Id),

Constraint Lease\_Agreement\_Letting\_Id\_Fk Foreign Key(Letting\_Agent\_Id) References Letting\_Agent(Letting\_Agent\_Id),

Constraint Lease\_Agreement\_Solicit\_Id\_Fk Foreign Key(Solicitor\_Id) References Solicitor(Solicitor\_Id)

);



## **4.2 Inserting Values**

/\* Table Name: Landlord \*/

/\*Landlord\_Id,Fname,Sname,Address,Town,County,Tel\_No,Email,Specialism \*/

Insert Into Landlord Values (1,'Boris','Shcherbina','4721 Quisque Avenue','Drogheda','Louth',900001,'BorisShcherbina@ie','Medium Pubs');

Insert Into Landlord Values (2,'Mary','Shelley','3854 Ut Street','Dingle','Kerry',900002,'MaryShelley@ie','Small Pubs');

Insert Into Landlord Values (3,'John','Hudson','3854 Ut Street','Athlone','Westmeath',900003,'JohnHudson@ie','Medium Pubs');

Insert Into Landlord Values (4,'Fanny','Brawne','2790 At St.','Dublin','Dublin',900004,'FannyBrawne@ie','Large Pubs');

Insert Into Landlord Values (5,'Shauna','O Brien','80 Nec Rd.','Cork','Cork',900005,'ShaunaO Brien@ie','Large Pubs');

Insert Into Landlord Values (6,'Peter','O Connor','503 Pede, Av.','Athlone','Westmeath',900006,'PeterO Connor@ie','Medium Pubs');

Insert Into Landlord Values (7,'Jack','Williams','19 Quisque Rd.','Limerick','Limerick',900007,'JackWilliams@ie','Large Pubs');

Insert Into Landlord Values (8,'Kevin','Limey','9 Consectetuer Rd.','Cork','Cork',900008,'KevinLimey@ie','Large Pubs');

Insert Into Landlord Values (9,'Sandra','Cox','114 Dolor, Ave','Dublin','Dublin',900009,'SandraCox@ie','Large Pubs');

Insert Into Landlord Values (10,'David','Morrison','3 Interdum Rd.','Dublin','Dublin',900010,'DavidMorrison@ie','Small Pubs');

Insert Into Landlord Values (11,'Hillary','Swans','8681 A, Ave','Letterkenny','Donegal',900011,'HillarySwans@ie','Small Pubs');

Insert Into Landlord Values (12,'Patricia','Burns','62 Egestas St.','Sligo','Sligo',900012,'PatriciaBurns@ie','Small Pubs');

Insert Into Landlord Values (13,'Rachael','Brown','7 Morbi Avenue','Waterford','Waterford',900013,'RachaelBrown@ie','Medium Pubs');

commit;

/\* Table Name: Pub \*/

/\*Pub\_Id,Name,Address,City,County,Tel\_No,Email,Capacity,Year\_Built,Sq\_Meters,Customer\_Profile,Current\_Value,Specialism,Location,Landlord\_Id \*/

Insert Into Pub Values (101,'Charlies','79 Ante Ave','Athlone','Westmeath',400001,'Charlies@ie',500,1990,5500,'Students',450000,'Music','Good',1);

Insert Into Pub Values (102,'Tack Room','9 Sapien. St.','Athlone','Westmeath',400002,'Tack Room@ie',400,1970,6000,'Students',680000,'Music','Good',2);

Insert Into Pub Values (103,'Seans Bar','604 Ut, Rd.','Athlone','Westmeath',400003,'Seans Bar@ie',750,200,7500,'Professionals',5000000,'Ambience','Excellent',3);

Insert Into Pub Values (104,'Piano Bar','431 Mi Av.','Dublin','Dublin',400004,'Piano Bar@ie',1000,1850,9950,'Professionals',750000,'Music',NULL,4);

Insert Into Pub Values (105,'Granary Gastro Pub','563 Turpis St.','Dundalk','Louth',400005,'Granary Gastro Pub@ie',250,1875,3500,'Pensioners',210000,'Food','Poor',5);

Insert Into Pub Values (106,'Bacchus','3930 In Avenue','Ballymahon','Longford',400006,'Bacchus@ie',300,1698,3750,'Pensioners',175000,'Food','Excellent',6);

Insert Into Pub Values (107,'Murphys Law','2260 Quisque Ave','Cork','Cork',400007,'Murphys Law@ie',150,2001,1800,'Sports Fans',475000,'Sports',NULL,7);

Insert Into Pub Values (108,'Gertie Browne','6434 Adipiscing Rd.','Limerick','Limerick',400008,'Gertie Browne@ie',950,1997,10000,'Students',345000,'Parties','Poor',8);

Insert Into Pub Values (109,'Careys Tavern','1628 Luctus. St.','Sligo','Sligo',400009,'Careys Tavern@ie',600,2010,6000,'Sports Fans',310000,'Sports','Good',9);

Insert Into Pub Values (110,'Shack Pub','9139 Erat Rd.','Waterford','Waterford',400010,'Shack Pub@ie',1250,1995,12500,'Sports Fans',490000,'Sports',NULL,10);

Insert Into Pub Values (111,'Snug Bar','3495 Massa. Rd.','Dublin','Dublin',400011,'Snug Bar@ie',1500,1960,13500,'Professionals',680000,'Ambience','Good',11);

Insert Into Pub Values (112,'Dark Horse','7897 Sollicitudin Rd.','Cork','Cork',400012,'Dark Horse@ie',1250,1950,10500,'Students',715000,'Parties','Excellent',12);

Insert Into Pub Values (113,'Castle Inn','9576 Eu Rd.','Limerick','Limerick',400013,'Castle Inn@ie',750,1750,7950,'Pensioners',500000,'Food','Good',13);

Insert Into Pub Values (114,'Flannerys Bar','6508 Litora St.','Dingle','Kerry',400014,'Flannerys Bar@ie',1150,1850,11250,'Pensioners',290000,'Food','Good',1);

Insert Into Pub Values (115,'Malt House','9327 Magna. Av.','Dublin','Dublin',400015,'Malt House@ie',1750,1999,17500,'Students',650000,'Parties','Poor',2);

Insert Into Pub Values (116,'Nuts Corner','2432 Id, St.','Waterford','Waterford',400016,'Nuts Corner@ie',950,2005,10000,'Students',450000,'Music',NULL,3);

Insert Into Pub Values (117,'Fiddlers','46 Et St.','Drogheda','Louth',400017,'Fiddlers@ie',400,1940,5950,'Sports Fans',175000,'Sports','Poor',4);

Insert Into Pub Values (118,'Three Jolly Pigeons','1380 Velit St.','Letterkenny','Donegal',400018,'Three Jolly Pigeons@ie',250,1963,2100,'Pensioners',375000,'Music',NULL,5);

Insert Into Pub Values (119,'Seerys Bar','3080 Magna. Rd.','Cork','Cork',400019,'Seerys Bar@ie',550,2007,5700,'Professionals',680000,'Food','Good',2);

Insert Into Pub Values (120,'Temple Bar','5586 Eu, Rd.','Dublin','Dublin',400020,'Temple Bar@ie',750,1993,8100,'Professionals',4800000,'Ambience',NULL,1);

commit;

/\* Table Name: Publican \*/

/\*Publican\_Id,Fname,Sname,Address,County,Gender,Date\_of\_Birth,Mobile\_No,Email\_Address,Qualifications,Specialism \*/

Insert Into Publican Values (200,'Jim','Gallows','502 Ac Rd.','Louth','M','18-Aug-74',500001,'JimGallows@ie','Bar Manager','Music');

Insert Into Publican Values (210,'Jerry','Ryan','94 Enim. Av.','Kerry','F','02-Feb-67',500002,'JerryRyan@ie','Bar Manager','Food');

Insert Into Publican Values (220,'Jane','Campion','7804 Phasellus Rd.','Westmeath','F','05-Mar-52',500003,'JaneCampion@ie','Bar Manager','Sports');

Insert Into Publican Values (230,'Rachael','Maurier','7347 Et, St.','Dublin','F','03-Nov-60',500004,'RachaelMaurier@ie','Chef','Parties');

Insert Into Publican Values (240,'Sophie','Turner','7379 Vulputate Ave','Waterford','F','24-Sep-74',500005,'SophieTurner@ie','Bar Supervisor','Music');

Insert Into Publican Values (250,'Rebecca','Jones','8956 Est, St.','Limerick','F','16-Dec-77',500006,'RebeccaJones@ie','Bar Supervisor','Food');

Insert Into Publican Values (260,'Donald','Glover','5315 Nam Av.','Cork','M','22-Apr-55',500007,'DonaldGlover@ie','Chef','Sports');

Insert Into Publican Values (270,'Shannan','O Reilly','5919 Ac Rd.','Galway','F','28-Oct-51',500008,'ShannanO Reilly@ie','Bar Manager','Parties');

Insert Into Publican Values (280,'Max','Rocketowsky','6434 Mauris Rd.','Dublin','M','12-Apr-60',500009,'MaxRocketowsky@ie','Bar Supervisor','Music');

Insert Into Publican Values (290,'Tristan','Starr','772 Amet Street','Westmeath','M','09-Apr-78',500010,'TristanStarr@ie','Bar Supervisor','Food');

Insert Into Publican Values (300,'Jaime','Murray','39 Feugiat Road','Sligo','F','04-Jan-52',500011,'JaimeMurray@ie','Bar Manager','Sports');

Insert Into Publican Values (310,'Barry','Sands','3716 Adipiscing Street','Dublin','M','09-Oct-67',500012,'BarrySands@ie','Chef','Parties');

commit;

/\* Table Name: Letting\_Agent \*/

/\*Letting\_Agent\_Id,Name,City,County,Specialism,Web\_Site,Email,Year\_Founded,Annual\_Turnover,Reputation \*/

Insert Into Letting\_Agent Values (501,'Gore Vidal','Galway','Galway','Long Term Contracts','gorevidalgalway.ie','gorevidalgalway@ie',1968,35000,'Good');

Insert Into Letting\_Agent Values (502,'James Hatfield','Dingle','Kerry','Medium Term Leases','jameshatfielddingle.ie','jameshatfielddingle@ie',1971,21500,'Good');

Insert Into Letting\_Agent Values (503,'Chris Conner','Athlone','Westmeath','Medium Term Leases','chrisconnerathlone.ie','chrisconnerathlone@ie',1991,19500,'Fair');

Insert Into Letting\_Agent Values (504,'Teresa Palmer','Letterkenny','Donegal','Long Term Agreements','teresapalmerletterkenny.ie','teresapalmerletterkenny@ie',1986,39500,'Excellent');

Insert Into Letting\_Agent Values (505,'Katie Holmes','Dublin','Dublin','Medium Term Leases','katieholmesdublin.ie','katieholmesdublin@ie',1999,41500,'Good');

Insert Into Letting\_Agent Values (506,'Jessica Rogers','Limerick','Limerick','Long Term Agreements','jessicarogerslimerick.ie','jessicarogerslimerick@ie',2001,27500,'Fair');

Insert Into Letting\_Agent Values (507,'Gary Limey','Dundalk','Louth','Long Term Contracts','garylimeydundalk.ie','garylimeydundalk@ie',1973,35000,'Good');

Insert Into Letting\_Agent Values (508,'Jacob Bloggs','Dublin','Dublin','Long Term Agreements','jacobbloggsdublin.ie','jacobbloggsdublin@ie',1969,52500,'Good');

Insert Into Letting\_Agent Values (509,'Rick Sanchez','Cork','Cork','Medium Term Leases','ricksanchezcork.ie','ricksanchezcork@ie',1986,47500,'Excellent');

Insert Into Letting\_Agent Values (510,'Erica Yong','Dublin','Dublin','Medium Term Leases','ericayongdublin.ie','ericayongdublin@ie',1989,24000,'Fair');

Insert Into Letting\_Agent Values (511,'Louis Sixx','Waterford','Waterford','Long Term Contracts','louissixxwaterford.ie','louissixxwaterford@ie',1995,40000,'Good');

Insert Into Letting\_Agent Values (512,'Felicia Day','Galway','Galway','Medium Term Leases','feliciadaygalway.ie','feliciadaygalway@ie',2000,50000,'Excellent');

Insert Into Letting\_Agent Values (513,'Salmon Leads','Cork','Cork','Long Term Contracts','salmonleadscork.ie','salmonleadscork@ie',2003,32500,'Fair');

commit;

/\* Table Name: Solicitor \*/

/\*Solicitor\_Id,Fname,Sname,Address,County,Gender,Date\_of\_Birth,Mobile\_No,Email\_Address,Qualifications \*/

Insert Into Solicitor Values (701,'Ashley','Horns','48 Vestibulum Rd.','Westmeath','F','09-Oct-67',700001,'AshleyHorns@ie','Master');

Insert Into Solicitor Values (702,'Benny','Lane','51 Semper Rd.','Limerick','M','11-Mar-58',700002,'BennyLane@ie','PhD');

Insert Into Solicitor Values (703,'Frank','Murphy','241 Amet St.','Dublin','M','19-May-57',700003,'FrankMurphy@ie','PhD');

Insert Into Solicitor Values (704,'Janice','Pierce','2 At Rd.','Galway','F','26-Feb-72',700004,'JanicePierce@ie','Master');

Insert Into Solicitor Values (705,'Ian','Freeman','74 Interdum. Road','Dublin','M','31-Dec-66',700005,'IanFreeman@ie','Master');

Insert Into Solicitor Values (706,'Patsy','Vance','99 Neque Av.','Dublin','F','29-Aug-71',700006,'PatsyVance@ie','Master');

Insert Into Solicitor Values (707,'Elise','Nichols','95 Semper, Road','Cork','F','31-Mar-80',700007,'EliseNichols@ie','Bachelor');

Insert Into Solicitor Values (708,'Simone','Miramar','33 Nulla. Road','Louth','F','24-Jun-57',700008,'SimoneMiramar@ie','PhD');

Insert Into Solicitor Values (709,'Quincy','Kent','15 Augue Avenue','Waterford','M','11-May-74',700009,'QuincyKent@ie','Master');

Insert Into Solicitor Values (710,'Darren','Goldberg','8 Risus. St.','Kerry','M','14-Feb-67',700010,'DarrenGoldberg@ie','PhD');

Insert Into Solicitor Values (711,'Warren','Edwards','150 Magna. Rd.','Cork','M','04-Nov-78',700011,'WarrenEdwards@ie','Bachelor');

Insert Into Solicitor Values (712,'Gloria','Luan','38 Erat Av.','Limerick','F','10-Jul-68',700012,'GloriaLuan@ie','Master');

commit;

/\* Table Name: Lease\_Agreement \*/

/\*Lease\_Agreement\_Id,Pub\_Id,Publican\_Id,Start\_Date,End\_Date,Deposit,Installment\_frequency,Payment\_Method,Installment\_Amount,Letting\_Agent\_Id,Solicitor\_Id\*/

Insert Into Lease\_Agreement Values (1000,101,200,'20-Jan-79','03-Dec-00',5000,'Monthly','Cheque',20000,501,701);

Insert Into Lease\_Agreement Values (1010,102,210,'05-Jan-88','27-Jun-14',7500,'Quarterly','Cheque',50000,502,702);

Insert Into Lease\_Agreement Values (1020,103,220,'10-Oct-92','30-Mar-08',9550,'Annually','Standing Order',200000,503,703);

Insert Into Lease\_Agreement Values (1030,104,230,'24-Aug-85','03-Sep-12',9950,'Annually','Direct Debit',225000,504,704);

Insert Into Lease\_Agreement Values (1040,105,240,'29-Jun-97','20-Aug-09',5000,'Monthly','Cheque',15500,505,705);

Insert Into Lease\_Agreement Values (1050,106,250,'13-Feb-81','07-Aug-24',7500,'Quarterly','Direct Debit',65000,506,706);

Insert Into Lease\_Agreement Values (1060,107,260,'10-Aug-95','30-Oct-17',10000,'Annually','Standing Order',177500,507,707);

Insert Into Lease\_Agreement Values (1070,108,270,'08-Sep-86','11-Dec-00',12500,'Annually','Direct Debit',159000,501,708);

Insert Into Lease\_Agreement Values (1080,109,210,'11-Feb-96','26-Jan-19',2500,'Monthly','Cheque',12500,502,703);

Insert Into Lease\_Agreement Values (1090,110,220,'15-Jul-90','11-Feb-13',1570,'Monthly','Direct Debit',17750,503,704);

Insert Into Lease\_Agreement Values (1100,111,230,'13-Jan-87','23-Dec-10',2750,'Monthly','Direct Debit',18000,504,705);

Insert Into Lease\_Agreement Values (1110,112,240,'23-Sep-99','08-Jul-15',2500,'Monthly','Cheque',22500,505,706);

Insert Into Lease\_Agreement Values (1120,113,200,'18-Oct-87','20-Oct-18',3570,'Monthly','Direct Debit',27000,506,701);

Insert Into Lease\_Agreement Values (1130,114,200,'16-Aug-75','08-Jan-08',3500,'Monthly','Direct Debit',11500,501,701);

Insert Into Lease\_Agreement Values (1140,115,200,'07-Dec-92','25-Feb-01',7500,'Quarterly','Cheque',57000,501,701);

Insert Into Lease\_Agreement Values (1150,116,210,'05-Aug-84','30-Jun-05',8570,'Quarterly','Standing Order',55000,502,702);

Insert Into Lease\_Agreement Values (1160,117,210,'20-Oct-83','31-Aug-13',6500,'Monthly','Direct Debit',16000,503,702);

Insert Into Lease\_Agreement Values (1170,118,220,'09-Dec-79','20-Aug-00',8000,'Quarterly','Cheque',45000,504,703);

Insert Into Lease\_Agreement Values (1180,119,230,'09-Sep-91','29-Oct-05',2950,'Monthly','Direct Debit',21000,505,704);

Insert Into Lease\_Agreement Values (1190,120,240,'21-Nov-80','17-Jun-21',3150,'Monthly','Direct Debit',23500,506,705);

commit;

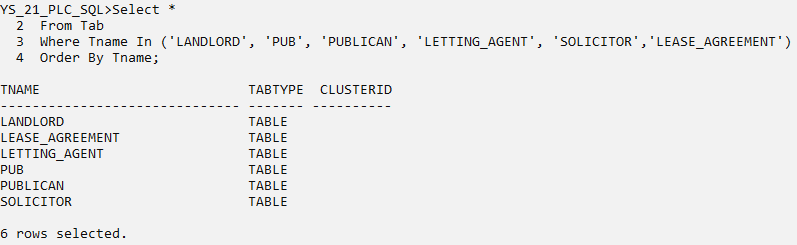
# TABLES

Select \*

From Tab

Where Tname In ('LANDLORD', 'PUB', 'PUBLICAN', 'LETTING\_AGENT', 'SOLICITOR','LEASE\_AGREEMENT')

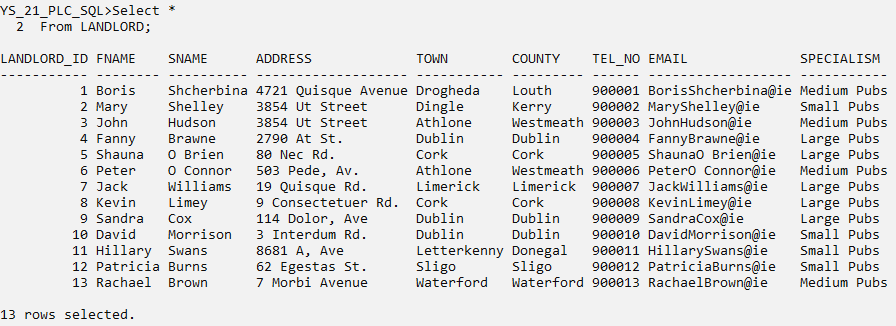
Order By Tname;



* Landlord

Select \*

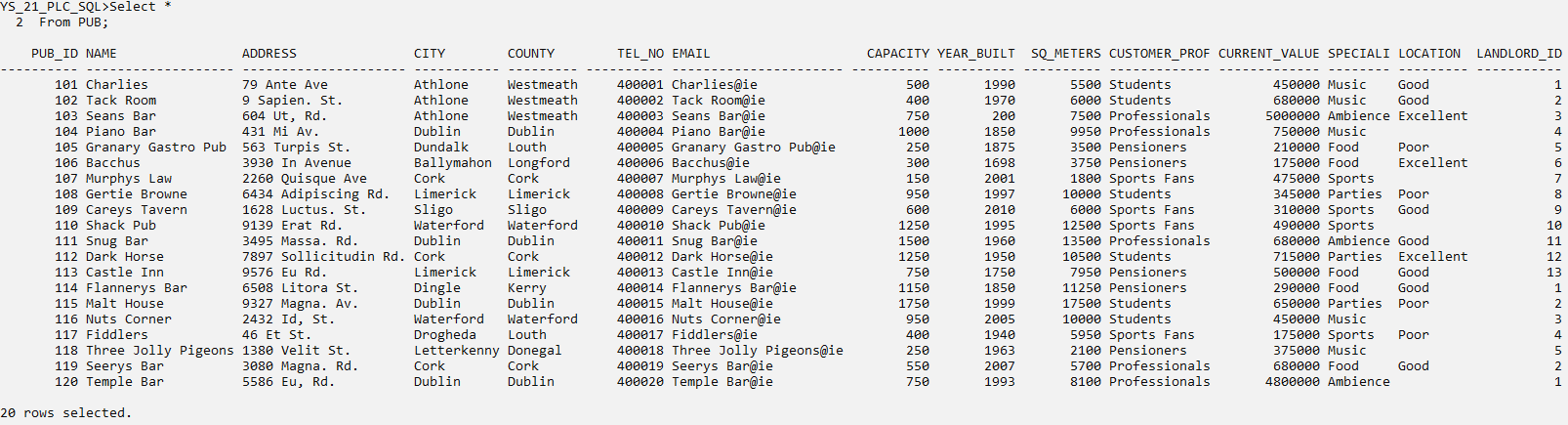
From LANDLORD;



* Pub

Select \*

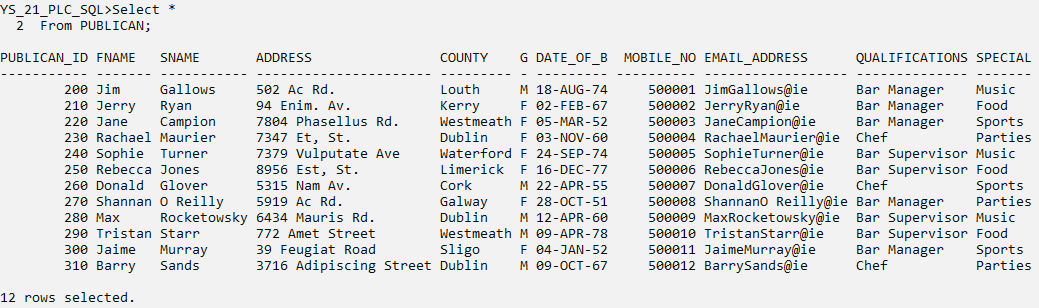
From PUB;



* Publican

Select \*

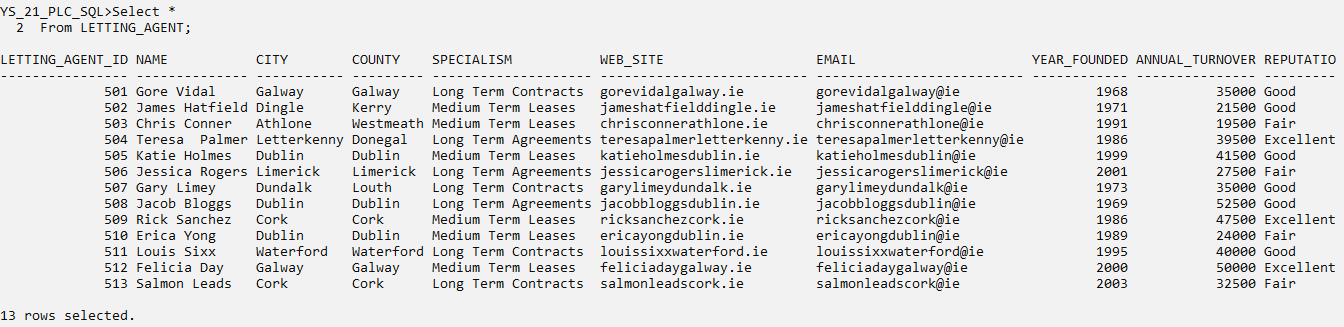
From PUBLICAN;



* Letting Agent

Select \*

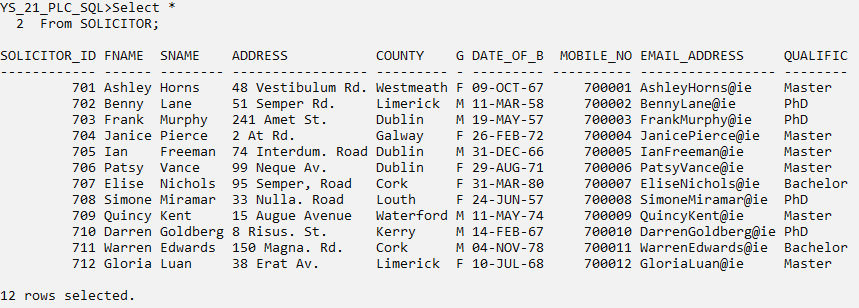
From LETTING\_AGENT;



* Solicitor

Select \*

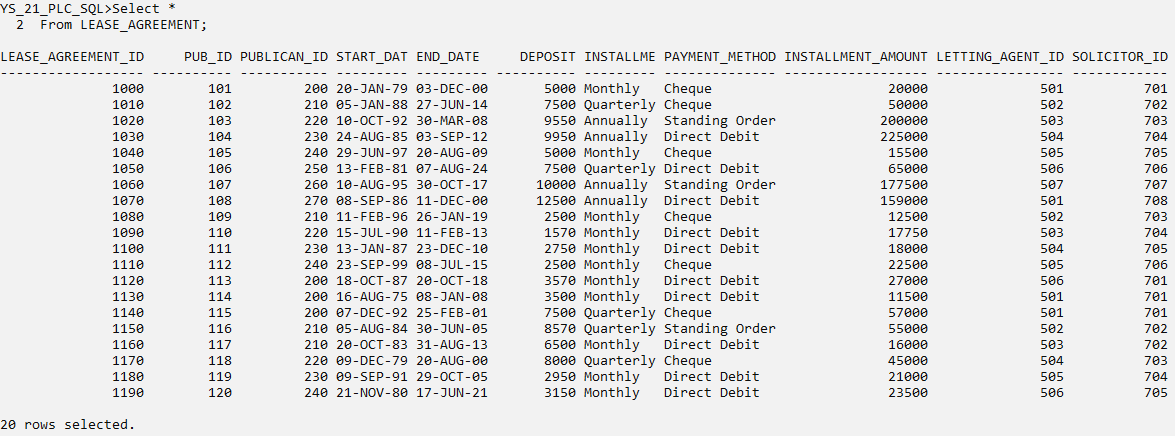
From SOLICITOR;



* Lease Agreement

Select \*

From LEASE\_AGREEMENT;



# QUERIES

## **6.1 Queries involving Joins**

### 6.1.1 Self Join

Question.1

Give the Id and Deposit of pairs of Lease Agreements that have the same amount of Deposits.

Solution

Cl scr

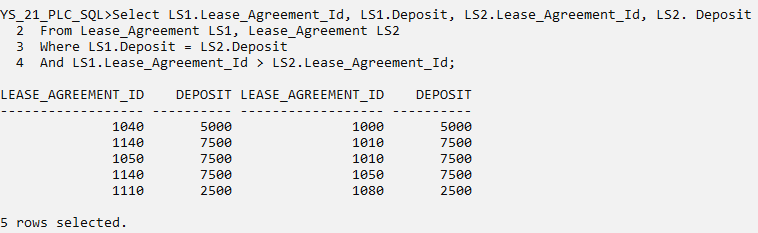
Select LS1.Lease\_Agreement\_Id, LS1.Deposit, LS2.Lease\_Agreement\_Id, LS2.Deposit

From Lease\_Agreement LS1, Lease\_Agreement LS2

Where LS1.Deposit = LS2.Deposit

And LS1.Lease\_Agreement\_Id > LS2.Lease\_Agreement\_Id;

Snip



### 6.1.2 Outer Join

Question.2

List the Name, City, Specialism and Website of every Letting Agent in Dublin, and where applicable, include the Deposit and Installment Amount for all the Lease Agreements facilitated by the Letting Agents with Instalment Amount of at least 20000 and at most 50000.

Solution

Cl scr

Select LT.Name, LT.City, LT.Specialism, LT.Web\_Site, LS.Deposit, LS.Installment\_Amount

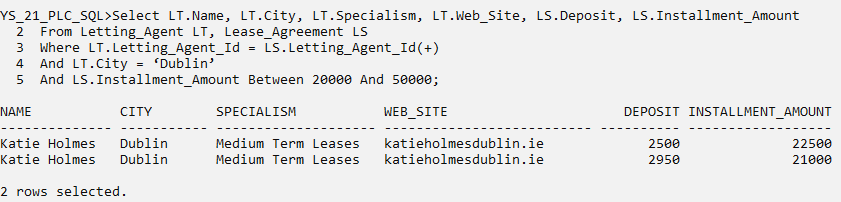
From Letting\_Agent LT, Lease\_Agreement LS

Where LT.Letting\_Agent\_Id = LS.Letting\_Agent\_Id(+)

And LT.City = ‘Dublin’

And LS.Installment\_Amount Between 20000 And 50000;

Snip



## **6.2 Subqueries**

### 6.2.1 Using Comparison Operators

Question.3

List the Start Date, End Date and Installment Amount of those Lease Agreements whose amount is higher than the combined Annual Turnover of Letting Agents with a Fair Reputation.

Solution

Cl scr

Select LS.Start\_Date, LS.End\_Date, LS.Installment\_Amount

From Lease\_Agreement LS

Where LS.Installment\_Amount >

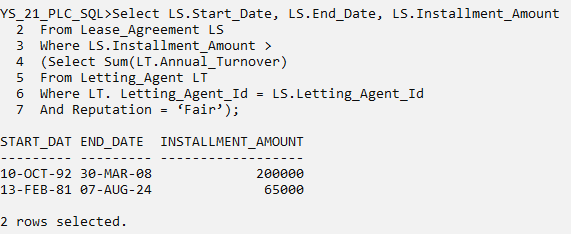
(Select Sum(LT.Annual\_Turnover)

From Letting\_Agent LT

Where LT. Letting\_Agent\_Id = LS.Letting\_Agent\_Id

And Reputation = ‘Fair’);

Snip



### 6.2.2 Using Any and All

Question.4

Give the Name, City and Area of Pubs in meters whose Specialism is Sports and whose Current Values are:

1. Larger than at least one of the Pubs whose Specialism is Parties

And

1. Shorter than every Pub whose Specialism is Ambience

Solution

Cl scr

Select Name, City, Current\_Value, Specialism

From Pub

Where Specialism = ‘Sports’

And Current\_Value > Any

(Select Current\_Value

From Pub

Where Specialism = ‘Parties’)

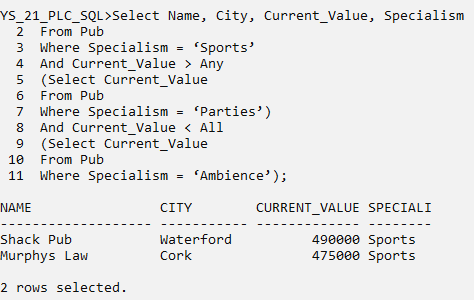
And Current\_Value < All

(Select Current\_Value

From Pub

Where Specialism = ‘Ambience’);

Snip



### 6.2.3 Using Exists

Question.5

State all the Male Publicans, their Qualifications that have not participated in any Lease Agreements yet.

Solution

Cl scr

Select PL.Fname, PL.Sname, PL.Gender, PL.Qualifications

From Publican PL

Where PL.Gender = ‘M’

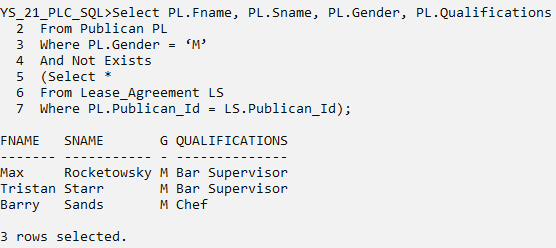
And Not Exists

(Select \*

From Lease\_Agreement LS

Where PL.Publican\_Id = LS.Publican\_Id);

Snip



### 6.2.4 Multi-legged Subquery

Question.6

Display the Lease Agreement Ids, Start Date, End Date and Method of Payment of those Agreements whose Instalment Frequency is Monthly which:

1. Are facilitated by Letting Agents whose Turnover is 30000 or more annually and have a Good or Excellent Public Reputation

And

1. Are arranged for those pubs which are Built after 1990 whose Landlords have specialized on Large Pubs

Or

1. Whose Publicans are Females having Qualifications as Bar Managers.

Solution

Cl scr

Select Lease\_Agreement\_Id, Start\_Date, End\_Date, Installment\_frequency

From Lease\_Agreement

Where Installment\_frequency = ‘Monthly’

And Letting\_Agent\_Id In

(Select Letting\_Agent\_Id

From Letting\_Agent

Where Annual\_Turnover >= 30000

And Reputation In (‘Good’, ‘Excellent’))

And Pub\_Id In

(Select Pub\_Id

From Pub

Where Year\_Built > 1990

And Pub\_Id In

(Select Pub\_Id

From Landlord

Where Specialism = ‘Large Pubs’))

Or Publican\_Id In

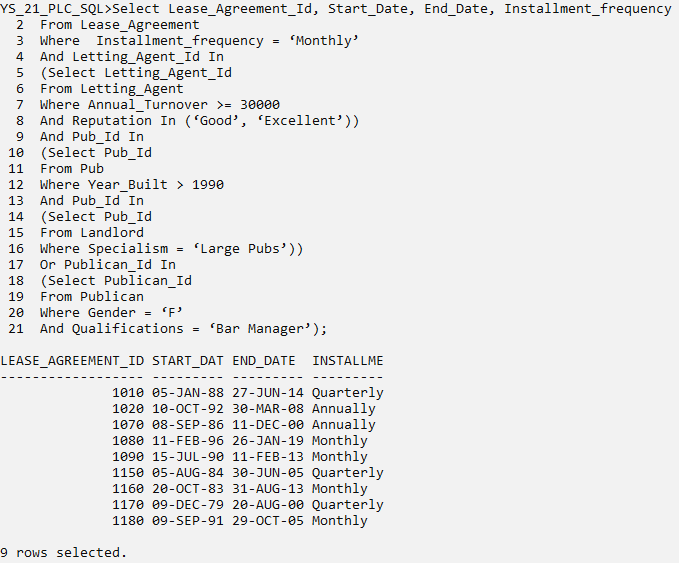
(Select Publican\_Id

From Publican

Where Gender = ‘F’

And Qualifications = ‘Bar Manager’);

Snip



## **6.3 Update, Delete and Alter Queries**

### 6.3.1 Update Query

Question.7

Decrease by 500, the Deposits of Lease Agreements payed by Direct Debit or Standing Order which are formulated by Solicitors that do not have their offices in the counties of major cities of Ireland (i.e. Dublin, Cork, Galway, Limerick).

Solution

Cl scr

Select Deposit, Payment\_Method

From Lease\_Agreement

Where Payment\_Method In (‘Direct Debit’, ‘Standing Order’)

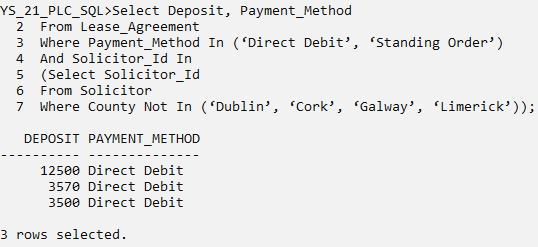
And Solicitor\_Id In

(Select Solicitor\_Id

From Solicitor

Where County Not In (‘Dublin’, ‘Cork’, ‘Galway’, ‘Limerick’));

Snip 1



Update Lease\_Agreement

Set Deposit = Deposit - 500

Where Payment\_Method In (‘Direct Debit’, ‘Standing Order’)

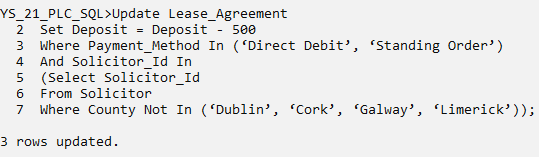
And Solicitor\_Id In

(Select Solicitor\_Id

From Solicitor

Where County Not In (‘Dublin’, ‘Cork’, ‘Galway’, ‘Limerick’));

Snip 2



Select Deposit, Payment\_Method

From Lease\_Agreement

Where Payment\_Method In (‘Direct Debit’, ‘Standing Order’)

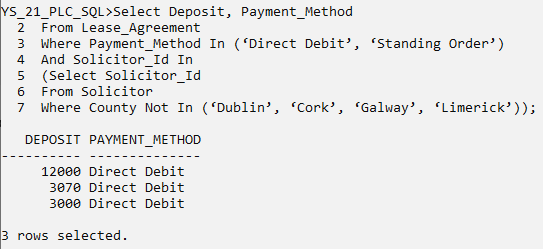
And Solicitor\_Id In

(Select Solicitor\_Id

From Solicitor

Where County Not In (‘Dublin’, ‘Cork’, ‘Galway’, ‘Limerick’));

Snip 3



### 6.3.2 Delete and Alter Queries

Question.8

Delete the Pubs with capacity less than or equal to 500 which are leased by Landlords whose Specialism is Small Pubs or whose Installment Amount in Lease Agreements is less than or equal to 30000.

Solution

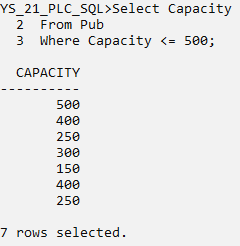
Cl scr

Select Capacity

From Pub

Where Capacity <= 500;

Snip 1



Delete

From Pub

Where Capacity <= 500

And Landlord\_Id In

(Select Landlord\_Id

From Landlord

Where Specialism = ‘Small Pubs’)

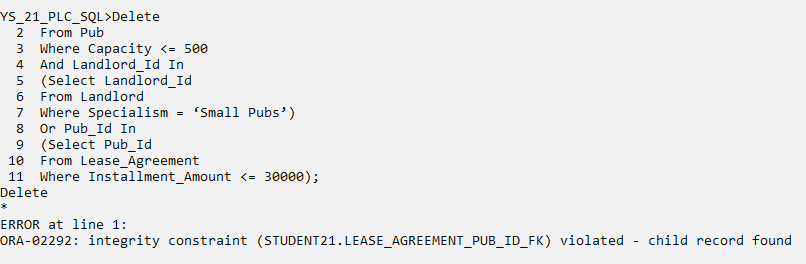
Or Pub\_Id In

(Select Pub\_Id

From Lease\_Agreement

Where Installment\_Amount <= 30000);

Snip 2

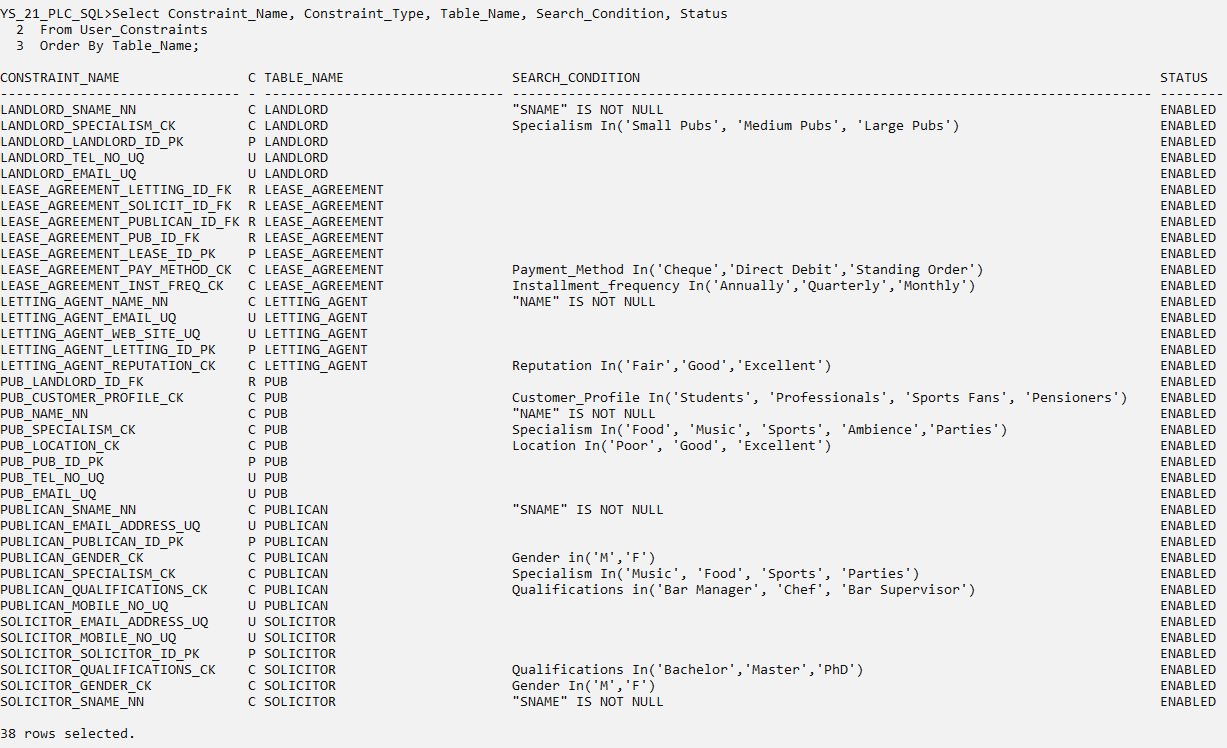


Select Constraint\_Name, Constraint\_Type, Table\_Name, Search\_Condition, Status

From User\_Constraints

Order By Table\_Name;

Snip 3

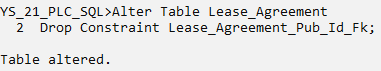


Cl scr

Alter Table Lease\_Agreement

Drop Constraint Lease\_Agreement\_Pub\_Id\_Fk;

Snip 4



Alter Table Lease\_Agreement

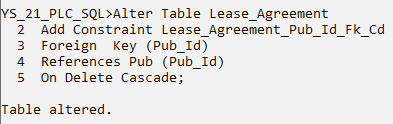
Add Constraint Lease\_Agreement\_Pub\_Id\_Fk\_Cd

Foreign Key (Pub\_Id)

References Pub (Pub\_Id)

On Delete Cascade;

Snip 5



Delete

From Pub

Where Capacity <= 500

And Landlord\_Id In

(Select Landlord\_Id

From Landlord

Where Specialism = ‘Small Pubs’)

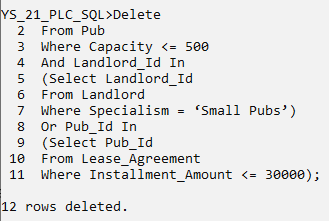
Or Pub\_Id In

(Select Pub\_Id

From Lease\_Agreement

Where Installment\_Amount <= 30000);

Snip 6

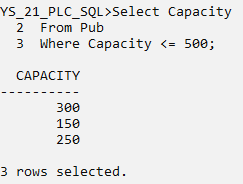


Select Capacity

From Pub

Where Capacity <= 500;

Snip 7



## **6.4 Views**

### 6.4.1 With Check Option

Question.9

Create a permanently stored query that will include the Landlord\_id, Surname, Town, County and Specialism for Landlords who leases Large Pubs. Then, use the permanently stored query to demonstrate that:

1. new rows may add to the table
2. existing rows may be modified
3. rows may be removed

Ensure that any data manipulation on the view is consistent with any criteria in that query.

Solution

Cl scr

Create View Large\_Pub\_Landlords As

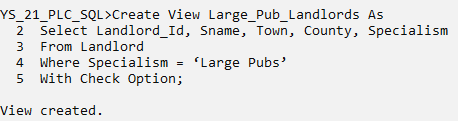
Select Landlord\_Id, Sname, Town, County, Specialism

From Landlord

Where Specialism = ‘Large Pubs’

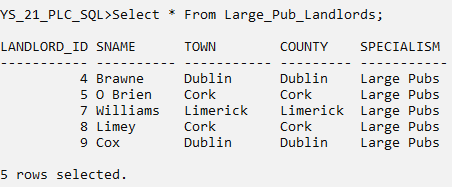
With Check Option;

Snip 1



Select \* From Large\_Pub\_Landlords;

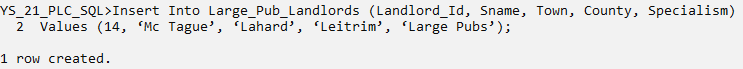
Snip 2



Insert Into Large\_Pub\_Landlords (Landlord\_Id, Sname, Town, County, Specialism)

Values (14, ‘Mc Tague’, ‘Lahard’, ‘Leitrim’, ‘Large Pubs’);

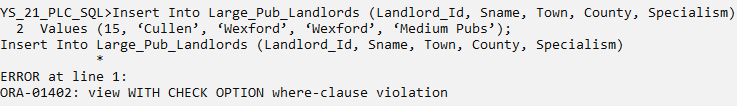
Snip 3



Insert Into Large\_Pub\_Landlords (Landlord\_Id, Sname, Town, County, Specialism)

Values (15, ‘Cullen’, ‘Wexford’, ‘Wexford’, ‘Medium Pubs’);

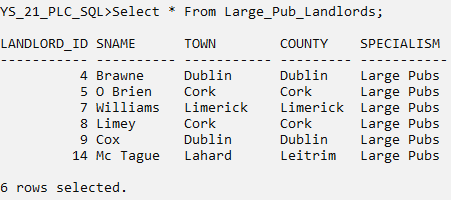
Snip 4



Cl scr

Select \* From Large\_Pub\_Landlords;

Snip 5



There has been a mistake in the insert. Actually Mr. Mc Tague’s office is in Ballinamore and not in Lahard. Please correct that.

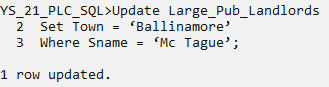
Cl scr

Update Large\_Pub\_Landlords

Set Town = ‘Ballinamore’

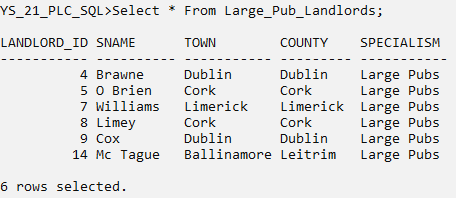
Where Sname = ‘Mc Tague’;

Snip 6



Select \* From Large\_Pub\_Landlords;

Snip 7



Mr. Mc Tague is actually retiring as a landlord, so remove that entry.

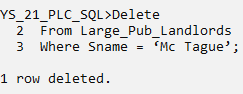
Cl scr

Delete

From Large\_Pub\_Landlords

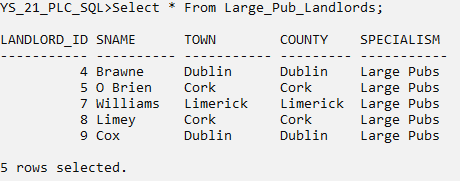
Where Sname = ‘Mc Tague’;

Snip 8



Select \* From Large\_Pub\_Landlords;

Snip 9



## **6.5 Grouping Queries**

### 6.5.1 Rollup with Grouping Sets and Union Alternatives

Question.10

List the Names of Pubs, its City, County and Customer Profile; aggregate total pub area in kilometers on levels within County -> City -> Customer Profile hierarchy.

Solution

#### Using Rollup

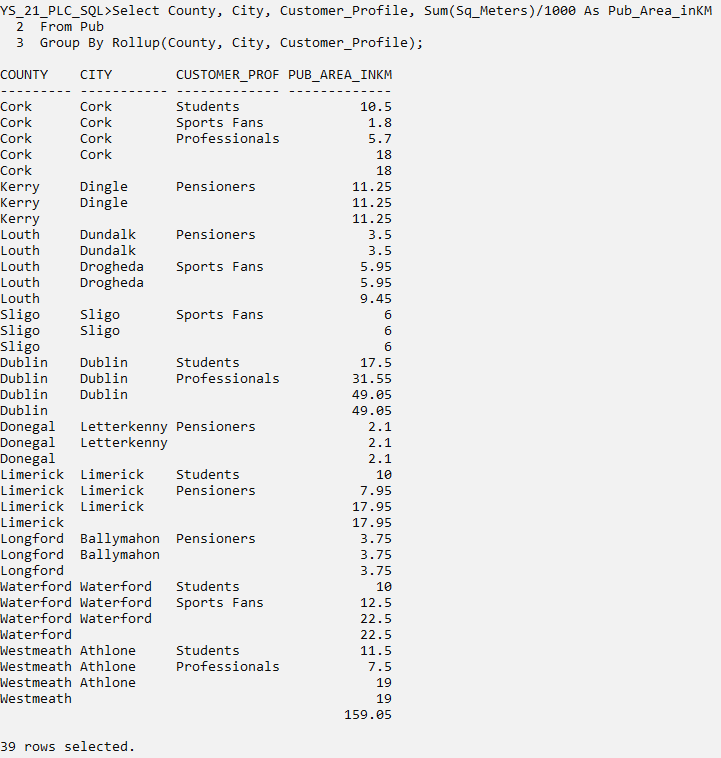
Cl scr

Select County, City, Customer\_Profile, Sum(Sq\_Meters)/1000 As Pub\_Area\_inKM

From Pub

Group By Rollup(County, City, Customer\_Profile);

Snip 1



#### Using Grouping Sets

Cl scr

Select County, City, Customer\_Profile, Sum(Sq\_Meters)/1000 As Pub\_Area\_inKM

From Pub

Group By Grouping Sets

(

(County, City, Customer\_Profile),

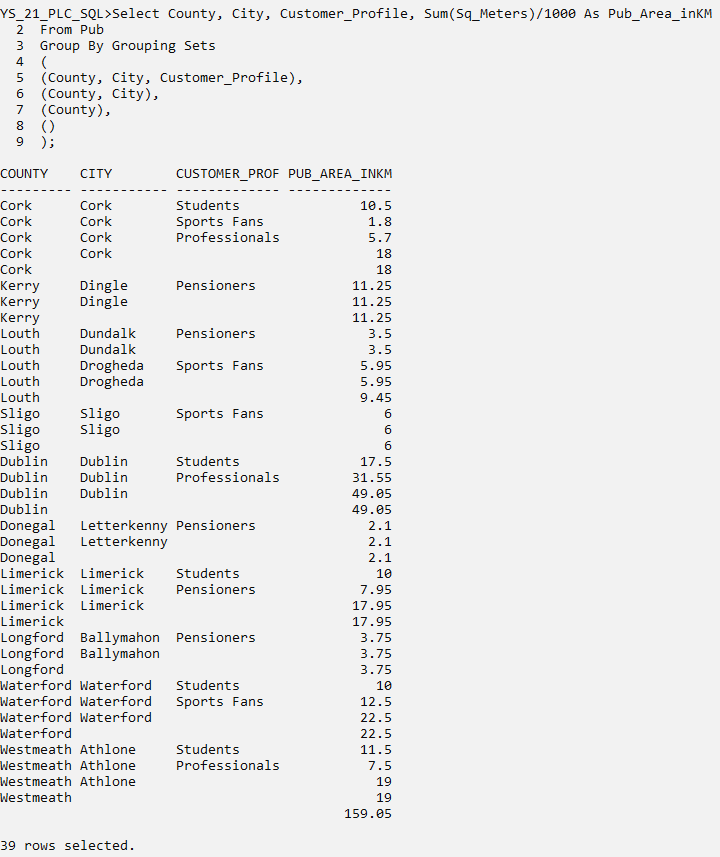
(County, City),

(County),

()

);

Snip 2



#### Using Unions with Simple Groupby

Cl scr

Select County, City, Customer\_Profile, Sum(Sq\_Meters)/1000 As Pub\_Area\_inKM

From Pub

Group By County, City, Customer\_Profile

UNION ALL

Select County, City, NULL, Sum(Sq\_Meters)/1000 As Pub\_Area\_inKM

From Pub

Group By County, City

UNION ALL

Select County, NULL, NULL, Sum(Sq\_Meters)/1000 As Pub\_Area\_inKM

From Pub

Group By County

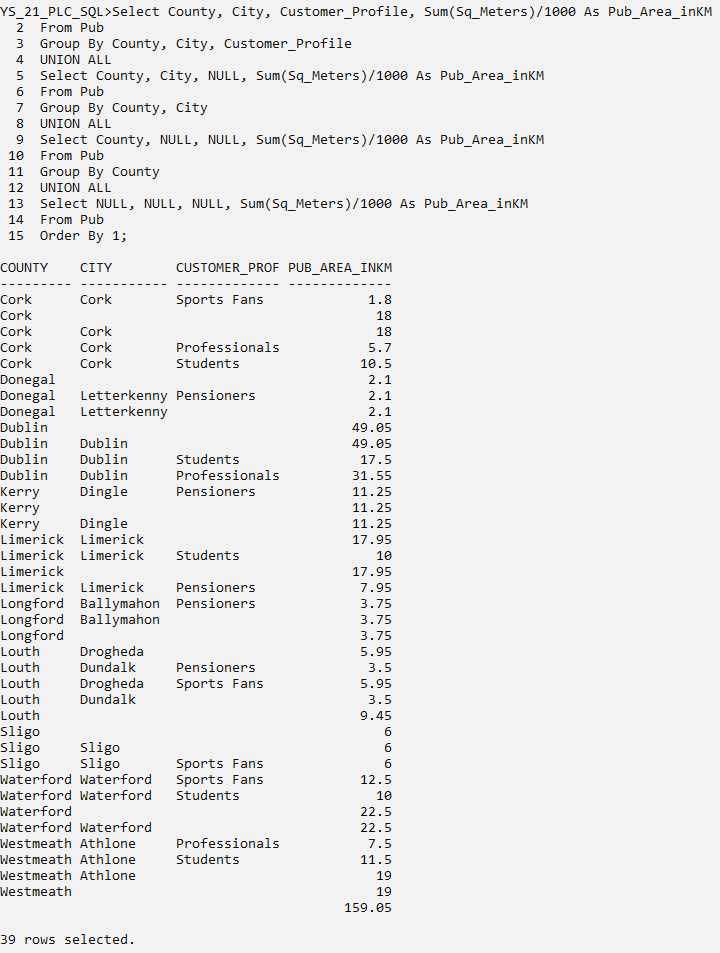
UNION ALL

Select NULL, NULL, NULL, Sum(Sq\_Meters)/1000 As Pub\_Area\_inKM

From Pub

Order By 1;

Snip 3



### 6.5.2 Cube with Grouping Sets and Union Alternatives

Question.11

List the Names of Pubs, its Customer Profile, Specialism, Location and Current Value. Show aggregates for total Current Value of pubs in millions on all combinations within Customer Profile, Specialism and Location. Order by Customer Profile.

Solution

#### Using Cube

Cl scr

Select Customer\_Profile, Specialism, Location, Sum(Current\_Value)/1000000 As Value\_inMillions

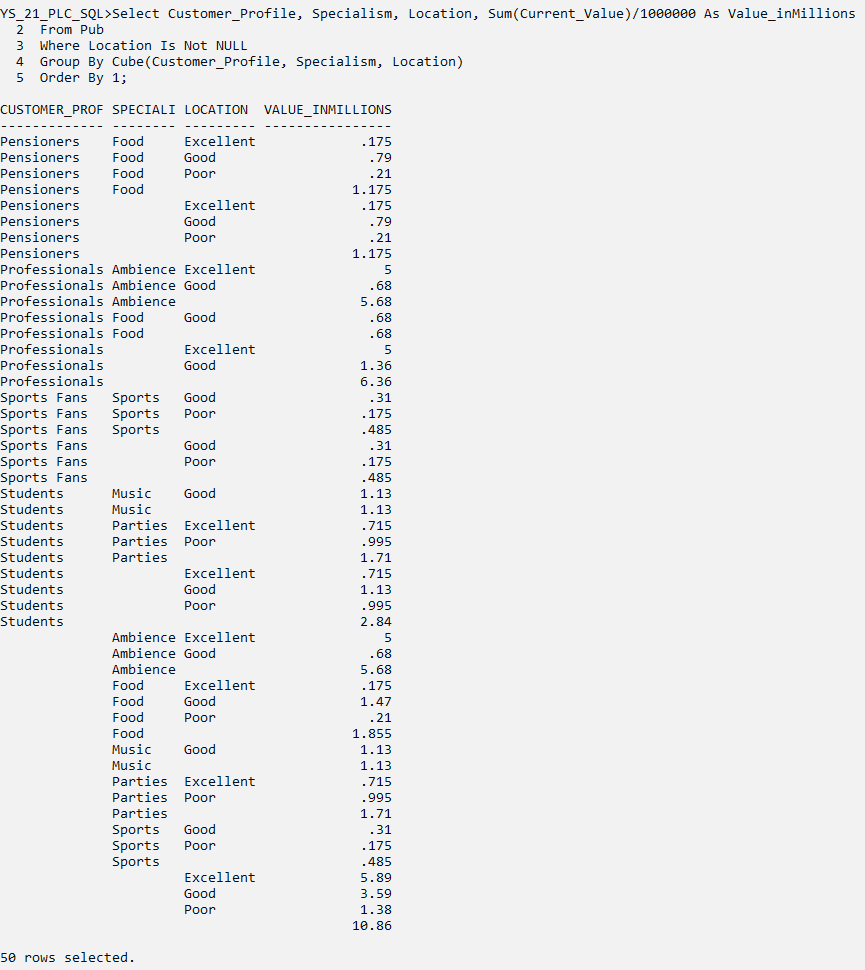
From Pub

Where Location Is Not NULL

Group By Cube(Customer\_Profile, Specialism, Location)

Order By 1;

Snip 1



#### Using Grouping Sets

Cl scr

Select Customer\_Profile, Specialism, Location, Sum(Current\_Value)/1000000 As Value\_inMillions

From Pub

Where Location is Not NULL

Group By Grouping Sets

(

(Customer\_Profile, Specialism, Location),

(Customer\_Profile, Specialism),

(Customer\_Profile, Location),

(Customer\_Profile),

(Specialism, Location),

(Specialism),

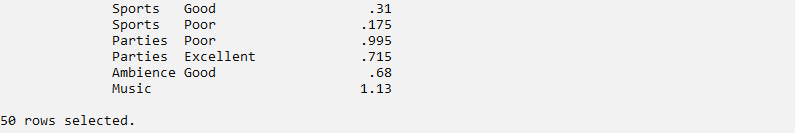
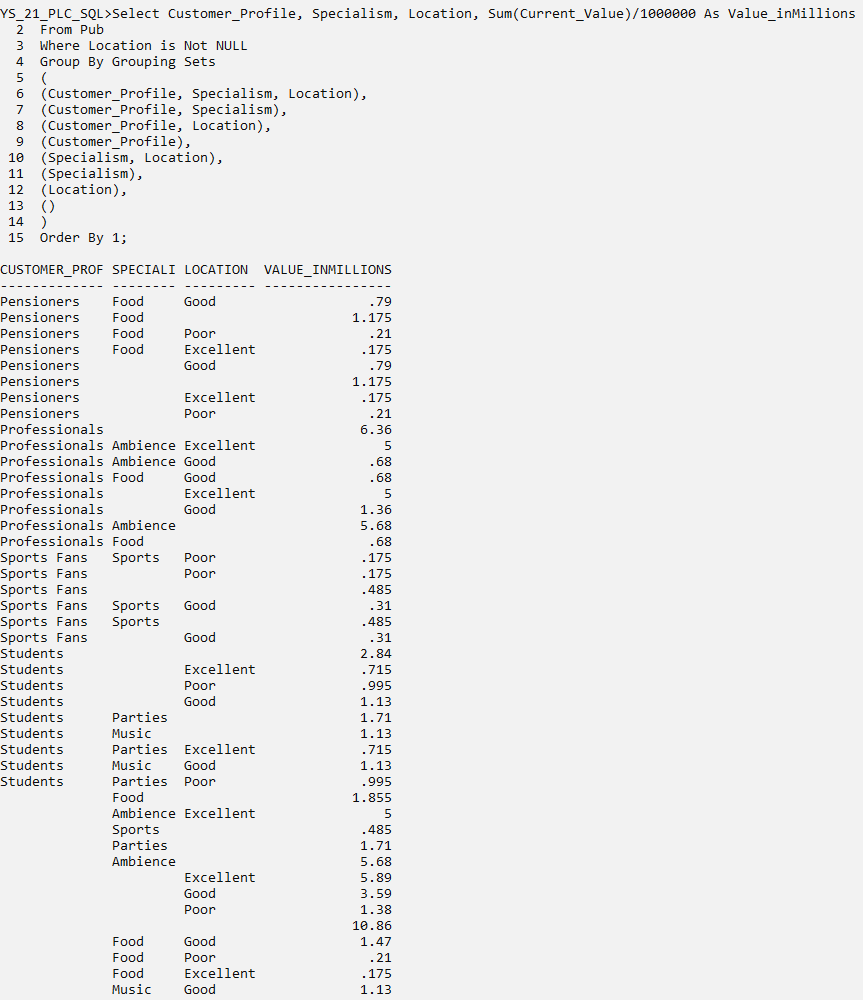
(Location),

()

)

Order By 1;

Snip 2



#### Using Unions with Simple Groupby

Cl scr

Select Customer\_Profile, Specialism, Location, Sum(Current\_Value)/1000000 As Value\_inMillions

From Pub

Where Location Is Not NULL

Group By Customer\_Profile, Specialism, Location

UNION ALL

Select Customer\_Profile, Specialism, NULL, Sum(Current\_Value)/1000000 As Value\_inMillions

From Pub

Group By Customer\_Profile, Specialism

UNION ALL

Select Customer\_Profile, NULL, Location, Sum(Current\_Value)/1000000 As Value\_inMillions

From Pub

Where Location Is Not NULL

Group By Customer\_Profile, Location

UNION ALL

Select Customer\_Profile, NULL, NULL, Sum(Current\_Value)/1000000 As Value\_inMillions

From Pub

Group By Customer\_Profile

UNION ALL

Select NULL, Specialism, Location, Sum(Current\_Value)/1000000 As Value\_inMillions

From Pub

Where Location Is Not NULL

Group By Specialism, Location

UNION ALL

Select NULL, Specialism, NULL, Sum(Current\_Value)/1000000 As Value\_inMillions

From Pub

Group By Specialism

UNION ALL

Select NULL, NULL, Location, Sum(Current\_Value)/1000000 As Value\_inMillions

From Pub

Where Location is Not NULL

Group By Location

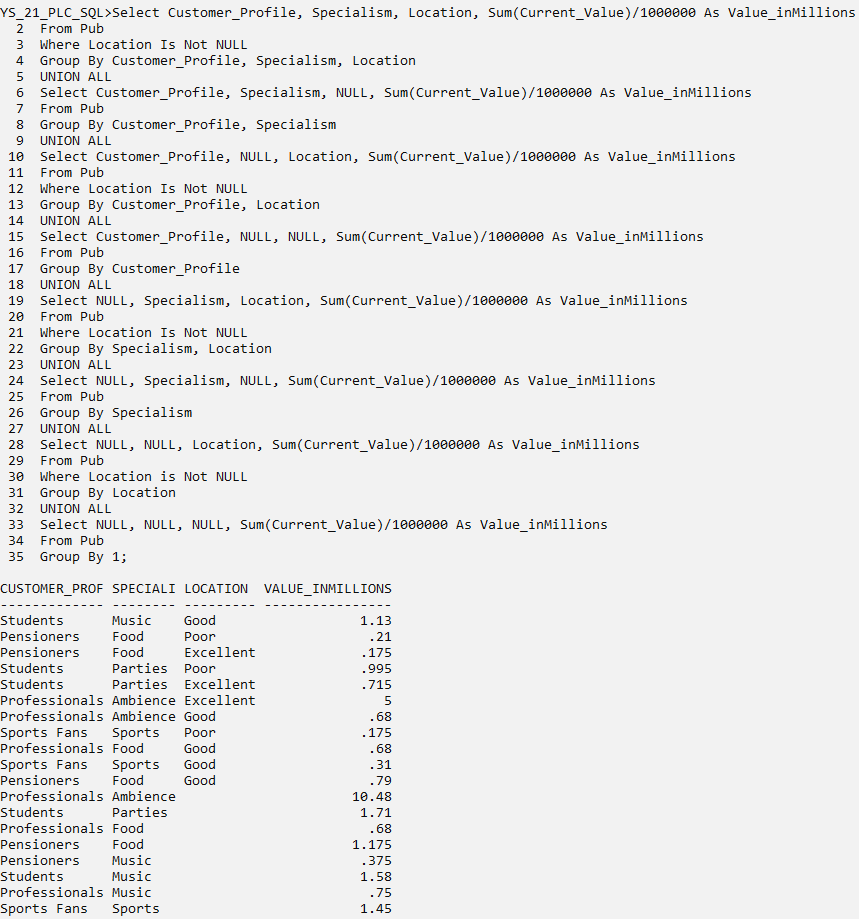
UNION ALL

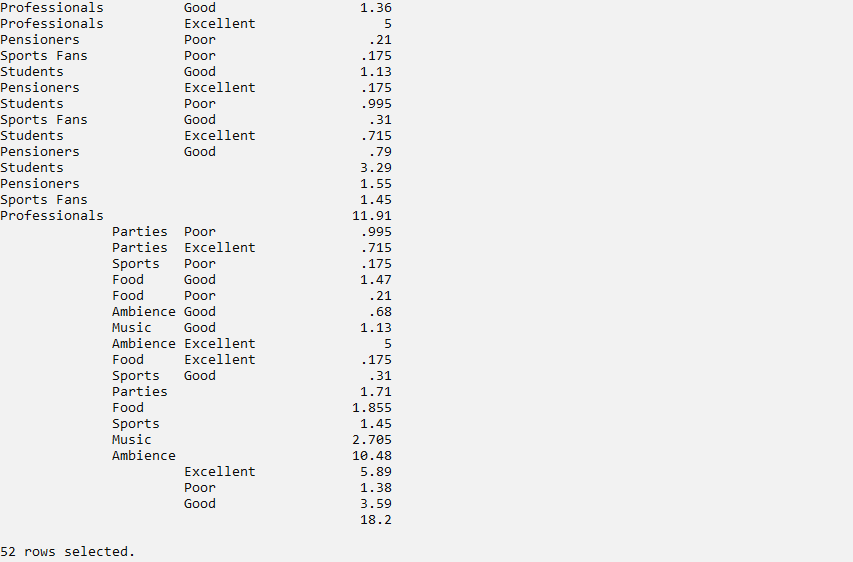
Select NULL, NULL, NULL, Sum(Current\_Value)/1000000 As Value\_inMillions

From Pub

Group By 1;

Snip 3





## **6.6 Partitioning Queries**

### 6.6.1 Aggregates with Running Total and Row Number

Question.12

Show Name, City, Specialism, Annual Turnover along with aggregated count for Specialism and aggregated average, min and max for Annual Turnover, partitioned by Specialism for each entry of Letting Agents.

Also, provide a running total for Annual Turnover within partitions of Specialism by each Letting Agent. And then, add row numbers for each partition in the end.

Solution

Cl scr

Select Name, City, Specialism, Annual\_Turnover,

Count(Specialism) Over (Partition By Specialism) As SpecialismTotals,

Avg(Annual\_Turnover) Over (Partition By Specialism) As AverageTurnover,

Min(Annual\_Turnover) Over (Partition By Specialism) As MinTurnover,

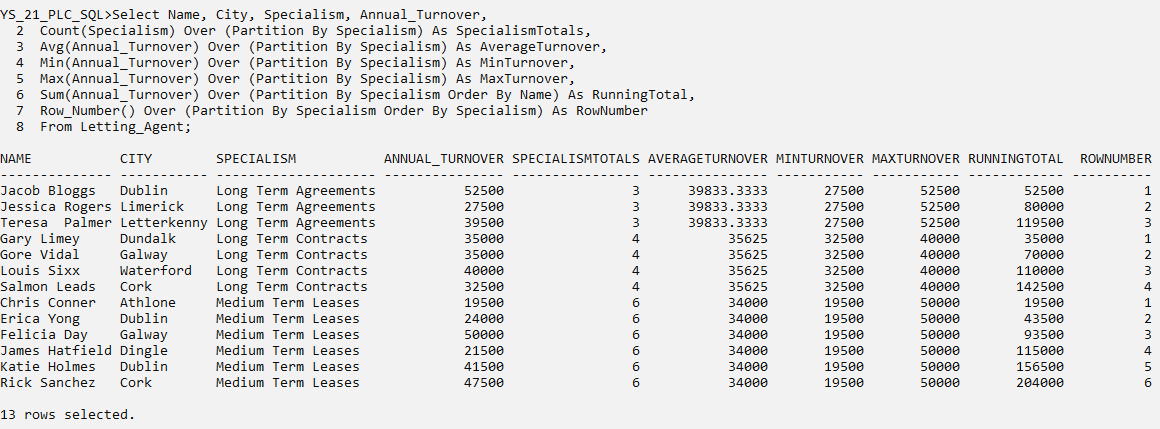
Max(Annual\_Turnover) Over (Partition By Specialism) As MaxTurnover,

Sum(Annual\_Turnover) Over (Partition By Specialism Order By Name) As RunningTotal,

Row\_Number() Over (Partition By Specialism Order By Specialism) As RowNumber

From Letting\_Agent;

Snip 1



### 6.6.2 Dense Rank

Question.13

List the Name, Address, City and Capacity of Pubs. Partition the results on the basis of Customer Profile and within partitions rank the results on the basis of decreasing Pub Capacity.

Then, display the top Pubs for each Customer Profile separately.

Solution

Cl scr

Select Name, Address, City, Customer\_Profile, Capacity,

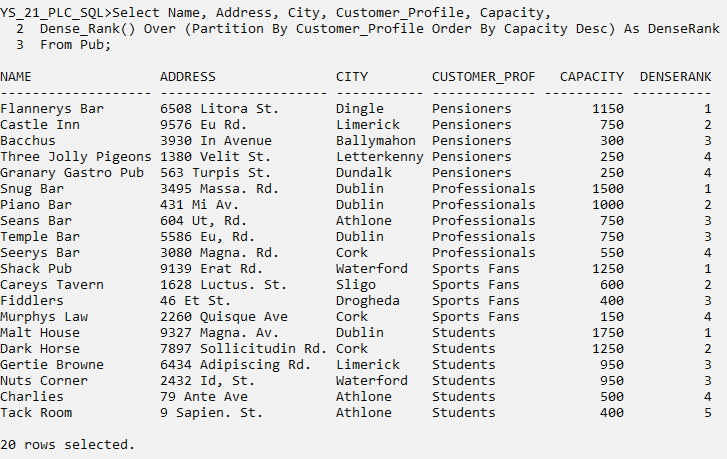
Dense\_Rank()

Over

(Partition By Customer\_Profile Order By Capacity Desc) As DenseRank,

From Pub;

Snip 1



With Tops As

(

Select Name, Address, City, Customer\_Profile, Capacity,

Dense\_Rank()

Over

(Partition By Customer\_Profile Order By Capacity Desc) As DenseRank

From Pub

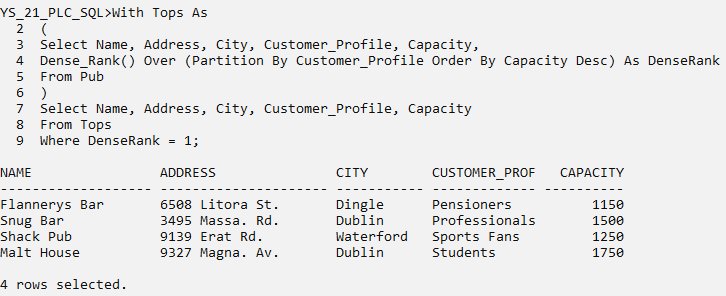
)

Select Name, Address, City, Customer\_Profile, Capacity

From Tops

Where DenseRank = 1;

Snip 2



### 6.6.3 Lead and Lag

Question.14

List Name, City, Specialism and Year when founded for each Letting Agent. Also, include the year founded for previous Letting Agent with the year founded of next letting agent within each specialism partition.

Display -1 when there are no previous or next value.

Solution

Cl scr

Select Name, City, Specialism,

Lag(Year\_Founded, 1, -1)

Over

(Partition By Specialism Order By Year\_Founded) As Prev\_Founded,

Year\_Founded,

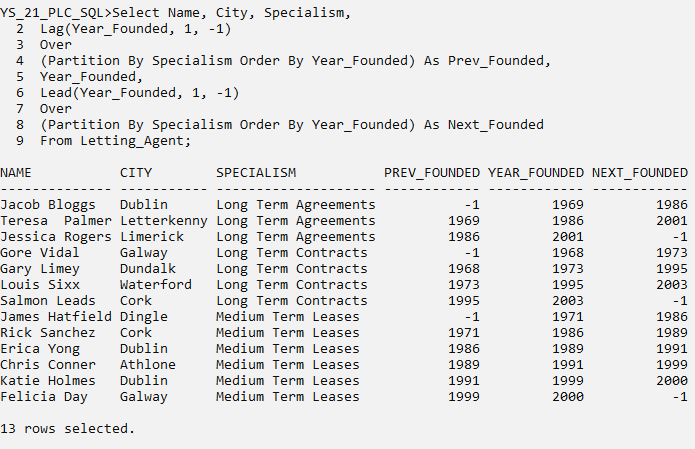
Lead(Year\_Founded, 1, -1)

Over

(Partition By Specialism Order By Year\_Founded) As Next\_Founded

From Letting\_Agent;

Snip 1



### 6.6.4 Windowing Aggregates

Question.15

Display Name, Address, City, Location and Current Value for each Pub. Also, show the count, average and sum of each Current Value entry with the two previous and two next entries.

Show the average and sum in millions.

Solution

Cl scr

Select Name, Address, City, Location, Current\_Value,

Count(Current\_Value)

Over

(Order By Current\_Value Rows Between 2 Preceding And 2 Following)

As Count\_Value,

Avg((Current\_Value)/1000000)

Over

(Order By Current\_Value Rows Between 2 Preceding And 2 Following)

As Avg\_Value,

Sum((Current\_Value)/1000000)

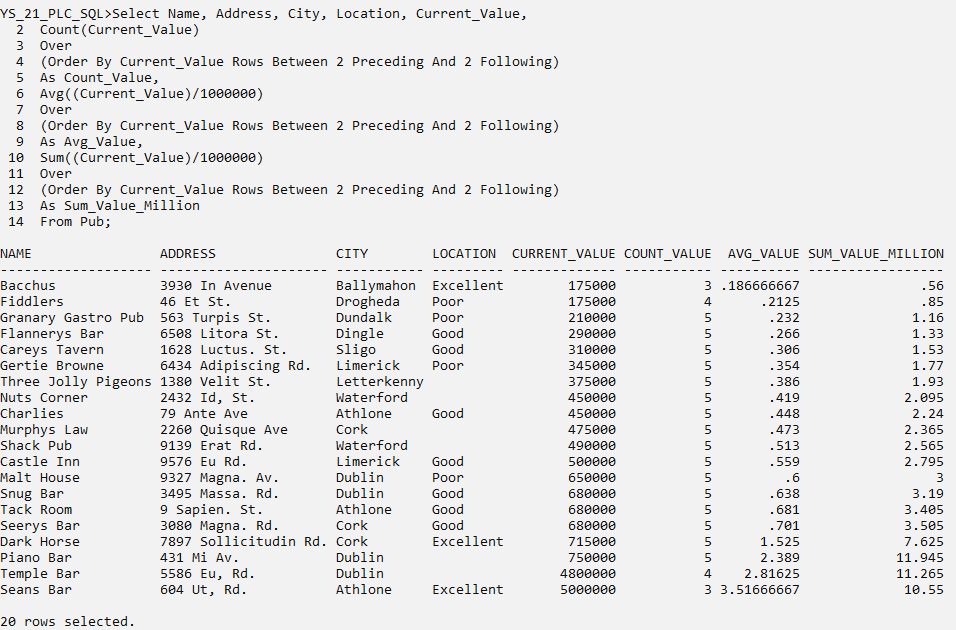
Over

(Order By Current\_Value Rows Between 2 Preceding And 2 Following)

As Sum\_Value\_Million

From Pub;

Snip 1



### 6.6.5 First Value and Last Value

Question.16

Show Name, City, Reputation, Annual\_Turnover, and the Name of the Letting Agent with minimum turnover and maximum turnover within each partition by Reputation.

Solution

Cl scr

Select Name, City, Reputation, Annual\_Turnover,

First\_Value(Name)

Over

(Partition By Reputation Order By Annual\_Turnover)

As Min\_Turnover\_Name,

Last\_Value(Name)

Over

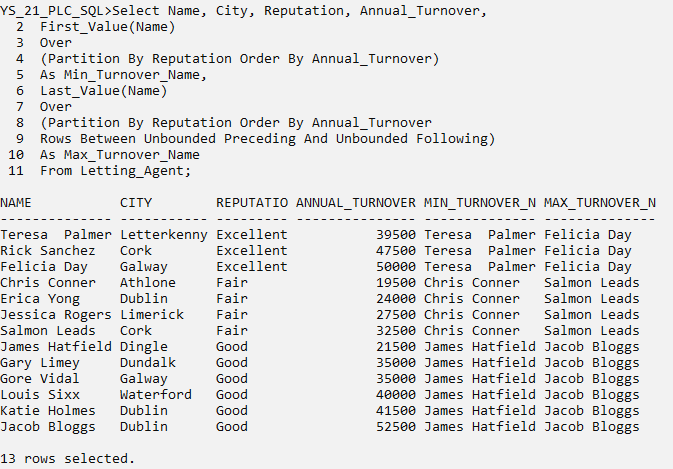
(Partition By Reputation Order By Annual\_Turnover

Rows Between Unbounded Preceding And Unbounded Following)

As Max\_Turnover\_Name

From Letting\_Agent;

Snip 1



## **6.7 Case Expressions**

### 6.7.1 Case Labels

Question.17

Show Name, City, Year when built, Customer Profile for each Pub and display labels in separate column for year when built as follows:

1. Before 1800, ‘Very Old’
2. Between 1800 and 1900, ‘Old’
3. Between 1900 and 2000, ‘New’
4. After 2000, ‘Recent’

Solution

Cl scr

Select Name, City, Year\_Built, Customer\_Profile,

(Case

When Year\_Built < 1800 Then ‘Very Old’

When Year\_Built Between 1800 And 1900 Then ‘Old’

When Year\_Built Between 1900 And 2000 Then ‘New’

When Year\_Built > 2000 Then ‘Recent’

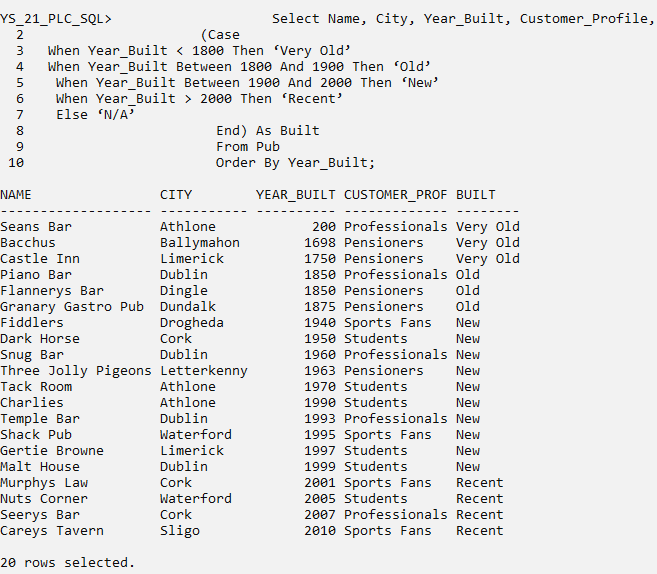
Else ‘N/A’

End) As Built

From Pub

Order By Year\_Built;

Snip 1



### 6.7.2 Labels Within Groups

Question.18

Show Name, County, Specialism, Reputation, Annual\_Turnover for each Letting\_Agent. Also, display the labels for Letting Agents in two separate columns as follows:

i. Top Earning within each Specialism

ii. Lowest Earning within each Specialism

iii. Top Earning within each Reputation

iv. Lowest Earning within each Reputation

Solution

Cl scr

Select Name, County, Specialism, Reputation, Annual\_Turnover,

Case When Annual\_Turnover = Max\_Specialism

Then ‘Top Earn In Spclsm’

When Annual\_Turnover = Min\_Specialism

Then ‘Low Earn In Spclsm’

End Specialism\_Status,

Case When Annual\_Turnover = Max\_Reputation

Then ‘Top Earn In Reputn’

When Annual\_Turnover = Min\_Reputation

Then ‘Low Earn In Reputn’

End Reputation\_Status

From (Select Name, County, Specialism, Reputation, Annual\_Turnover,

Max(Annual\_Turnover) Over (Partition By Specialism) Max\_Specialism,

Min(Annual\_Turnover) Over (Partition By Specialism) Min\_Specialism,

Max(Annual\_Turnover) Over (Partition By Reputation) Max\_Reputation,

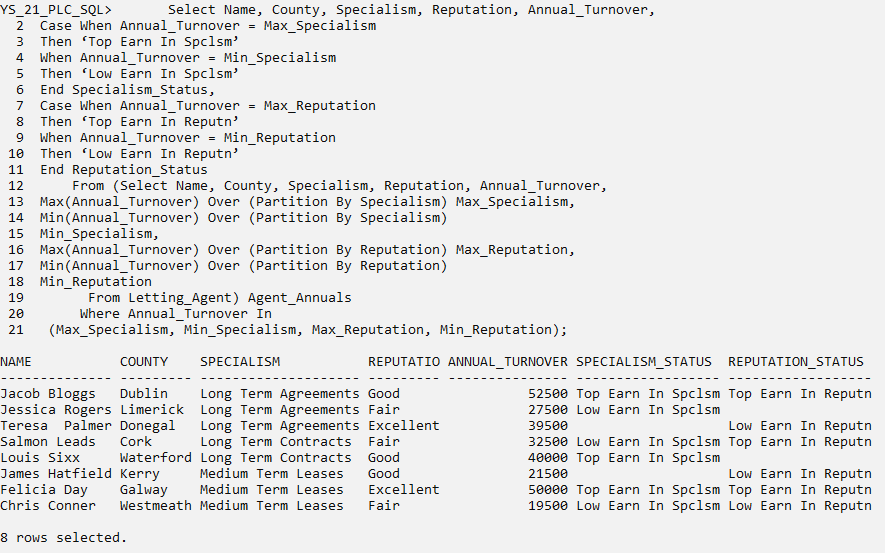
Min(Annual\_Turnover) Over (Partition By Reputation) Min\_Reputation

From Letting\_Agent) Agent\_Annuals

Where Annual\_Turnover In

(Max\_Specialism, Min\_Specialism, Max\_Reputation, Min\_Reputation);

Snip 1



### 6.7.3 Pivoting into Multiple Rows

Question.19

Pivot Surname for each Publican on the basis of Qualifications, i.e. form columns for each category in Qualifications and display Surname for each Publican under their corresponding category.

Solution

Cl scr

Select Qualifications, Sname,

Row\_Number()

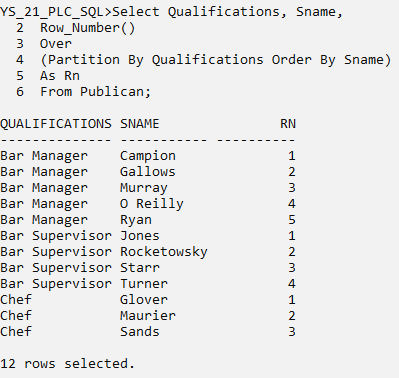
Over

(Partition By Qualifications Order By Sname)

As Rn

From Publican;

Snip 1



Select Max(Case When Qualifications = ‘Bar Manager’

Then Sname Else Null End) As Managers,

Max(Case When Qualifications = ‘Bar Supervisor’

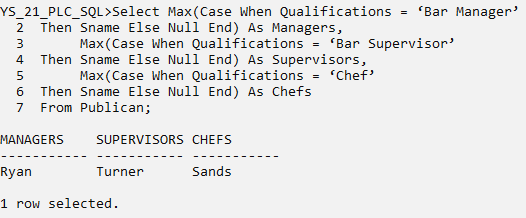
Then Sname Else Null End) As Supervisors,

Max(Case When Qualifications = ‘Chef’

Then Sname Else Null End) As Chefs

From Publican;

Snip 2



Select Rn,

Case When Qualifications = ‘Bar Manager’

Then Sname Else Null End As Managers,

Case When Qualifications = ‘Bar Supervisor’

Then Sname Else Null End As Supervisors,

Case When Qualifications = ‘Chef’

Then Sname Else Null End As Chefs

From (Select Qualifications, Sname,

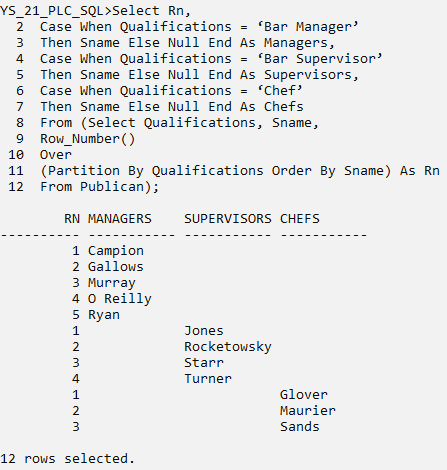
Row\_Number()

Over

(Partition By Qualifications Order By Sname) As Rn

From Publican);

Snip 3



Select Max(Case When Qualifications = ‘Bar Manager’

Then Sname Else Null End) As Managers,

Max(Case When Qualifications = ‘Bar Supervisor’

Then Sname Else Null End) As Supervisors,

Max(Case When Qualifications = ‘Chef’

Then Sname Else Null End) As Chefs

From (Select Qualifications, Sname,

Row\_Number()

Over

(Partition by Qualifications Order By Sname) Rn

From Publican) X

Group By Rn;

Snip 4

