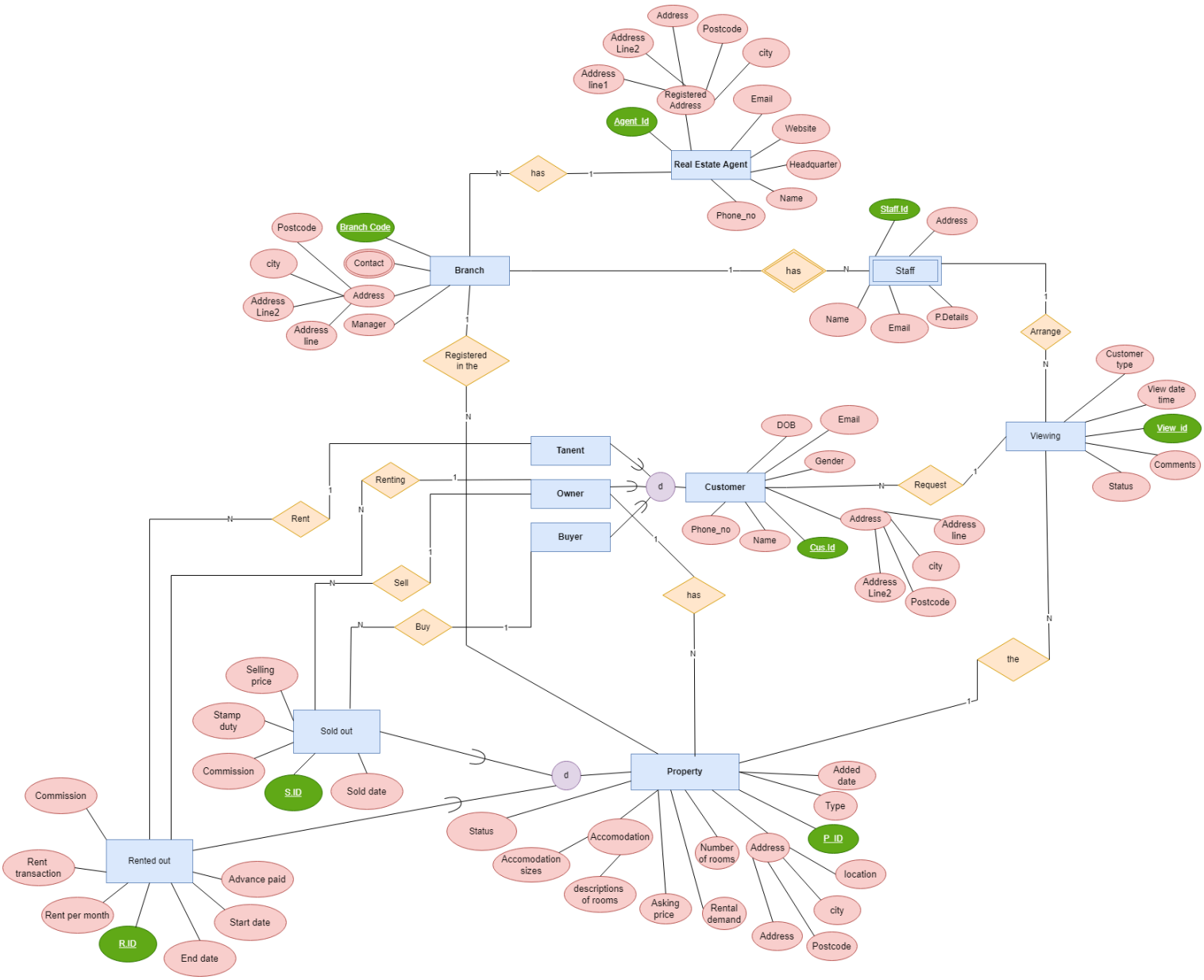
This part is based on the MOVEHOME scenario

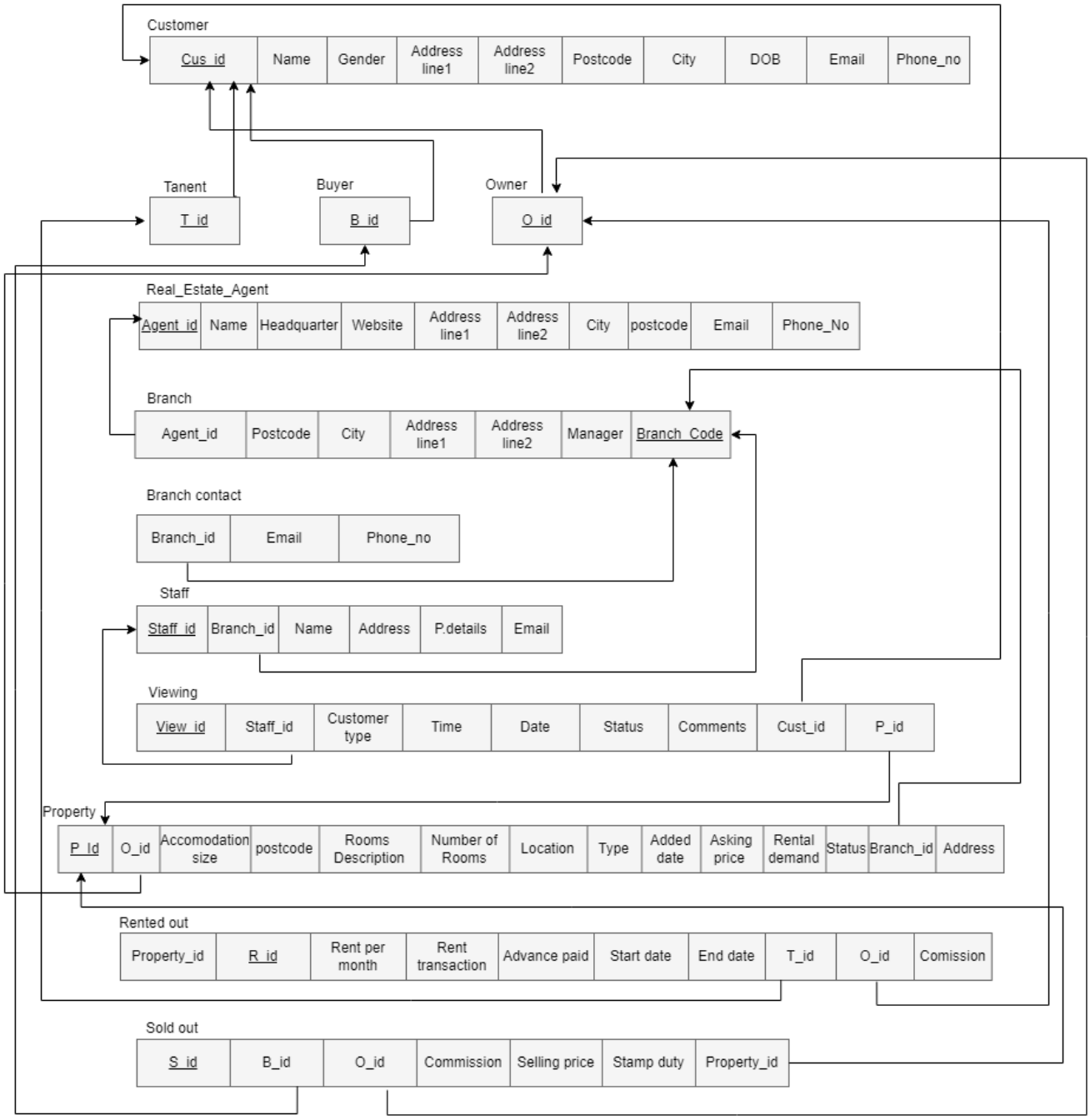
Using entity-relationship (ER) OR enhanced entity-relationship (EER) modelling, produce a conceptual design for the database to support the MOVEHOME business activities.

ER or EER Diagram Below



Convert the ER / EER diagram from Part 1(A) to produce a logical relational schema using ER / EER to relational mapping.

Logical Relational Design/Schema Below



Based on your logical design from Part 1 (B) and the information available in the scenario, produce an SQL script file using Oracle 11g/12c/higher.

SQL DDL Script file contents (i.e., the SQL code for creating / altering your Tables / Constraints etc)

```
CREATE TABLE customer(
        cust_id CHAR(2) NOT NULL,
        name VARCHAR(50),
        gender VARCHAR(10),
        address_line1 VARCHAR(50),
        address_line2 VARCHAR(50),
        city VARCHAR(20) NOT NULL,
        postcode VARCHAR(10) NOT NULL,
        dob DATE,
        email VARCHAR(20) NOT NULL,
        phone_no VARCHAR(20),
        CONSTRAINT pkey_customer Primary Key(cust_id));
CREATE TABLE owner(
        o_id CHAR(2) NOT NULL
        CONSTRAINT uniq_owner_id UNIQUE
        CONSTRAINT owner_fk REFERENCES customer(cust_id));
CREATE TABLE buyer(
        b_id CHAR(2) NOT NULL
        CONSTRAINT uniq_buyer_id UNIQUE
        CONSTRAINT buyer_fk REFERENCES customer(cust_id));
CREATE TABLE tenant(
        t_id CHAR(2) NOT NULL
        CONSTRAINT uniq_tenant_id UNIQUE
        CONSTRAINT tenant_fk REFERENCES customer(cust_id));
CREATE TABLE real_estate_agent(
        agent_id CHAR(2) NOT NULL,
        name VARCHAR(50),
```

```
        headquarter VARCHAR(50),
        website VARCHAR(50),
        address_line1 VARCHAR(50),
        address_line2 VARCHAR(50),
        city VARCHAR(20) NOT NULL,
        postcode VARCHAR(10) NOT NULL,
        phone_no VARCHAR(15),
        email VARCHAR(50),
        CONSTRAINT pkey_agent Primary Key(agent_id));
CREATE TABLE branch(
        branch_code CHAR(10) NOT NULL,
        address_line1 VARCHAR(50),
        address_line2 VARCHAR(50),
        city VARCHAR(20) NOT NULL,
        postcode VARCHAR(10) NOT NULL,
        manager VARCHAR(20) NOT NULL,
        agent_id CHAR(2) NOT NULL
        CONSTRAINT agent_id_branch UNIQUE
        CONSTRAINT fk_agent_id_branch REFERENCES
real_estate_agent(agent_id),
        CONSTRAINT pkey_branch Primary Key(branch_code));
CREATE TABLE branch_contact(
        branch_id CHAR(10) NOT NULL
        CONSTRAINT branch_contact_id_branch UNIQUE
        CONSTRAINT fk_branch_id REFERENCES branch(branch_code),
        email VARCHAR(50),
        phone_number VARCHAR(50));
```

```
CREATE TABLE staff(
        staff_id CHAR(10) NOT NULL,
        branch_id CHAR(10) NOT NULL
        CONSTRAINT staff_id_branch UNIQUE
        CONSTRAINT fk_branch_id_staff REFERENCES branch(branch_code),
        name VARCHAR(50),
        address VARCHAR(50),
        personal_details VARCHAR(50) NOT NULL,
        email VARCHAR(20) NOT NULL,
        CONSTRAINT pkey_staff Primary Key(staff_id));
CREATE TABLE property(
        p_id CHAR(2) NOT NULL,
        o_id CHAR(2) NOT NULL
        CONSTRAINT property_fk_owner REFERENCES owner(o_id),
                branch_id CHAR(10) NOT NULL
        CONSTRAINT property_id_branch UNIQUE
        CONSTRAINT fk_branch_id_property REFERENCES branch(branch_code),
        address_line VARCHAR(50),
        location VARCHAR(50),
        city VARCHAR(20) NOT NULL,
        postcode VARCHAR(10) NOT NULL,
        accomodation_size VARCHAR(50) NOT NULL,
        rooms_description VARCHAR(50) NOT NULL,
        no_of_rooms NUMBER(10),
        rental_demand VARCHAR(20),
        asking_price VARCHAR(20),
        type VARCHAR(30)
```

```
            CONSTRAINT type_of_property CHECK (type IN

           ('flats', 'detached', 'semi-detached', 'terraced', 'house',
'apartment')),

        status CHAR(20)

            CONSTRAINT status_of_property CHECK (status IN

           ('for_sale', 'for_rent', 'sold_out', 'rented_out')),

        added_date DATE default sysdate,

        CONSTRAINT pkey_property Primary Key(p_id));
```

CREATE TABLE viewing(

```
        view_id CHAR(2) NOT NULL,

        staff_id CHAR(10) NOT NULL

        CONSTRAINT staff_id_viewing UNIQUE

        CONSTRAINT fk_staff_id_viewing REFERENCES staff(staff_id),

        customer_id CHAR(2) NOT NULL

        CONSTRAINT view_fk_customer REFERENCES customer(cust_id),

        property_id CHAR(2) NOT NULL

        CONSTRAINT view_fk_property REFERENCES property(p_id),

        customer_type CHAR(20)

            CONSTRAINT customer_type CHECK (customer_type IN

           ('buyer', 'tenant')),

        status CHAR(15)

            CONSTRAINT status_request CHECK (status IN

           ('approved', 'declined', 'viewed')),

        Comments VARCHAR(50),

        view_date DATE NOT NULL,

        CONSTRAINT pkey_viewing Primary Key(view_id));
```

CREATE TABLE rented_out(

```
        R_id CHAR(2) NOT NULL,
        property_id CHAR(2) NOT NULL
        CONSTRAINT rented_fk_property REFERENCES property(p_id),
        o_id CHAR(2) NOT NULL
        CONSTRAINT rented_fk_ownerpk REFERENCES owner(o_id),
        t_id CHAR(2) NOT NULL
        CONSTRAINT rented_tenant_id UNIQUE
        CONSTRAINT tenant_fk_rented REFERENCES tenant(t_id),
        rent_per_month NUMBER(10),
        rent_transaction NUMBER(10),
        advance_paid NUMBER(10),
        start_date DATE,
        end_date DATE,
        CONSTRAINT date_t_f_not_b_s
                        CHECK(end_date >= start_date),
        commission NUMBER(10),
        CONSTRAINT pkey_rented_out Primary Key(R_id));
CREATE TABLE sold_out(
        s_id CHAR(2) NOT NULL,
        property_id CHAR(2) NOT NULL
        CONSTRAINT sold_fk_property REFERENCES property(p_id),
        b_id CHAR(2) NOT NULL
        CONSTRAINT sold_fk_buyer REFERENCES buyer(b_id),
        o_id CHAR(2) NOT NULL
        CONSTRAINT rented_owner_id UNIQUE
        CONSTRAINT owner_fk_rented REFERENCES owner(o_id),
        selling_price NUMBER(10),
```

```
        stamp_duty NUMBER(10),

        commission NUMBER(10),

        sold_date DATE default sysdate,

        CONSTRAINT pkey_sold_out Primary Key(s_id));
```

SQL DDL Output (e.g., SPOOL file contents or output you got when you executed your above SQL Table Creation code, this should show the SQL code as well as its output)

```
w21040872> @C:\Users\w21040872\Downloads\creation.sql
w21040872> Set Verify on
w21040872>
w21040872> DROP TABLE customer CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872> DROP TABLE owner CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872> DROP TABLE buyer CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872> DROP TABLE tenant CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872> DROP TABLE real_estate_agent CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872> DROP TABLE agent_contact CASCADE CONSTRAINTS PURGE;
DROP TABLE agent_contact CASCADE CONSTRAINTS PURGE
           *
ERROR at line 1:
ORA-00942: table or view does not exist


w21040872> DROP TABLE branch CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872> DROP TABLE branch_contact CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872> DROP TABLE staff CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872> DROP TABLE staff_contact CASCADE CONSTRAINTS PURGE;
DROP TABLE staff_contact CASCADE CONSTRAINTS PURGE
           *
ERROR at line 1:
```

```
ORA-00942: table or view does not exist


w21040872> DROP TABLE viewing CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872> DROP TABLE property CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872> DROP TABLE sold_out CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872> DROP TABLE rented_out CASCADE CONSTRAINTS PURGE;

Table dropped.

w21040872>
w21040872>
w21040872> CREATE TABLE customer(
  2        cust_id CHAR(2) NOT NULL,
  3        name VARCHAR(50),
  4        gender VARCHAR(10),
  5        address_line1 VARCHAR(50),
  6        address_line2 VARCHAR(50),
  7        city VARCHAR(20) NOT NULL,
  8        postcode VARCHAR(10) NOT NULL,
  9        dob DATE,
 10        email VARCHAR(20) NOT NULL,
 11        phone_no VARCHAR(20),
 12        CONSTRAINT pkey_customer Primary Key(cust_id));
```

Table created.

```
w21040872>
w21040872>
w21040872> CREATE TABLE owner(
  2        o_id CHAR(2) NOT NULL
  3        CONSTRAINT uniq_owner_id UNIQUE
  4        CONSTRAINT owner_fk REFERENCES customer(cust_id));
```

Table created.

```
w21040872>
w21040872> CREATE TABLE buyer(
  2        b_id CHAR(2) NOT NULL
  3        CONSTRAINT uniq_buyer_id UNIQUE
  4        CONSTRAINT buyer_fk REFERENCES customer(cust_id));
```

Table created.

```
w21040872>
w21040872>
w21040872> CREATE TABLE tenant(
  2        t_id CHAR(2) NOT NULL
  3        CONSTRAINT uniq_tenant_id UNIQUE
  4        CONSTRAINT tenant_fk REFERENCES customer(cust_id));
```

Table created.

```
w21040872>
w21040872>
w21040872> CREATE TABLE real_estate_agent(
  2        agent_id CHAR(2) NOT NULL,
  3        name VARCHAR(50),
  4        headquarter VARCHAR(50),
  5        website VARCHAR(50),
  6        address_line1 VARCHAR(50),
  7        address_line2 VARCHAR(50),
  8        city VARCHAR(20) NOT NULL,
  9        postcode VARCHAR(10) NOT NULL,
 10        phone_no VARCHAR(15),
 11        email VARCHAR(50),
 12        CONSTRAINT pkey_agent Primary Key(agent_id));
```

Table created.

```
w21040872>
w21040872>
w21040872> CREATE TABLE branch(
  2        branch_code CHAR(10) NOT NULL,
  3        address_line1 VARCHAR(50),
  4        address_line2 VARCHAR(50),
  5        city VARCHAR(20) NOT NULL,
  6        postcode VARCHAR(10) NOT NULL,
  7        manager VARCHAR(20) NOT NULL,
  8        agent_id CHAR(2) NOT NULL
  9        CONSTRAINT agent_id_branch UNIQUE
 10        CONSTRAINT fk_agent_id_branch REFERENCES
real_estate_agent(agent_id),
 11        CONSTRAINT pkey_branch Primary Key(branch_code));
```

Table created.

```
w21040872>
w21040872>
w21040872> CREATE TABLE branch_contact(
  2        branch_id CHAR(10) NOT NULL
  3        CONSTRAINT branch_contact_id_branch UNIQUE
  4        CONSTRAINT fk_branch_id REFERENCES branch(branch_code),
  5        email VARCHAR(50),
  6        phone_number VARCHAR(50));
```

Table created.

```
w21040872>
w21040872>
w21040872> CREATE TABLE staff(
  2        staff_id CHAR(10) NOT NULL,
  3        branch_id CHAR(10) NOT NULL
  4        CONSTRAINT staff_id_branch UNIQUE
  5        CONSTRAINT fk_branch_id_staff REFERENCES branch(branch_code),
  6        name VARCHAR(50),
  7        address VARCHAR(50),
  8        personal_details VARCHAR(50) NOT NULL,
  9        email VARCHAR(20) NOT NULL,
 10        CONSTRAINT pkey_staff Primary Key(staff_id));
```

Table created.

```
w21040872>
w21040872>
w21040872>
w21040872> CREATE TABLE property(
  2        p_id CHAR(2) NOT NULL,
  3        o_id CHAR(2) NOT NULL
  4        CONSTRAINT property_fk_owner REFERENCES owner(o_id),
  5             branch_id CHAR(10) NOT NULL
  6        CONSTRAINT property_id_branch UNIQUE
  7        CONSTRAINT fk_branch_id_property REFERENCES branch(branch_code),
  8        address_line VARCHAR(50),
  9        location VARCHAR(50),
 10        city VARCHAR(20) NOT NULL,
 11        postcode VARCHAR(10) NOT NULL,
 12        accomodation_size VARCHAR(50) NOT NULL,
 13        rooms_description VARCHAR(50) NOT NULL,
 14        no_of_rooms NUMBER(10),
 15        rental_demand VARCHAR(20),
 16        asking_price VARCHAR(20),
 17        type VARCHAR(30)
 18           CONSTRAINT type_of_property CHECK (type IN
 19          ('flats', 'detached', 'semi-detached', 'terraced', 'house',
'apartment')),
 20        status CHAR(20)
 21           CONSTRAINT status_of_property CHECK (status IN
 22          ('for_sale', 'for_rent', 'sold_out', 'rented_out')),
 23        added_date DATE,
 24        CONSTRAINT pkey_property Primary Key(p_id));
```

Table created.

```
w21040872>
w21040872>
w21040872> CREATE TABLE viewing(
  2        view_id CHAR(2) NOT NULL,
  3        staff_id CHAR(10) NOT NULL
  4        CONSTRAINT staff_id_viewing UNIQUE
  5        CONSTRAINT fk_staff_id_viewing REFERENCES staff(staff_id),
  6        customer_id CHAR(2) NOT NULL
  7        CONSTRAINT view_fk_customer REFERENCES customer(cust_id),
  8        property_id CHAR(2) NOT NULL
  9        CONSTRAINT view_fk_property REFERENCES property(p_id),
 10        customer_type CHAR(20)
 11           CONSTRAINT customer_type CHECK (customer_type IN
 12          ('buyer', 'tenant')),
 13        status CHAR(15)
 14           CONSTRAINT status_request CHECK (status IN
 15          ('approved', 'declined', 'viewed')),
 16        Comments VARCHAR(50),
 17        view_date DATE NOT NULL,
 18        CONSTRAINT pkey_viewing Primary Key(view_id));
```

Table created.

```
w21040872>
w21040872> CREATE TABLE rented_out(
  2        R_id CHAR(2) NOT NULL,
  3        property_id CHAR(2) NOT NULL
  4        CONSTRAINT rented_fk_property REFERENCES property(p_id),
  5        o_id CHAR(2) NOT NULL
  6        CONSTRAINT rented_fk_ownerpk REFERENCES owner(o_id),
```

```
  7        t_id CHAR(2) NOT NULL
  8        CONSTRAINT rented_tenant_id UNIQUE
  9        CONSTRAINT tenant_fk_rented REFERENCES tenant(t_id),
 10        rent_per_month NUMBER(10),
 11        rent_transaction NUMBER(10),
 12        advance_paid NUMBER(10),
 13        start_date DATE,
 14        end_date DATE,
 15        CONSTRAINT date_t_f_not_b_s
 16                         CHECK(end_date >= start_date),
 17        commission NUMBER(10),
 18        CONSTRAINT pkey_rented_out Primary Key(R_id));
```

Table created.

```
w21040872>
w21040872> CREATE TABLE sold_out(
  2        s_id CHAR(2) NOT NULL,
  3        property_id CHAR(2) NOT NULL
  4        CONSTRAINT sold_fk_property REFERENCES property(p_id),
  5        b_id CHAR(2) NOT NULL
  6        CONSTRAINT sold_fk_buyer REFERENCES buyer(b_id),
  7        o_id CHAR(2) NOT NULL
  8        CONSTRAINT rented_owner_id UNIQUE
  9        CONSTRAINT owner_fk_rented REFERENCES owner(o_id),
 10        selling_price NUMBER(10),
 11        stamp_duty NUMBER(10),
 12        commission NUMBER(10),
 13        sold_date DATE default sysdate,
 14        CONSTRAINT pkey_sold_out Primary Key(s_id));
```

Table created.

```
w21040872>
w21040872>
w21040872>
w21040872>
w21040872>
w21040872> spool off
```

This part is based on your answer / solution to Part 1, i.e., design and implementation of the database for the MOVEHOME scenario.

Populate the database with some sample data (e.g., you should generate your own dummy data and load it into the MOVIEHOME database, consider 5 to 10 rows for each table and enough data to see meaningful output for the queries below).

SQL code below for populating the above relational database

```
INSERT INTO customer
    VALUES ('1', 'Paul', 'male', '132 kenton', 'kenton lane', 'Newcastle',
'ne42c4', '06-OCT-1978', 'paul@yahoo.com', '07700 900521');
INSERT INTO customer
    VALUES ('2', 'Allen', 'male', '139 kenton', 'kenton
lane','Newcastle','ne42c4', '06-OCT-1970', 'allen@yahoo.com', '0115 496
0529');
```

```
INSERT INTO customer
    VALUES ('3', 'Teddy',  'female', '161 fenham',
'fenham','Newcastle','ne42c4','06-OCT-1980', 'teddy@gmail.com', '0117 496
0816');
INSERT INTO customer
    VALUES ('4', 'Mark',  'male', '121 fawdon',
'fawdon','Newcastle','ne42c4', '06-OCT-1987', 'mark@hotmail.com', '0787 899
2166');
INSERT INTO customer
    VALUES ('5', 'Mouzal',  'female', '167 kenton', 'kenton
lane','Newcastle','ne42c4','06-OCT-1983', 'maouzal@yahoo.com', '0115 496
0299');
INSERT INTO customer
    VALUES ('6', 'David', 'male', '132 freman',
'freman','Newcastle','ne42c4', '06-OCT-1991', 'david@gmail.com',  '0151 496
0220');
INSERT INTO customer
    VALUES ('7', 'Ameer', 'male', '154 fenham', 'fenham',
'Newcastle','ne42c4', '06-OCT-1976', 'ameer@hotmail.com',  '020 7946
0870');
INSERT INTO customer
    VALUES ('8', 'Naba',  'female', '152 kenton', 'kenton lane',
'Newcastle','ne42c4','06-OCT-1972',  'naba@gmail.com','0118 496 0496');
INSERT INTO customer
    VALUES ('9', 'Hiba', 'female', '112 fawdon', 'fawdon',
'Newcastle','ne42c4', '06-OCT-1993', 'hiba@yahoo.com', '0118 496 0286');
INSERT INTO customer
    VALUES ('10', 'Smith', 'male', '102 kingston', 'kingston
park','Newcastle','ne42c4', '23-FEB-2009', 'smith@yahoo.com',  '0114 496
0550');
```

INSERT INTO owner

```
    VALUES ('1');
INSERT INTO owner
    VALUES ('2');
INSERT INTO owner
    VALUES ('3');
INSERT INTO owner
    VALUES ('4');
```

INSERT INTO buyer

```
    VALUES ('8');
INSERT INTO buyer
    VALUES ('9');
INSERT INTO buyer
    VALUES ('10');
```

INSERT INTO tenant

```
    VALUES ('5');
INSERT INTO tenant
    VALUES ('6');
INSERT INTO tenant
    VALUES ('7');
```

INSERT INTO real_estate_agent

```
    VALUES ('a1', 'Joli', 'Newcastle','joli.com','132 kenton', 'kenton
lane', 'Newcastle', 'ne42c4', '07878952166','joli@gmail.com');
INSERT INTO real_estate_agent
    VALUES ('a2', 'Ali', 'London','ali.com','139 kenton', 'kenton
lane','Newcastle','ne42c4','07878952167','ali@yahoo.com');
INSERT INTO real_estate_agent
    VALUES ('a3', 'Nadia','Manchester', 'nadia.com','161 fenham',
'fenham','Newcastle','ne42c4', '07878952168','nadia@gmail.com');
INSERT INTO real_estate_agent
    VALUES ('a4', 'Ruby','London', 'ruby.com','121 fawdon',
'fawdon','Newcastle','ne42c4', '07878954166','ruby@hotmail.com');
INSERT INTO real_estate_agent
    VALUES ('a5', 'Tommy', 'Derby','tommy.com','152 kenton', 'kenton lane',
'Newcastle','ne42c4', '07878952166','tommy@hotmail.com');
INSERT INTO real_estate_agent
    VALUES ('a6', 'Joel', 'Bristol','joel.com','102 kingston', 'kingston
park','Newcastle','ne42c4', '07878952166','joel@yahoo.com');
INSERT INTO real_estate_agent
    VALUES ('a7', 'iffi', 'Scotland','iffi.com','112 fawdon', 'fawdon',
'Newcastle','ne42c4','07878952166', 'iffi@yahoo.com');
```

INSERT INTO branch

```
    VALUES ('b1', '132 kenton', 'kenton lane', 'Newcastle',
'ne42c4','David','a1');
INSERT INTO branch
    VALUES ('b2', '152 kenton', 'kenton lane',
'Newcastle','ne42c4','Parker','a2');
INSERT INTO branch
    VALUES ('b3', '121 fawdon', 'fawdon','Newcastle','ne42c4','Paul','a3');
INSERT INTO branch
    VALUES ('b4', '141 fenham',
'fenham','Newcastle','ne42c4','Darwen','a4');
INSERT INTO branch
    VALUES ('b5', '139 kenton', 'kenton
lane','Newcastle','ne42c4','Smith','a5');
INSERT INTO branch
    VALUES ('b6', '161 fenham',
'fenham','Newcastle','ne42c4','Ellishy','a6');
INSERT INTO branch
```

```
    VALUES ('b7', '119 kenton', 'kenton
lane','Newcastle','ne42c4','Kurtal','a7');

INSERT INTO branch_contact
    VALUES ('b1', '0780562', 'iffi@yahoo.com');
INSERT INTO branch_contact
    VALUES ('b2', '078756', 'tommy@hotmail.com');
INSERT INTO branch_contact
    VALUES ('b3', '078556', 'ruby@hotmail.com');
INSERT INTO branch_contact
    VALUES ('b4', '078556', 'joel@yahoo.com');
INSERT INTO branch_contact
    VALUES ('b5', '078786', 'jackob@yahoo.com');

INSERT INTO staff
    VALUES ('s1', 'b1', 'Hanna', 'Brimingham England', 'hanna@yahoo.com',
'07855678556');
INSERT INTO staff
    VALUES ('s2', 'b2', 'Zoal', 'Coventry England', 'zoal@yahoo.com',
'07984565556');
INSERT INTO staff
    VALUES ('s3', 'b2', 'Jimmy', 'Newcastle England', 'jimmy@hotmail.com',
'07987655556');
INSERT INTO staff
    VALUES ('s4', 'b3', 'Kulsam','Edinburg Scotland', 'kulsam@yahoo.com',
'07098765556');
INSERT INTO staff
    VALUES ('s5', 'b4', 'Smoul', 'Manchester England', 'smoul@hotmail.com',
'07234567896');
INSERT INTO staff
    VALUES ('s6', 'b5', 'Joe', 'Dundee Scotland','zoe@gmail.com',
'07876745676');
INSERT INTO staff
    VALUES ('s7', 'b6', 'Amal', 'Swansea Wales','amal@gmail.com',
'07987654336');

INSERT INTO property
    VALUES ('p1', '1', 'b1', '132 Jesmond', 'Jesmond', 'Newcastle',
'ne42c4','40.6 square meters', '3 bedrooms', '3','null', '125000','semi-
detached', 'for_sale', '19-NOV-2021');
INSERT INTO property
    VALUES ('p2', '2', 'b2','119 Jesmond',
'Jesmond','Newcastle','ne42c4','27.6 square meters','2 bedrooms',
'2','null','290000','semi-detached', 'sold_out', '27-FEB-2021');
INSERT INTO property
    VALUES ('p3', '2', 'b3', '161 fenham',
'fenham','Newcastle','ne42c4','20.6 square meters','1
bedroom','1','null','160000', 'apartment', 'for_sale', '21-NOV-2020');
```

```
INSERT INTO property
    VALUES ('p4', '2', 'b4', '161 fawdon',
'fawdon','Newcastle','ne42c4','45.6 square meters','2 bedrooms 1
master','2', '625','null', 'detached', 'rented_out', '26-OCT-2020');
INSERT INTO property
    VALUES ('p5', '1', 'b5', '132 freman',
'freman','Newcastle','ne42c4','48.6 square meters', '3
bedrooms','3','null','145000','detached', 'for_sale', '22-NOV-2021');
INSERT INTO property
    VALUES ('p6', '1', 'b6', '121 Jesmond',
'Jesmond','Newcastle','ne42c4','60.6 square meters','5 bedrooms',
'5','600', null, 'semi-detached', 'for_rent', '28-OCT-2021');
INSERT INTO property
    VALUES ('p7', '3', 'b7','152 Jesmond', 'Jesmond',
'Newcastle','ne42c4','41.6 square meters', '3
bedrooms','3','null','110900','semi-detached', 'for_sale', '21-NOV-2021');
```

INSERT INTO viewing

```
    VALUES ('v1', 's2', '1', 'p5', 'buyer', 'approved', 'best house i was
looking for', '22-NOV-2021');
INSERT INTO viewing
    VALUES ('v2', 's3', '3', 'p7', 'buyer', 'viewed', 'rooms are too
small', '22-NOV-2021');
INSERT INTO viewing
    VALUES ('v3', 's5', '7', 'p6', 'tenant', 'declined', 'bad cleaning
situation', '22-NOV-2021');
INSERT INTO viewing
    VALUES ('v4', 's4', '5', 'p2', 'buyer', 'viewed', 'seems good, will
think', '22-NOV-2021');
INSERT INTO viewing
    VALUES ('v5', 's1', '9', 'p1', 'tenant', 'approved', 'not bad to live
for short time', '22-NOV-2021');
```

INSERT INTO rented_out

```
    VALUES ('r2', 'p2', '2', '5', '520', '2000', '400','20-OCT-2020', '22-
JAN-2025', 200);
INSERT INTO rented_out
    VALUES ('r3', 'p3', '3', '6', '650', '2000','400', '28-OCT-2020', '26-
APR-2022', 200);
```

INSERT INTO sold_out

```
    VALUES ('s1', 'p1', '8','2', '177000', '60', '2000', '26-FEB-2019');
INSERT INTO sold_out
    VALUES ('s2', 'p6', '9', '3','900000', '70', '3000', '28-FEB-2020');
INSERT INTO sold_out
    VALUES ('s3', 'p5', '10','4', '120000', '80', '3000', '28-OCT-2019');
```

Output from running the above SQL code for populating your relational database (e.g., contents from Spool file or screenshots, etc)

```
w21040872> @C:\Users\w21040872\Downloads\insertq.sql
w21040872> INSERT INTO customer
  2    VALUES ('1', 'Paul', 'male', '132 kenton', 'kenton lane',
'Newcastle', 'ne42c4', '06-OCT-1978', 'paul@yahoo.com', '07700 900521');

1 row created.

w21040872> INSERT INTO customer
  2    VALUES ('2', 'Allen', 'male', '139 kenton', 'kenton
lane','Newcastle','ne42c4', '06-OCT-1970', 'allen@yahoo.com', '0115 496
0529');

1 row created.

w21040872> INSERT INTO customer
  2    VALUES ('3', 'Teddy',  'female', '161 fenham',
'fenham','Newcastle','ne42c4','06-OCT-1980', 'teddy@gmail.com', '0117 496
0816');

1 row created.

w21040872> INSERT INTO customer
  2    VALUES ('4', 'Mark',  'male', '121 fawdon',
'fawdon','Newcastle','ne42c4', '06-OCT-1987', 'mark@hotmail.com', '0787 899
2166');

1 row created.

w21040872> INSERT INTO customer
  2    VALUES ('5', 'Mouzal',  'female', '167 kenton', 'kenton
lane','Newcastle','ne42c4','06-OCT-1983', 'maouzal@yahoo.com', '0115 496
0299');

1 row created.

w21040872> INSERT INTO customer
  2    VALUES ('6', 'David', 'male', '132 freman',
'freman','Newcastle','ne42c4', '06-OCT-1991', 'david@gmail.com',  '0151 496
0220');

1 row created.

w21040872> INSERT INTO customer
  2    VALUES ('7', 'Ameer', 'male', '154 fenham', 'fenham',
'Newcastle','ne42c4', '06-OCT-1976', 'ameer@hotmail.com',  '020 7946
0870');

1 row created.

w21040872> INSERT INTO customer
  2    VALUES ('8', 'Naba',  'female', '152 kenton', 'kenton lane',
'Newcastle','ne42c4','06-OCT-1972',  'naba@gmail.com','0118 496 0496');

1 row created.

w21040872> INSERT INTO customer
```

```
  2    VALUES ('9', 'Hiba', 'female', '112 fawdon', 'fawdon',
'Newcastle','ne42c4', '06-OCT-1993', 'hiba@yahoo.com', '0118 496 0286');

1 row created.

w21040872> INSERT INTO customer
  2    VALUES ('10', 'Smith', 'male', '102 kingston', 'kingston
park','Newcastle','ne42c4', '23-FEB-2009', 'smith@yahoo.com',  '0114 496
0550');

1 row created.

w21040872>
w21040872> INSERT INTO owner
  2    VALUES ('1');

1 row created.

w21040872> INSERT INTO owner
  2    VALUES ('2');

1 row created.

w21040872> INSERT INTO owner
  2    VALUES ('3');

1 row created.

w21040872> INSERT INTO owner
  2    VALUES ('4');

1 row created.

w21040872>
w21040872> INSERT INTO buyer
  2    VALUES ('8');

1 row created.

w21040872> INSERT INTO buyer
  2    VALUES ('9');

1 row created.

w21040872> INSERT INTO buyer
  2    VALUES ('10');

1 row created.

w21040872>
w21040872> INSERT INTO tenant
  2    VALUES ('5');

1 row created.

w21040872> INSERT INTO tenant
  2    VALUES ('6');

1 row created.

w21040872> INSERT INTO tenant
```

```
  2    VALUES ('7');

1 row created.

w21040872>
w21040872> INSERT INTO real_estate_agent
  2    VALUES ('a1', 'Joli', 'Newcastle','joli.com','132 kenton', 'kenton
lane', 'Newcastle', 'ne42c4', '07878952166','joli@gmail.com');

1 row created.

w21040872> INSERT INTO real_estate_agent
  2    VALUES ('a2', 'Ali', 'London','ali.com','139 kenton', 'kenton
lane','Newcastle','ne42c4','07878952167','ali@yahoo.com');

1 row created.

w21040872> INSERT INTO real_estate_agent
  2    VALUES ('a3', 'Nadia','Manchester', 'nadia.com','161 fenham',
'fenham','Newcastle','ne42c4', '07878952168','nadia@gmail.com');

1 row created.

w21040872> INSERT INTO real_estate_agent
  2    VALUES ('a4', 'Ruby','London', 'ruby.com','121 fawdon',
'fawdon','Newcastle','ne42c4', '07878954166','ruby@hotmail.com');

1 row created.

w21040872> INSERT INTO real_estate_agent
  2    VALUES ('a5', 'Tommy', 'Derby','tommy.com','152 kenton', 'kenton
lane', 'Newcastle','ne42c4', '07878952166','tommy@hotmail.com');

1 row created.

w21040872> INSERT INTO real_estate_agent
  2    VALUES ('a6', 'Joel', 'Bristol','joel.com','102 kingston', 'kingston
park','Newcastle','ne42c4', '07878952166','joel@yahoo.com');

1 row created.

w21040872> INSERT INTO real_estate_agent
  2    VALUES ('a7', 'iffi', 'Scotland','iffi.com','112 fawdon', 'fawdon',
'Newcastle','ne42c4','07878952166', 'iffi@yahoo.com');

1 row created.

w21040872>
w21040872>
w21040872> INSERT INTO branch
  2    VALUES ('b1', '132 kenton', 'kenton lane', 'Newcastle',
'ne42c4','David','a1');

1 row created.

w21040872> INSERT INTO branch
  2    VALUES ('b2', '152 kenton', 'kenton lane',
'Newcastle','ne42c4','Parker','a2');

1 row created.
```

```
w21040872> INSERT INTO branch
  2    VALUES ('b3', '121 fawdon',
'fawdon','Newcastle','ne42c4','Paul','a3');

1 row created.

w21040872> INSERT INTO branch
  2    VALUES ('b4', '141 fenham',
'fenham','Newcastle','ne42c4','Darwen','a4');

1 row created.

w21040872> INSERT INTO branch
  2    VALUES ('b5', '139 kenton', 'kenton
lane','Newcastle','ne42c4','Smith','a5');

1 row created.

w21040872> INSERT INTO branch
  2    VALUES ('b6', '161 fenham',
'fenham','Newcastle','ne42c4','Ellishy','a6');

1 row created.

w21040872> INSERT INTO branch
  2    VALUES ('b7', '119 kenton', 'kenton
lane','Newcastle','ne42c4','Kurtal','a7');

1 row created.

w21040872>
w21040872>
w21040872> INSERT INTO branch_contact
  2    VALUES ('b1', '0780562', 'iffi@yahoo.com');

1 row created.

w21040872> INSERT INTO branch_contact
  2    VALUES ('b2', '078756', 'tommy@hotmail.com');

1 row created.

w21040872> INSERT INTO branch_contact
  2    VALUES ('b3', '078556', 'ruby@hotmail.com');

1 row created.

w21040872> INSERT INTO branch_contact
  2    VALUES ('b4', '078556', 'joel@yahoo.com');

1 row created.

w21040872> INSERT INTO branch_contact
  2    VALUES ('b5', '078786', 'jackob@yahoo.com');

1 row created.

w21040872>
w21040872>
w21040872> INSERT INTO staff
```

```
  2    VALUES ('s1', 'b1', 'Hanna', 'Brimingham England',
'hanna@yahoo.com', '07855678556');

1 row created.

w21040872> INSERT INTO staff
  2    VALUES ('s2', 'b2', 'Zoal', 'Coventry England', 'zoal@yahoo.com',
'07984565556');

1 row created.

w21040872> INSERT INTO staff
  2    VALUES ('s3', 'b3', 'Jimmy', 'Newcastle England',
'jimmy@hotmail.com', '07987655556');

1 row created.

w21040872> INSERT INTO staff
  2    VALUES ('s4', 'b4', 'Kulsam','Edinburg Scotland',
'kulsam@yahoo.com', '07098765556');

1 row created.

w21040872> INSERT INTO staff
  2    VALUES ('s5', 'b5', 'Smoul', 'Manchester England',
'smoul@hotmail.com', '07234567896');

1 row created.

w21040872> INSERT INTO staff
  2    VALUES ('s6', 'b6', 'Joe', 'Dundee Scotland','zoe@gmail.com',
'07876745676');

1 row created.

w21040872> INSERT INTO staff
  2    VALUES ('s7', 'b7', 'Amal', 'Swansea Wales','amal@gmail.com',
'07987654336');

1 row created.

w21040872>
w21040872>
w21040872> INSERT INTO property
  2    VALUES ('p1', '1', 'b1', '132 Jesmond', 'Jesmond', 'Newcastle',
'ne42c4','40.6 square meters', '3 bedrooms', '3','null', '125000','semi-
detached', 'for_sale', '19-NOV-2021');

1 row created.

w21040872> INSERT INTO property
  2    VALUES ('p2', '2', 'b2','119 Jesmond',
'Jesmond','Newcastle','ne42c4','27.6 square meters','2 bedrooms',
'2','null','290000','semi-detached', 'sold_out', '27-FEB-2021');

1 row created.

w21040872> INSERT INTO property
  2    VALUES ('p3', '2', 'b3', '161 fenham',
'fenham','Newcastle','ne42c4','20.6 square meters','1
bedroom','1','null','160000', 'apartment', 'for_sale', '21-NOV-2020');
```

```
1 row created.

w21040872> INSERT INTO property
  2    VALUES ('p4', '2', 'b4', '161 fawdon',
'fawdon','Newcastle','ne42c4','45.6 square meters','2 bedrooms 1
master','2', '625','null', 'detached', 'rented_out', '26-OCT-2020');

1 row created.

w21040872> INSERT INTO property
  2    VALUES ('p5', '1', 'b5', '132 freman',
'freman','Newcastle','ne42c4','48.6 square meters', '3
bedrooms','3','null','145000','detached', 'for_sale', '22-NOV-2021');

1 row created.

w21040872> INSERT INTO property
  2    VALUES ('p6', '1', 'b6', '121 Jesmond',
'Jesmond','Newcastle','ne42c4','60.6 square meters','5 bedrooms',
'5','600', null, 'semi-detached', 'for_rent', '28-OCT-2021');

1 row created.

w21040872> INSERT INTO property
  2    VALUES ('p7', '3', 'b7','152 Jesmond', 'Jesmond',
'Newcastle','ne42c4','41.6 square meters', '3
bedrooms','3','null','110900','semi-detached', 'for_sale', '21-NOV-2021');

1 row created.

w21040872>
w21040872>
w21040872> INSERT INTO viewing
  2    VALUES ('v1', 's1', '1', 'p5', 'buyer', 'approved', 'best house i
was looking for', '22-NOV-2021');

1 row created.

w21040872> INSERT INTO viewing
  2    VALUES ('v2', 's2', '3', 'p7', 'buyer', 'viewed', 'rooms are too
small', '22-NOV-2021');

1 row created.

w21040872> INSERT INTO viewing
  2    VALUES ('v3', 's3', '7', 'p6', 'tenant', 'declined', 'bad cleaning
situation', '22-NOV-2021');

1 row created.

w21040872> INSERT INTO viewing
  2    VALUES ('v4', 's4', '5', 'p2', 'buyer', 'viewed', 'seems good, will
think', '22-NOV-2021');

1 row created.

w21040872> INSERT INTO viewing
  2    VALUES ('v5', 's5', '9', 'p1', 'tenant', 'approved', 'not bad to
live for short time', '22-NOV-2021');
```

```
1 row created.

w21040872>
w21040872>
w21040872> INSERT INTO rented_out
  2    VALUES ('r2', 'p2', '2', '5', '520', '2000', '400','20-OCT-2020',
'22-JAN-2025', 200);

1 row created.

w21040872> INSERT INTO rented_out
  2    VALUES ('r3', 'p3', '3', '6', '650', '2000','400', '28-OCT-2020',
'26-APR-2022', 200);

1 row created.

w21040872>
w21040872>
w21040872>
w21040872> INSERT INTO sold_out
  2    VALUES ('s1', 'p1', '8','2', '177000', '60', '2000', '26-FEB-2019');

1 row created.

w21040872> INSERT INTO sold_out
  2    VALUES ('s2', 'p6', '9', '3','900000', '70', '3000', '28-FEB-2020');

1 row created.

w21040872> INSERT INTO sold_out
  2    VALUES ('s3', 'p5', '10','4', '120000', '80', '3000', '28-OCT-
2019');

1 row created.
w21040872> spool off
```

Relational Algebra and SQL.

Display details of *semi-detached* properties for sale having at least three bedrooms in the *Jesmond* area of Newcastle upon Tyne that were added to the system in the last 14 days.

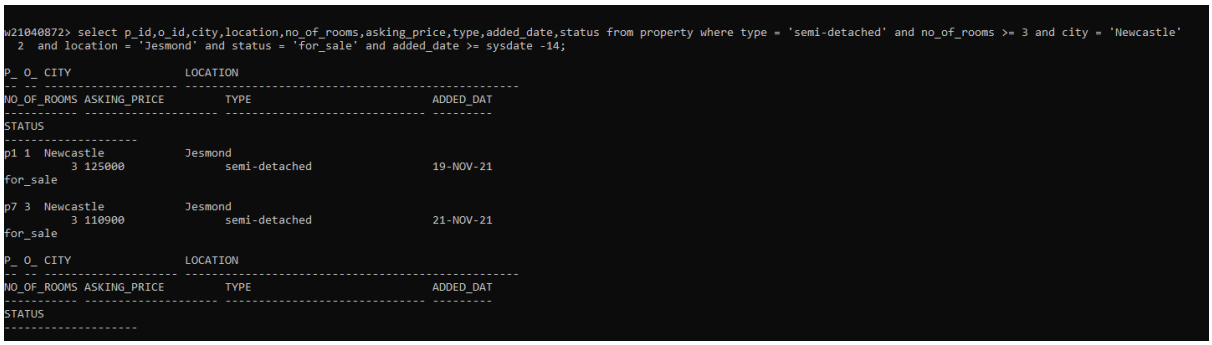
Relational Algebra expression

```
π p_id, o_id, city, location, no_of_rooms, asking_price, type, added_date,
status
 σ type = "semi-detached" AND no_of_rooms >= 3 AND city = "Newcastle" AND
location = "Jesmond" AND status = "for_sale" AND added_date >= sysdate - 14
property
```

SQL query code and output

```
Select
p_id,o_id,city,location,no_of_rooms,asking_price,type,added_date,status
from property where type = 'semi-detached' and no_of_rooms >= 3 and city =
'Newcastle'
```

```
and location = 'Jesmond' and status = 'for_sale' and added_date >= sysdate
-14;
```



q1) Display details of properties sold in Newcastle, Sunderland, Gateshead or Durham for £157,000 to £279,000 in the years 2019 or 2020.

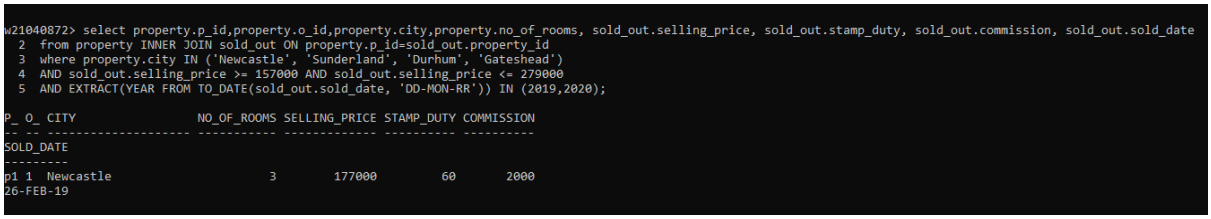
Relational Algebra expression

π property . p_id, property . o_id, property . city, property . no_of_rooms, sold_out . selling_price, sold_out . stamp_duty, sold_out . commission, sold_out . sold_date

σ property . city = "Newcastle" OR property . city = "Sunderland" OR property . city = "Durhum" OR property . city = "Gateshead" AND sold_out . selling_price >= 157000 AND sold_out . selling_price <= 279000 (property ⋈ property . p_id = sold_out . property_id sold_out)

SQL query code and output

```
Select property.p_id,property.o_id,property.city,property.no_of_rooms,
sold_out.selling_price, sold_out.stamp_duty, sold_out.commission,
sold_out.sold_date
from property INNER JOIN sold_out ON property.p_id=sold_out.property_id
where property.city IN ('Newcastle', 'Sunderland', 'Durhum', 'Gateshead')
AND sold_out.selling_price >= 157000 AND sold_out.selling_price <= 279000
AND EXTRACT(YEAR FROM TO_DATE(sold_out.sold_date, 'DD-MON-RR')) IN
(2019,2020);
```



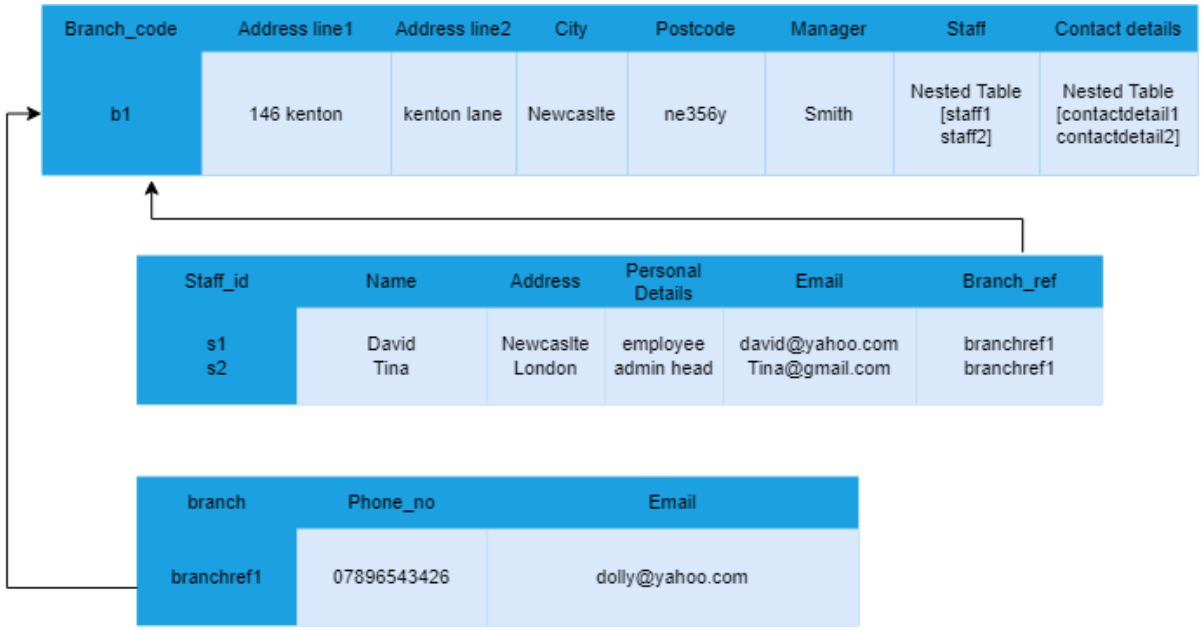
This part is based on your answer / solution to Part 1 (A), i.e., conceptual design of the database for the MOVEHOME scenario.

Choose and justify what aspects of MOVEHOME conceptual design would be better off if implemented using object-relational database; then provide logical design and implementation of the subset of the MOVEHOME using ER/EER to object-relational mapping and object-relational features of Oracle Database System (Kannan); populate the object-tables with sample data and demonstrate your choice of design and implementation by running two complex queries on your object-tables.

1) **Choice and justification of what aspects (subset) of the MOVEHOME conceptual design from Part 1.A you would like to implement using object relational databases**

   I choose branch entity to implement object relational database it will allow inheritance of staff and contact details directly from branch and use in database schemas. It will helpful to run queries using the branch object.

2) **Logical design for your chosen subset using ER/EER to object-relational mapping (2 marks)**



SQL code and output for implementing your above logical object-relational design

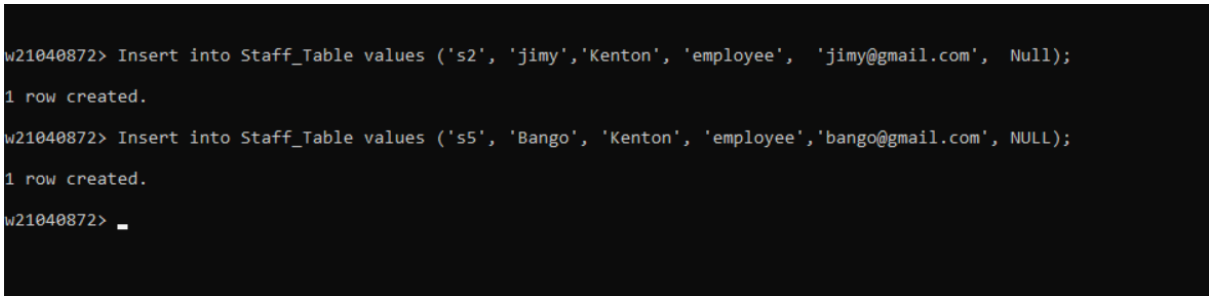
```
CREATE Type branch_type
/
```

```
CREATE Type staff_type AS Object
 (s_id CHAR(2),
  name VARCHAR(50),
  address VARCHAR(50),
  personal_details VARCHAR(50),
  email VARCHAR(20),
  branch_ref REF branch_type
 )
/
CREATE Type branch_contact_type AS Object
 (
  email VARCHAR(50),
  phone_number VARCHAR(15),
  branch_ref REF branch_type
 )
/
CREATE TYPE staff_nt_type as Table of REF staff_type
/
CREATE TYPE contact_nt_type as Table of REF branch_contact_type
/
CREATE OR Replace Type branch_type as Object
 (branch_code CHAR(2),
  address_line1 VARCHAR(50),
  address_line2 VARCHAR(50),
  city VARCHAR(20),
  postcode VARCHAR(10),
  manager VARCHAR(20),
  staff staff_nt_type,
  contact_details contact_nt_type
)
/
Create Table branch_Table of branch_type
(primary key (branch_code)
)
Nested Table staff Store As branch_staff_NTab,
Nested Table contact_details Store As branch_contact_NTab
/
Create Table branch_contact_Table of branch_contact_type
(
foreign key (branch_ref) References branch_Table)
/
```

```
Create Table staff_Table of staff_type
(primary key (s_id),
  foreign key (branch_ref) References branch_Table
)/
```

SQL code and output for populating your above object-relational subset of the MOVEHOME database

```
w21040872 >Insert into Staff_Table values ('s1',
'Ali','Kingston', 'employee', 'ali@gmail.com', b1);

1 row created.

w21040872> Insert into Staff_Table values ('s2',
'jimy','Kenton', 'employee', 'jimy@gmail.com', Null);

1 row created.

w21040872> Insert into Staff_Table values ('s5', 'Bango',
'Kenton', 'employee','bango@gmail.com', NULL);

1 row created.

w21040872>
```




```
w21040872 >Insert into Branch_Table values ('b1', '132 kenton', 'kenton
lane', 'Newcastle', 'ne42c4','David', Staff_nt_type(), Null);

1 row created.

w21040872 >Insert into Branch_Table values ('b2', '152 kenton', 'kenton
lane', 'Newcastle','ne42c4','Parker', Staff_nt_type(), Null);
```

1 row created.

w21040872 >Insert into Branch_Table values ('b3', '121 fawdon', 'fawdon','Newcastle','ne42c4','Paul', Staff_nt_type(), null);

1 row created.

w21040872 >Insert into Branch_Table values ('b4', '141 fenham', 'fenham','Newcastle','ne42c4','Darwen', Staff_nt_type(), null);

1 row created.

```
w21040872> Insert into Branch_Table values ('b2', '152 kenton', 'kenton lane', 'Newcastle','ne42c4','Parker', Staff_nt_t
ype(), Null);

1 row created.

w21040872> Insert into Branch_Table values ('b3', '121 fawdon', 'fawdon','Newcastle','ne42c4','Paul', Staff_nt_type(), n
ull);

1 row created.

w21040872> Insert into Branch_Tab values ('b4', '141 fenham', 'fenham','Newcastle','ne42c4','Darwen',, Staff_nt_type(),
null);
Insert into Branch_Tab values ('b4', '141 fenham', 'fenham','Newcastle','ne42c4','Darwen',, Staff_nt_type(), null)
                                                                                          *
ERROR at line 1:
ORA-00936: missing expression


w21040872> Insert into Branch_Table values ('b4', '141 fenham', 'fenham','Newcastle','ne42c4','Darwen', Staff_nt_type(),
 null);

1 row created.

w21040872>
```

SQL code and output for running two complex queries on the object-relational subset of the above MOVEHOME database

```
w21040872 >Update Staff_Table
  2  Set
  3   Works_for = (select ref(b) from Branch_Table b where b.code = 'b1')
where staff_id = 's2';

1 row updated.
w21040872 >INSERT INTO TABLE (Select b.Staff from Branch_Table b where
branch.code = 'b1')
  2  SELECT REF(e) from Staff_Table e where e.works_for.code = 's3';

2 rows created.
```

(B) Analyse the conceptual database design from Part 1 (A) and the MOVEHOME scenario in the Appendix and propose what aspects of the MOVEHOME database would benefit from incorporating NoSQL Database concepts. Illustrate your answer with code from a representative code from NoSQL Database implementation.

Provide below your choice and justification of what aspects (subset) of the MOVEHOME databases would benefit from incorporating NoSQL Database concepts (3 marks)

Branch and property both tables can incorporate into NOSQL Database.
Branch table has staff and branch contact details and it will be beneficial to make an array in for staff and contact details in single collection named as branch. It will be easy to access, update and run different operations and will be time effective to find information about staff, branch contact and branch. Store unstructured, semi-structured, or structured data.

Similarly, property may have more sub-entities such as sold_out and rented_out, images of property. It will be advantageous to implement it in NoSQL. Firstly No SQL is popular to store structured, unstructured and semi-structured data. Secondly, we don't need to make more schemas and fields for sub-entities, it can be updated in the property collection

Provide below code and output for implementing your proposed NoSQL Database subset of the MOVEHOME database, populate it with some data, and example queries & outputs (12 Marks)

```
db.createCollection("property");
db.property.insert(
{ p_id: "p1",
  o_id: 'o1',
  branch_id: "b1",
  address: '132 Jesmond',
  location: 'Jesmond',
  city:'Newcastle',
  postcode: 'ne32d4',
  accomodation_size: '40.6 square meters',
  rooms_description: '3 bedrooms',
  no_of_rooms: '3',
  rental_demand: 'null',
  asking_price: '125000',
  type:'semi-detached',
  status: 'for_sale',
  added_date: ISODate('2021-11-21')});

db.property.insert(
{ p_id: "p2",
  o_id: 'o2',
  branch_id: "b2",
  address: '132 freeman',
  location: 'freeman',
  city:'Newcastle',
  postcode: 'n256d4',
  accomodation_size: '36.6 square meters',
  rooms_description: '2 bedrooms',
  no_of_rooms: '2',
  rental_demand: '600',
  asking_price: 'null',
  type:'detached',
  status: 'for_rent',
  added_date: ISODate('2021-11-21')});

db.property.insert(
{ p_id: "p3",
  o_id: 'o3',
  branch_id: "b3",
  address: '146 Jesmond',
  location: 'Jesmond',
  city:'Newcastle',
  postcode: 'ne42d4',
  accomodation_size: '20.6 square meters',
  rooms_description: '1 bedrooms',
  no_of_rooms: '1',
  rental_demand: 'null',
  asking_price: '125000',
  type:'house',
  status: 'for_sale',
  added_date: ISODate('2021-11-15')});



db.createCollection("branch");
db.branch.insert(
```

```
{ b_id: "b1",
  agent_id: "a1",
  address_line1: '132 Kenton',
  address_line2: 'Kenton lane',
  city:'Newcastle',
  postcode: 'ne32c4',
  manager: "David",
  staff:
  [ {s_id: "s1", name: 'Hanna', address: 'Brimingham England',
personal_details: 'staff head',email: 'hanna@yahoo.com'},
    {s_id: "s2", name: 'Zoal', address: 'Coventry England',
personal_details: 'admin', email: 'zoal@yahoo.com'},
    {s_id: "s3", name: 'Jimmy', address: 'Newcastle England',
personal_details: 'employee', email: 'jimmy@hotmail.com'}],
  branch_contact:
  [ {phone_number: "07952475106",  email: 'jimmy@gmail.com'} ],
});

db.branch.insert(
{ b_id: "b2",
  agent_id: "a2",
  address_line1: '132 fenham',
  address_line2: 'fenham',
  city:'Newcastle',
  postcode: 'ne3sd4',
  manager: "Smith",
  staff:
  [ {s_id: "s4", name: 'Hania', address: 'Manchester England',
personal_details: 'employee',email: 'hania@yahoo.com'},
    {s_id: "s5", name: 'Jimican', address: 'Newcastle England',
personal_details: 'employee', email: 'jimica@hotmail.com'}],
  branch_contact:
  [ {phone_number: "07957896506",  email: 'jimina@gmail.com'} ],
});

db.branch.insert(
{ b_id: "b3",
  agent_id: "a3",
  address_line1: '135 fawdon',
  address_line2: 'fawdon',
  city:'Newcastle',
  postcode: 'ne97c4',
  manager: "Dongle",
  staff:
  [{s_id: "s6", name: 'Amica', address: 'Newcastle England',
personal_details: 'employee', email: 'amica@hotmail.com'}],
  branch_contact:
  [ {phone_number: "07945675116",  email: 'ewcas@gmail.com'} ],
});
```

Data in NoSQL Collections

```
> db.branch.find();
{ "_id" : ObjectId("619e8b18e5c96de721afd6f7"), "b_id" : "b1", "agent_id" :
"a1", "address_line1" : "132 Kenton", "address_line2" : "Kenton lane",
"city" : "Newcastle", "postcode" : "ne32c4", "manager" : "David", "staff" :
[ { "s_id" : "s1", "name" : "Hanna", "address" : "Brimingham England",
"personal_details" : "staff head", "email" : "hanna@yahoo.com" }, { "s_id"
: "s2", "name" : "Zoal", "address" : "Coventry England", "personal_details"
: "admin", "email" : "zoal@yahoo.com" }, { "s_id" : "s3", "name" : "Jimmy",
```

```
"address" : "Newcastle England", "personal_details" : "employee", "email" :
"jimmy@hotmail.com" } ], "branch_contact" : [ { "phone_number" :
"07952475106", "email" : "jimmy@gmail.com" } ] }
{ "_id" : ObjectId("619e8b2be5c96de721afd6f8"), "b_id" : "b2", "agent_id" :
"a2", "address_line1" : "132 fenham", "address_line2" : "fenham", "city" :
"Newcastle", "postcode" : "ne3sd4", "manager" : "Smith", "staff" : [ {
"s_id" : "s4", "name" : "Hania", "address" : "Manchester England",
"personal_details" : "employee", "email" : "hania@yahoo.com" }, { "s_id" :
"s5", "name" : "Jimican", "address" : "Newcastle England",
"personal_details" : "employee", "email" : "jimica@hotmail.com" } ],
"branch_contact" : [ { "phone_number" : "07957896506", "email" :
"jimina@gmail.com" } ] }
{ "_id" : ObjectId("619e8b46e5c96de721afd6f9"), "b_id" : "b3", "agent_id" :
"a3", "address_line1" : "135 fawdon", "address_line2" : "fawdon", "city" :
"Newcastle", "postcode" : "ne97c4", "manager" : "Dongle", "staff" : [ {
"s_id" : "s6", "name" : "Amica", "address" : "Newcastle England",
"personal_details" : "employee", "email" : "amica@hotmail.com" } ],
"branch_contact" : [ { "phone_number" : "07945675116", "email" :
"ewcas@gmail.com" } ] }

> db.property.find();
{ "_id" : ObjectId("619e8899e5c96de721afd6f4"), "p_id" : "p1", "o_id" :
"o1", "branch_id" : "b1", "address" : "132 Jesmond", "location" :
"Jesmond", "city" : "Newcastle", "postcode" : "ne32d4", "accomodation_size"
: "40.6 square meters", "rooms_description" : "3 bedrooms", "no_of_rooms" :
"3", "rental_demand" : "null", "asking_price" : "125000", "type" : "semi-
detached", "status" : "for_sale", "added_date" : ISODate("2021-11-
21T00:00:00Z") }
{ "_id" : ObjectId("619e891de5c96de721afd6f5"), "p_id" : "p2", "o_id" :
"o2", "branch_id" : "b2", "address" : "132 freeman", "location" :
"freeman", "city" : "Newcastle", "postcode" : "n256d4", "accomodation_size"
: "36.6 square meters", "rooms_description" : "2 bedrooms", "no_of_rooms" :
"2", "rental_demand" : "600", "asking_price" : "null", "type" : "detached",
"status" : "for_rent", "added_date" : ISODate("2021-11-21T00:00:00Z") }
{ "_id" : ObjectId("619e893ce5c96de721afd6f6"), "p_id" : "p3", "o_id" :
"o3", "branch_id" : "b3", "address" : "146 Jesmond", "location" :
"Jesmond", "city" : "Newcastle", "postcode" : "ne42d4", "accomodation_size"
: "20.6 square meters", "rooms_description" : "1 bedrooms", "no_of_rooms" :
"1", "rental_demand" : "null", "asking_price" : "125000", "type" : "house",
"status" : "for_sale", "added_date" : ISODate("2021-11-15T00:00:00Z") }
>
```


Example Queries and its Output

Display details of properties sold in Newcastle and area must be Jesmond from 2019 to 2020

```
db.property.find(
     {
```

```
     status: 'for_sale', location: 'Jesmond', city: 'Newcastle', type:
'semi-detached',
     'added_date': { $gt: new Date('2020-01-01'), $lt: new Date('2022-01-
01') }
     } );
```

Output:

```
{ "_id" : ObjectId("619e8899e5c96de721afd6f4"), "p_id" : "p1", "o_id" :
"o1", "branch_id" : "b1", "address" : "132 Jesmond", "location" :
"Jesmond", "city" : "Newcastle", "postcode" : "ne32d4", "accomodation_size"
: "40.6 square meters", "rooms_description" : "3 bedrooms", "no_of_rooms" :
"3", "rental_demand" : "null", "asking_price" : "125000", "type" : "semi-
detached", "status" : "for_sale", "added_date" : ISODate("2021-11-
21T00:00:00Z") }
>
```

```
Display all staff members which are under Newcastle city
db.branch.find(
...      {
...      city: 'Newcastle'
...      },
...      {staff: 1, city: 1}
...       );
```

Output:

```
{ "_id" : ObjectId("619e8b18e5c96de721afd6f7"), "city" : "Newcastle",
"staff" : [ { "s_id" : "s1", "name" : "Hanna", "address" : "Brimingham
England", "personal_details" : "staff head", "email" : "hanna@yahoo.com" },
{ "s_id" : "s2", "name" : "Zoal", "address" : "Coventry England",
"personal_details" : "admin", "email" : "zoal@yahoo.com" }, { "s_id" :
"s3", "name" : "Jimmy", "address" : "Newcastle England", "personal_details"
: "employee", "email" : "jimmy@hotmail.com" } ] }
{ "_id" : ObjectId("619e8b2be5c96de721afd6f8"), "city" : "Newcastle",
"staff" : [ { "s_id" : "s4", "name" : "Hania", "address" : "Manchester
England", "personal_details" : "employee", "email" : "hania@yahoo.com" }, {
"s_id" : "s5", "name" : "Jimican", "address" : "Newcastle England",
"personal_details" : "employee", "email" : "jimica@hotmail.com" } ] }
{ "_id" : ObjectId("619e8b46e5c96de721afd6f9"), "city" : "Newcastle",
"staff" : [ { "s_id" : "s6", "name" : "Amica", "address" : "Newcastle
England", "personal_details" : "employee", "email" : "amica@hotmail.com" }
] }
```

Consider the MOVEHOME scenario in the Appendix. Produce a report for the managing director elaborating on professional, legal, ethical and security issues that need to be considered and make recommendations that you think are appropriate for MOVEHOME.

Part 4

Databases are necessary part in many areas, such as scientific research, IT development, bioinformatics, law enforcement organizations, e-commerce, and government agencies. As an important part of whole process, databases legal, ethical and security issues are increasing (Goguen, 1999). This report intend to elaborate the legal, ethical, and, social issues need to be highlight, address and to determine the relevant guidelines and recommendations for addressing these problems in the MOVEHOME system.

Legal, ethical and security issues

There are numerous ethical and security issues when collecting, storing, and accessing the data from databases. Organizations stores plenty of information about their customers in their databases, and to protect that information in critical and complex.

Privacy, security and trust all are interrelated, similarly as are ethics and law. Maintain the privacy and security depends on trust, for example, only the trustable individuals will be allowed to access the protected information.

Privacy breaches in database leads to risk the security, disturb trust, dishonor ethical values and violate laws. There are no legal laws to govern how individuals morally behave. Nevertheless, legal and ethical guides/rules must be applied to protect the information collected in databases.

Some of the main points related to data privacy includes:

- Secure from the unauthorized access
- Appropriate use and process of collected data
- Properly collect the complete and accurate data from customers.
- Data ownership and legal to right

Maintain data privacy saves the organizations from financial penalties, help to develop a trustworthy relationship with clients and also impact to build positive reputation of the company. In terms of law and ethics, data breaches violate the law and show disrespect to the ethics. Ethics provides context to implement law because violating

law means substance while ethics are more about moral values (Lee, et al., 2016). Data privacy or protection of data is securely accessing the data collection and secure the database from unauthorized access. Relevant ethical, social and legal measures and standards need to be considered for MOVEHOME system while collection and processing of data to make the system transparent, secure and reliable for customers.

Technical and social solutions

There are different ways to protect the data using social and technical measures as well as data protection laws. Technical measures are from organization itself which include to secure data from unauthorized access, respect the user's confidential date and maintain the reliability of information. Social solutions are with respect to the customer's awareness about their own information that why their data is stored, and how they can help the organization to secure their personal and sensitive information. Companies can spread awareness among their users related to the data security rules and instruct them that how customers can help to maintain the data privacy.

Standards

To adopt the standards to protect the data within and outside the organization. Relevant standards must be considered while collecting, processing and presenting the client's information. Introduce and implement international standard for database design and access can help the MOVEHOME system such as:

- Prevent unauthorized access of database.
- Maintain data transparency through legal privacy agreements with customers.
- Ensure the protection of copyrights
- Address data protection and security concerns.
- Implement the concept of consent to use the customer's data.
- Keep proper backups of data to avoid any misconduct and lose of information.
- Learn international laws and policies while expanding worldwide.
- Protect Intellectual Property Rights.

Database Security

An extensive database management system security needs to design to protect against the breaches, malicious and cyber-attacks (e.g., Spyware, Trojan, DOS, virus,

malware, and adware). SQL injection must be an important concern as it is a code injection technique which used to extract data from different applications. Through SQL queries, attacker gain unauthorized access to database and retrieve the sensitive and valuable information from database (e.g. Bank accounts, card numbers, and transactions) (Devi, et al., 2016).

Few following recommendations must be adopted for MOVEHOME to maintain the credibility and reliability of system in future:

- Recommend and allow users to create strong credentials (e.g. Usernames and passwords).
- Delete the unused add-ons and extensions
- Install necessary modules only and avoid complex components.
- Improve the security through updates and fixing bugs.
- Implement secure and safe coding practices.
- Protect the server which contains the data.
- Safe the physical machines and devices and provide access to authorized and authenticate users only.
- Manage the security through firewalls, intrusion detection and prevention systems.
- Maintain scheduled backups, encrypt backups and save on secured systems.
- Manage strict business standard including accountability according to the particular role.
- Encrypt the data traffic.
- Auditing and monitoring the database through tools to identify vulnerabilities, few tools have capabilities to prevent as well.
- Maintain a recovery plan which should be documented to minimize cost and time loss.
- Ensure data integrity and implement constraints by validation of correct information.
- Documentation should be maintained to record database histories, logs, reports and updates.
- Train the employees and ensure their understandings about database and cloud database security. (UKEssays., 2017)

References

Devi, R., Venkatesan, R. & Koteeswaran, R., 2016. A study on SQL injection techniques. *International Journal of Pharmacy and Technology*.
Goguen, J. A., 1999. *THE ETHICS OF DATABASES.* [Online]
Available at: https://cseweb.ucsd.edu/~goguen/papers/4s/4s.html
[Accessed 24 11 2021].
Lee, W. W., Zankl, W. & Chang, a. H., 2016. An Ethical Approach to Data Privacy Protection. *ISACA JOURNAL*.
UKEssays., 2017. *Database Management: Law, Ethics and Security.* [Online]
Available at: https://www.ukessays.com/essays/computer-science/database-management-law-ethics-9552.php?vref=1
[Accessed 24 11 2021].