--create address table

CREATE TABLE ADDRESS AS

SELECT \* FROM MonRE.Address;

--view table

SELECT \* FROM ADDRESS;

A screenshot of a cell phone

Description automatically generated

--check missing value

SELECT \* FROM ADDRESS WHERE address\_id IS NULL;

SELECT \* FROM ADDRESS WHERE street IS NULL;

SELECT \* FROM ADDRESS WHERE suburb IS NULL;

SELECT \* FROM ADDRESS WHERE postcode IS NULL;

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated--check duplicated values

SELECT Address\_id, COUNT(\*)

FROM ADDRESS

GROUP BY Address\_id

HAVING COUNT(\*) > 1;

A close up of text on a white background

Description automatically generated

SELECT Street, Suburb, Postcode, COUNT(\*)

FROM ADDRESS

GROUP BY Street, Suburb, Postcode

HAVING COUNT(\*) > 1;

SELECT \* FROM address where street = '5/115 Monument Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '405s/1 Lardelli Drive';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '3 Bright Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '219/41 Chandler Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '321/12 Provan Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '25/201 Goyder Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '13/29-31 State Circle';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '90/15 Coranderrk Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = 'Ann Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '187 Kent Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street LIKE '%Grazier Lane%' AND street LIKE '%dusk%';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '39/19 Howitt Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '30/11 Fawkner Street'; A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '10/14 Osborne Avenue';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '9/44 Duffield Road';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '2 York Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '39 Deerdale Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '96 Cochrane';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '71 Barr Smith Avenue';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '31 Jansz Crescent';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '30 Printers Way';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = 'Felix Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '26b Wakefield Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '8 Napier Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '34/55 Harries Road';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '11/153 George St';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street LIKE '%Grazier Lane%' AND street LIKE '%republic%';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '1/7 Macroberston';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '2 Wordsworth Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '35 Crescent Road';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '5 Spensley Place';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '305/4 Masson Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '29 Cypress Drive';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '15 Balls Lane';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '8 Aurora Close';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '23 Shelley Street';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '18 Park Lane';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '15 Emerald Place';

A screenshot of a cell phone

Description automatically generated

SELECT \* FROM address where street = '183/1 Mouat Street';

A screenshot of a cell phone

Description automatically generated

--delete duplicated addresses

DELETE FROM Address WHERE Address\_Id IN (4547,4434,1484,4969,4715,4948,4976,4701,835,4187,4329, 4645, 4673, 4680, 5742, 840, 3972, 4186, 3567, 3542, 4838, 5043, 5103, 831, 832, 3042, 3225, 3522, 4482, 4647, 1447, 3176, 3544, 4919, 5020, 5172, 5167, 1710, 4338, 4450,4283, 3374, 4904);

--check duplicated address after data cleaning

SELECT Street, Suburb, Postcode, COUNT(\*)

FROM ADDRESS

GROUP BY Street, Suburb, Postcode

HAVING COUNT(\*) > 1;

A screenshot of a cell phone

Description automatically generated

--address table after data cleaning

SELECT \* FROM ADDRESS;

A screenshot of a cell phone

Description automatically generated

-- Create advertisement table

CREATE TABLE ADVERTISEMENT AS

SELECT \* FROM MonRE.Advertisement;

--view table

SELECT \* FROM ADVERTISEMENT;

A screenshot of a cell phone

Description automatically generated

It is clear to see that there are no null values in this advertisement table because this table only have 25 rows of data as shown in the graph. And the advertisement id can be conclude without duplicated value from the graph.

--check duplicated values of advertisement name

SELECT advert\_name, COUNT(\*)

FROM ADVERTISEMENT

GROUP BY advert\_name

HAVING COUNT(\*) > 1;

A screenshot of a cell phone

Description automatically generated

--create state table

CREATE TABLE STATE AS

SELECT \* FROM MonRE.State;

--view table

SELECT \* FROM STATE;

A screenshot of a cell phone

Description automatically generated

--delete null values

DELETE FROM STATE WHERE State\_code IS NULL;

--check table content after clean

SELECT \* FROM STATE;

A screenshot of a cell phone

Description automatically generated

--create postcode table

CREATE TABLE POSTCODE AS

SELECT \* FROM MonRE.Postcode;

--view table

SELECT \* FROM POSTCODE;

A screenshot of a cell phone

Description automatically generated

--check missing value

SELECT \* FROM POSTCODE WHERE postcode IS NULL;

SELECT \* FROM POSTCODE WHERE state\_code IS NULL;

A close up of a logo

Description automatically generated

--check duplicated values

SELECT postcode, COUNT(\*)

FROM POSTCODE

GROUP BY postcode

HAVING COUNT(\*) > 1;

A close up of a logo

Description automatically generated

--check country code existence

SELECT \* FROM POSTCODE

WHERE state\_code NOT IN (SELECT state\_code from STATE);

A picture containing table

Description automatically generated

--create person table

CREATE TABLE PERSON AS

SELECT \* FROM MonRE.Person;

--view table

SELECT \* FROM PERSON;

A screenshot of a cell phone

Description automatically generated

--check missing value

SELECT \* FROM PERSON WHERE person\_id IS NULL;

SELECT \* FROM PERSON WHERE title IS NULL;

SELECT \* FROM PERSON WHERE first\_name IS NULL;

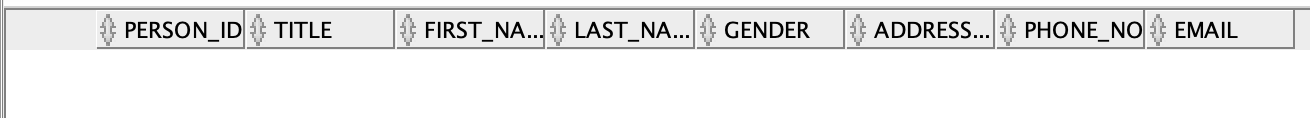
SELECT \* FROM PERSON WHERE last\_name IS NULL;

SELECT \* FROM PERSON WHERE gender IS NULL;

SELECT \* FROM PERSON WHERE address\_id IS NULL;

SELECT \* FROM PERSON WHERE phone\_no IS NULL;

SELECT \* FROM PERSON WHERE email IS NULL;



--check duplicated person id

SELECT person\_id, COUNT(\*)

FROM PERSON

GROUP BY person\_id

HAVING COUNT(\*) > 1;

A screenshot of a cell phone

Description automatically generated

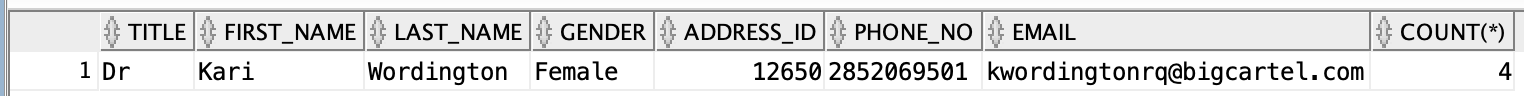
--check duplicated person

SELECT title, first\_name, last\_name, gender, address\_id, phone\_no, email, COUNT(\*)

FROM PERSON

GROUP BY title, first\_name, last\_name, gender, address\_id, phone\_no, email

HAVING COUNT(\*) > 1;



SELECT \* FROM PERSON WHERE person\_id = 6995;

A screenshot of a cell phone

Description automatically generated

--delete duplicated rows

DELETE FROM PERSON WHERE person\_id = 6995;

INSERT INTO PERSON VALUES(6995, 'Dr', 'Kari', 'Wordngton', 'Female', '12650', '2852069501', 'kwordingtonrq@bigcartel.com');

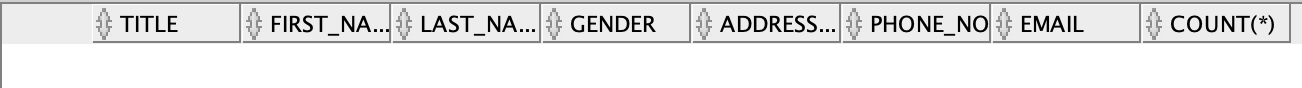
--check duplicated person after data cleanning

SELECT title, first\_name, last\_name, gender, address\_id, phone\_no, email, COUNT(\*)

FROM PERSON

GROUP BY title, first\_name, last\_name, gender, address\_id, phone\_no, email

HAVING COUNT(\*) > 1;



--check address id existence

SELECT \* FROM PERSON

WHERE address\_id NOT IN (SELECT address\_id from ADDRESS);



--delete dirty data

DELETE FROM PERSON WHERE person\_id = 7001;

--create agent table

CREATE TABLE AGENT AS

SELECT \* FROM MonRE.Agent;

--view table

SELECT \* FROM AGENT;

A screenshot of a cell phone

Description automatically generated

--check missing value

SELECT \* FROM AGENT WHERE person\_id IS NULL;

SELECT \* FROM AGENT WHERE salary IS NULL;

A screenshot of a cell phone

Description automatically generated

--check person id existence

SELECT \* FROM AGENT

WHERE person\_id NOT IN (SELECT person\_id from PERSON);

A screenshot of a cell phone

Description automatically generated

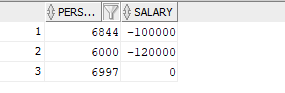
--delete dirty data

DELETE FROM AGENT WHERE person\_id = 6997;

--check salary illegal values

SELECT \* FROM AGENT

WHERE salary <=0;



It is likely that the data is entered wrong with a minus symbol in front of the number, therefore, the number of changed to positive instead of deleting the entire row.

--clean dirty data

UPDATE AGENT

SET salary = 100000

WHERE person\_id = 6844;

UPDATE AGENT

SET salary = 120000

WHERE person\_id = 6000;

--create office table

CREATE TABLE OFFICE AS

SELECT \* FROM MonRE.Office;

--check duplicated rows

SELECT person\_id, salary, COUNT(\*)

FROM AGENT

GROUP BY person\_id, salary

HAVING COUNT(\*) > 1;

A close up of a logo

Description automatically generated

--view table

SELECT \* FROM OFFICE;

A screenshot of a cell phone

Description automatically generated

--check missing value

SELECT \* FROM OFFICE WHERE office\_id IS NULL;

SELECT \* FROM OFFICE WHERE office\_name IS NULL;

A screenshot of a cell phone

Description automatically generated

--check duplicated rows

SELECT office\_id, COUNT(\*)

FROM OFFICE

GROUP BY office\_id

HAVING COUNT(\*) > 1;

A close up of a logo

Description automatically generated

SELECT office\_name, COUNT(\*)

FROM OFFICE

GROUP BY office\_name

HAVING COUNT(\*) > 1;

A close up of a logo

Description automatically generated

--create agent\_office table

CREATE TABLE AGENT\_OFFICE AS

SELECT \* FROM MonRE.Agent\_Office;

--view table

SELECT \* FROM AGENT\_OFFICE;

A screenshot of a cell phone

Description automatically generated

--check missing value

SELECT \* FROM AGENT\_OFFICE WHERE office\_id IS NULL;

SELECT \* FROM AGENT\_OFFICE WHERE person\_id IS NULL;

A close up of a logo

Description automatically generated

--check duplicated rows

SELECT person\_id, office\_id, COUNT(\*)

FROM AGENT\_OFFICE

GROUP BY person\_id, office\_id

HAVING COUNT(\*) > 1;

A close up of a logo

Description automatically generated

--check person id existence

SELECT \* FROM AGENT\_OFFICE

WHERE person\_id NOT IN (SELECT person\_id from AGENT);

A screenshot of a cell phone

Description automatically generated

The result shows that person id 6997 is not exist in the agent table, which is the one that has been deleted from agent table because id 6997 is not exist in person table. Therefore, this row of data is deleted as a dirty data.

--delete row

DELETE FROM AGENT\_OFFICE WHERE person\_id = 6997;

--check office id existence

SELECT \* FROM AGENT\_OFFICE

WHERE office\_id NOT IN (SELECT office\_id from OFFICE);

A close up of a logo

Description automatically generated