**ISEM 3006 Data Management in Business**

**First Semester 2020/21**

**Exercise 3**

**SQL SELECT 1**

Trustful Property Rental Company Database

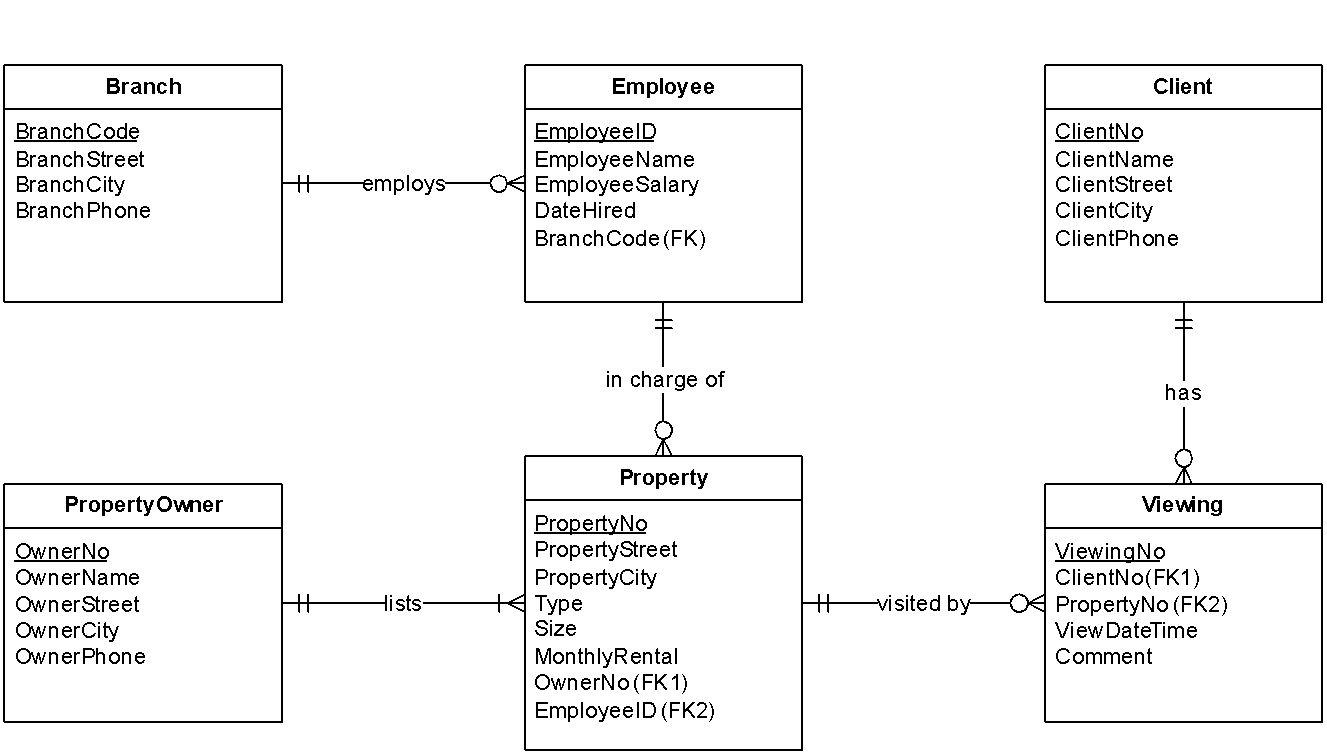
The Entity Relationship Model shown below models the data requirements supporting the operation of the Trustful Property Rental Company.

Trustful Property Rental Company is a real estate agency which focuses on the residential rental market. The company has a number of branches throughout Hong Kong. Each branch has a unique branch code, a street address, a city, and a phone number. A branch employs many employees, none if it is a new branch, but an employee works in only one branch. Each employee has a unique employee ID. Name, salary, date hired of an employee are also recorded.

A property owner can list one or many properties for rent. One property belongs to only one property owner. Each property owner has a unique owner number. The company also records the name, street address, city, and the phone number of the owners. Each property has a unique property number. The street address, the city, property type, size (in square feet), and monthly rental amount of the properties are also recorded.

An employee can be in charge of one or many properties, none if he or she is a new employee. A property is taken care by exactly one employee only.

A client is a person who wants to rent a property. The company records a unique client number, name, street address, city, and phone for each of its client. A client may have viewed many properties for comparison. He/she may even view the same property a number of times. For each "Viewing", a unique viewing number is assigned; the date and time viewed and the comment of the client is also recorded. A Client may have none, one, or many viewings. A viewing belongs to only one client. A property may have many viewings, none if it has not been viewed by anyone. A viewing only visits one property.



The following tables show the rows from each table.

Branch

|  |  |  |  |
| --- | --- | --- | --- |
| BranchCode | BranchStreet | BranchCity | BranchPhone |
| NT1 | 23 Tai Ho Street | NT | 24681357 |
| KLN2 | 35 Nattan Road | KLN | 24681234 |
| NT2 | 1 Wing Fat Ave | NT | 34445566 |

Employee

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| EmployeeID | EmployeeName | EmployeeSalary | DateHired | BranchCode |
| 90001 | Chan Tai Man | 15000 | 2015-05-12 | NT1 |
| 90002 | Li ho ma | 20000 | 2016-01-11 | KLN2 |
| 90003 | Wong Tai Shuk | 13000 | 2016-02-12 | KLN2 |

Client

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ClientNo | ClientName | CleintStreet | ClientCity | ClientPhone |
| 1001 | Lee Ka Shing | 225 Tai Po Road | NT | 22221111 |
| 1002 | Kwok Tsing | 334 Sha Tin Road | NT | 10001200 |
| 1003 | Tin Liang | 55 Fo Wah Street | KLN | 10002222 |
| 5001 | Lee Ah Ho | 123 Low Street | MOS | 20010011 |

PropertyOwner

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| OwnerNo | OwnerName | OwnerStreet | OwnerCity | OwnerPhone |
| 678 | Wong Peter | 9 Central Road | HK | 23397766 |
| 321 | Ng Paul | 8 Tin Shui Ave | NT | 34111123 |
| 3344 | Au Mary | 10 Tai Ho Street | KLN | 28097654 |

Property

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| PropertyNo | ProperyStreet | PropertyCity | Type | Size | Monthly Rental | OwnerNo | EmployeeID |
| 1 | 1 Devok Road | HK | Flat | 1000 | 30000 | 678 | 90001 |
| 2 | 3 Peak Road | HK | House | 3500 | 76800 | 321 | 90001 |
| 3 | 6 Ho H Street | KLN | Flat | 1388 | 25000 | 678 | 90002 |
| 4 | 4 Low Street | MOS | Flat | 500 | 15000 | 3344 | 90002 |
| 5 | 9 Nine Ave | KLN | Flat | 90 | 9000 | 3344 | 90002 |

Viewing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ViewingNo | ClientNo | PropertyNo | ViewDateTime | Comment |
| 1881 | 1001 | 1 | 2019-10-02 10:20:00 | Can consider |
| 1992 | 1001 | 2 | 2019-11-09 11:30:00 |  |
| 1993 | 5001 | 2 | 2019-12-25 14:15:00 | Good |
| 2001 | 1001 | 3 | 2020-02-02 09:17:00 | Bad smell |
| 2005 | 5001 | 2 | 2020-02-05 10:15:00 | Too expensive |
| 3016 | 1001 | 4 | 2020-03-31 18:45:00 | Too remote |
| 4001 | 5001 | 4 | 2020-04-04 20:20:00 |  |
| 4002 | 1002 | 1 | 2020-04-04 21:50:00 | Very good |
| 4466 | 1002 | 2 | 2020-05-04 16:00:00 | Too expensive |
| 5888 | 1001 | 5 | 2020-06-04 18:30:00 | Just right |
| 6543 | 5001 | 5 | 2020-06-28 11:15:00 | OK |
| 7890 | 1001 | 1 | 2020-06-29 18:00:00 | A bit small |

**Use SQL Select statement to answer the following queries. Unless state otherwise, the case of strings is important, i.e. if I ask you to retrieve properties that are located in 'KLN', you only retrieve properties that are located in 'KLN', but not 'kln', 'Kln', 'kLn', etc.**

**The ordering of the rows is not important, unless I ask you to sort the output.**

1. List the property number, address, type and size of the property which is located in the city ‘KLN’ and the size is larger than 1000 square feet.

SELECT property.propertyNo, property.propertystreet,

property.Type, property.Size FROM property

WHERE property.propertycity='KLN' AND property.size>=1000;

1. Suppose the properties have been rented out at the amount in MonthlyRental. List the property number, type, size, monthly rental, and the annual rental income from the property (use Annual Rental Income as the column heading) for each of the property.

SELECT property.propertyNo, property.Type, property.Size ,

property.MonthlyRental,

property.MonthlyRental\*12 AS "Annual Rental"

FROM property

1. List the employee ID and the name of the employee who have the word ‘Tai’ in his/her name and the third character of the name is ‘n’.

SELECT Employee.EmployeeID , Employee.EmployeeName FROM Employee

WHERE (Employee.EmployeeName LIKE '%Tai%')

AND

(Employee.EmployeeName LIKE '\_\_n%')

1. List the detail of the properties which are 'Flat'. Order the result by the owner number in ascending order, then within each owner, sort the monthly rental in descending order.

SELECT \* FROM property

WHERE property.Type='Flat'

ORDER BY OwnerNo ASC, MonthlyRental DESC ;

1. List the employee ID and the name of the employee who have the word ‘Ho’ in his/her name. The case is not important, i.e., you should also retrieve employees with 'HO', 'hO', etc. in his/her name.

SELECT Employee.EmployeeID , Employee.EmployeeName FROM Employee

WHERE upper(Employee.EmployeeName) LIKE '%HO%';

1. For each property owner, list the owner number, name, and the 2nd to 6th characters of his/her name.

SELECT OwnerNo ,OwnerName,

SUBSTRING(OwnerName, 2, 5) AS ExtractName

FROM PropertyOwner;

1. For each employee, list the employee number, name, his/her date hired, and the 5th anniversary of he or she being hired.

SELECT Employee.EmployeeID ,

Employee.EmployeeName,

Employee.DateHired ,

Employee.DateHired + interval '5 years' AS anniversary

FROM Employee ;

1. Suppose the company deposits the salary to the employees’ bank account on the last date of the month. For each employee, list the employee number, name, his/her date hired, and the date that he or she received the first salary.

SELECT Employee.EmployeeID ,

Employee.EmployeeName,

Employee.DateHired ,

date\_trunc('MONTH', (Employee.DateHired + interval '1 month')) - interval '1 day'

AS receive\_date

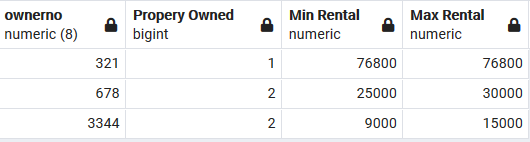
FROM Employee ;

1. List the viewing that happened on Tuesday in the year 2020.

SELECT \* FROM Viewing

where To\_CHAR(ViewDateTime,'YYYY') = '2020'

AND To\_CHAR(ViewDateTime,'dy') = 'tue';

1. For each property owner, list the owner number, the number of properties he/she listed with the company, the highest and the lowest monthly rental of his/her listed properties.

SELECT OwnerNo,

COUNT(PropertyNo) AS “Property Owned”

Min(Monthly Rental) AS “Min Rental” ,

Max(Monthly Rental) AS “Max Rental”

FROM Property

GROUP BY OwnerNo ;

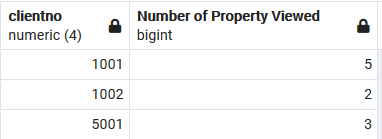
1. List property number and the number of viewings it has for those property having more than 2 viewings. The output should look like the following:

SELECT PropertyNo, COUNT(ViewingNo) AS “Number of viewings”

FROM Viewing

GROUP BY PropertyNo ;

HAVING COUNT(ViewingNo) > 2

1. For each client, list out the client number and the number of properties that he/she has viewed. (Note that a client may view a property more than 1 times.) The output should look like the following:

SELECT ClientNo, COUNT( DISTINCT PropertyNo) AS “Number of property viewed”

FROM Viewing

GROUP BY ClientNo ;

/\*

If you have other tables with the same name (such as Employee table in the Task database) in your

database, drop them first.

The reason is that the following drop table statement may fail if your existing tables

have foreign key constraint that may affect the drop table operation.

OR you can create another database and create the tables in the new database. But remember to open the Query Tool under that databse.

\*/

drop table if exists viewing;

drop table if exists property;

drop table if exists employee;

drop table if exists branch;

drop table if exists client;

drop table if exists propertyowner;

create table branch (

branchcode char(5) primary key,

Branchstreet varchar(20) not null,

branchcity char(4) not null,

branchphone char(8) not null

);

create table employee (

employeeID numeric(6) primary key,

employeename varchar(15) not null,

employeesalary numeric(6) not null,

datehired date not null,

branchcode char(5) not null

REFERENCES branch (branchcode)

ON DELETE RESTRICT);

create table client (

clientno numeric(4) primary key,

clientname varchar(15) not null,

clientstreet varchar(20) not null,

clientcity char(4) not null,

clientphone char(8) not null);

create table propertyowner (

ownerno numeric(8) primary key,

ownername varchar(15) not null,

ownerstreet varchar(20) not null,

ownercity char(4) not null,

ownerphone char(8) not null);

create table property (

propertyno numeric(6) primary key,

propertystreet varchar(15) not null,

propertycity char(4) not null,

type char(6) not null,

size numeric(5) not null,

monthlyrental numeric(10) not null,

ownerno numeric(8) not null,

employeeid numeric(6) not null,

foreign key (ownerno)

references PropertyOwner (ownerno)

on delete restrict,

foreign key (employeeid)

references employee (employeeid)

on delete restrict) ;

create table viewing (

viewingno numeric(5) primary key,

clientno numeric(4) not null,

propertyno numeric(6) not null,

viewdatetime timestamp not null,

comments varchar(15),

foreign key (clientno)

references client (clientno)

on delete restrict,

foreign key (propertyno)

references Property (propertyno)

on delete restrict );

insert into branch values ('NT1', '23 Tai Ho Street', 'NT', '24681357');

insert into branch values ('KLN2', '32 Nattan Road', 'KLN', '24681234');

insert into branch values ('NT2', '1 Wing Fat Ave', 'NT', '34445566');

insert into employee values (90001,'Chan Tai Man', 15000, '2015-05-12', 'NT1');

insert into employee values (90002,'Li ho ma', 20000, '2016-01-11', 'KLN2');

insert into employee values (90003,'Wong Tai Shuk', 13000, '2016-02-12', 'KLN2');

insert into client values (1001, 'Lee Ka Shing', '225 Tai Po Road', 'NT', '22221111');

insert into client values (1002, 'Kwok Tsing','334 Sha Tin Road','NT','10001200');

insert into client values (1003, 'Tin Liang','55 Fo Wah Street','KLN','10002222');

insert into client values (5001, 'Lee Ah Ho','123 Low Street','MOS','20010011');

insert into propertyowner values (678,'Wong Peter','9 Central Road','HK','23397766');

insert into propertyowner values (321,'Ng Paul','8 Tin Shui Ave','NT','34111123');

insert into propertyowner values (3344,'Au Mary','10 Tai Ho Street','KLN','28097654');

insert into property values (1,'1 Devok Road','HK','Flat',1000,30000,678,90001);

insert into property values (2,'3 Peak Road','HK','House',3500,76800,321,90001);

insert into property values (3,'6 Ho H Street','KLN','Flat',1388,25000,678,90002);

insert into property values (4,'4 Low Street','MOS','Flat',500,15000,3344,90002);

insert into property values (5,'9 Nine Ave','KLN','Flat',90,9000,3344,90002);

insert into viewing values (1881,1001,1,'2019-10-02 10:20:00','Can consider');

insert into viewing values (1992,1001,2,'2019-11-09 11:30:00', NULL);

insert into viewing values (1993,5001,2,'2019-12-25 14:15:00','Good');

insert into viewing values (2001,1001,3,'2020-02-02 09:17:00','Bad smell');

insert into viewing values (2005,5001,2,'2020-02-05 10:15:00','Too expensive');

insert into viewing values (3016,1001,4,'2020-03-31 18:45:00','Too remote');

insert into viewing values (4001,5001,4,'2020-04-04 20:20:00', NULL);

insert into viewing values (4002,1002,1,'2020-04-04 21:50:00','Very good');

insert into viewing values (4466,1002,2,'2020-05-04 16:00:00','Too expensive');

insert into viewing values (5888,1001,5,'2020-06-04 18:30:00','Just right');

insert into viewing values (6543,5001,5,'2020-06-28 11:15:00','OK');

insert into viewing values (7890,1001,1,'2020-06-29 18:00:00','A bit small');