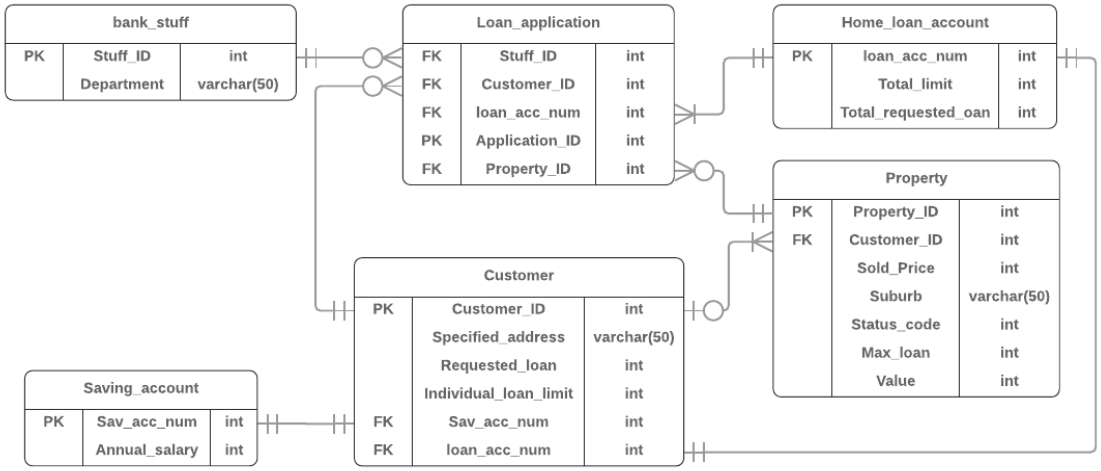
ASSIGNMENT 1A

SIT772

YANJUN CHEN
216327327

Task 1.1



Task 1.2

The digraph meets following requirements:

- Every customer should own a saving account and a property address. They may not own a home loan account yet, or already own their only loan account.
- A stuff may never support an application, but an application must be support by one stuff.
- Each customer has her/his own home loan max limit.
- One customer can apply loan for more than one property.
- One customer can own more than one property.
- A property must have maximum home loan and value.
- A property may not have a buyer yet, or maybe have been involved into loan application more than once since its owner changed.
- One home loan account should link with a customer, or maybe link with many customers.

The digraph meets following transactions:

- A customer can start a loan application with help from a stuff, for one property and on a loan account at once.
- A customer can start many applications on one loan account to allow this account to cover new properties.
- A loan account can be applicated for many times to allowed new customer to join or some new loan to be applied.

The diagram has following limitations:

- It can’t show the relationship between property value and the average of the sold price of the properties located in the same suburb.

- It can’t show that a customer’s requested loan limitation is depends on their total salary in ten years.
- It can’t show how the max loan limitation of a property has been calculated.
- It can’t help customer to request home loan that lower than their loan limitation.
- It can’t show that the home loan limit of a joint account depends on all the joint customers in the home loan
- It can’t show the rule that a customer should own a saving account before own a loan account.

Task 1.3

make1.sql

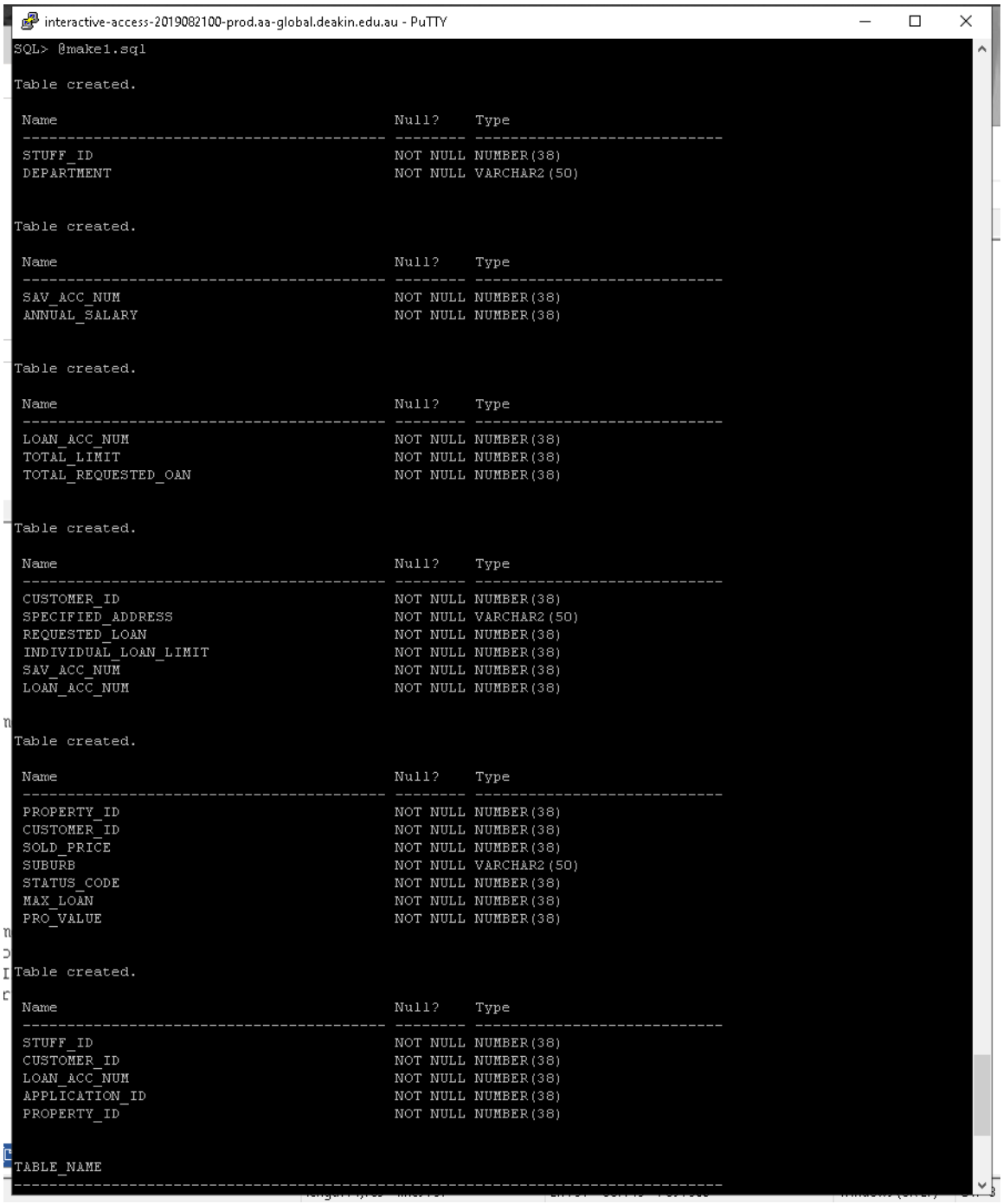
```
1  CREATE TABLE bank_stuff (
2    Stuff_ID int NOT NULL,
3    Department varchar(50) NOT NULL,
4    PRIMARY KEY (Stuff_ID)
5  );
6  DESCRIBE bank_stuff;
7  CREATE TABLE Saving_account (
8    Sav_acc_num int NOT NULL,
9    Annual_salary int NOT NULL,
10   PRIMARY KEY (Sav_acc_num)
11 );
12 DESCRIBE Saving_account;
13 CREATE TABLE Home_loan_account (
14   loan_acc_num int NOT NULL,
15   Total_limit int NOT NULL,
16   Total_requested_loan int NOT NULL,
17   PRIMARY KEY (loan_acc_num)
18 );
19 DESCRIBE Home_loan_account;
20 CREATE TABLE Customer (
21   Customer_ID int NOT NULL,
22   Specified_address varchar(50) NOT NULL,
23   Requested_loan int NOT NULL,
24   Individual_loan_limit int NOT NULL,
25   Sav_acc_num int NOT NULL,
26   loan_acc_num int NOT NULL,
27   PRIMARY KEY (Customer_ID),
28   FOREIGN KEY (Sav_acc_num) REFERENCES Saving_account(Sav_acc_num),
29   FOREIGN KEY (loan_acc_num) REFERENCES Home_loan_account(loan_acc_num)
30 );
31 DESCRIBE Customer;
```

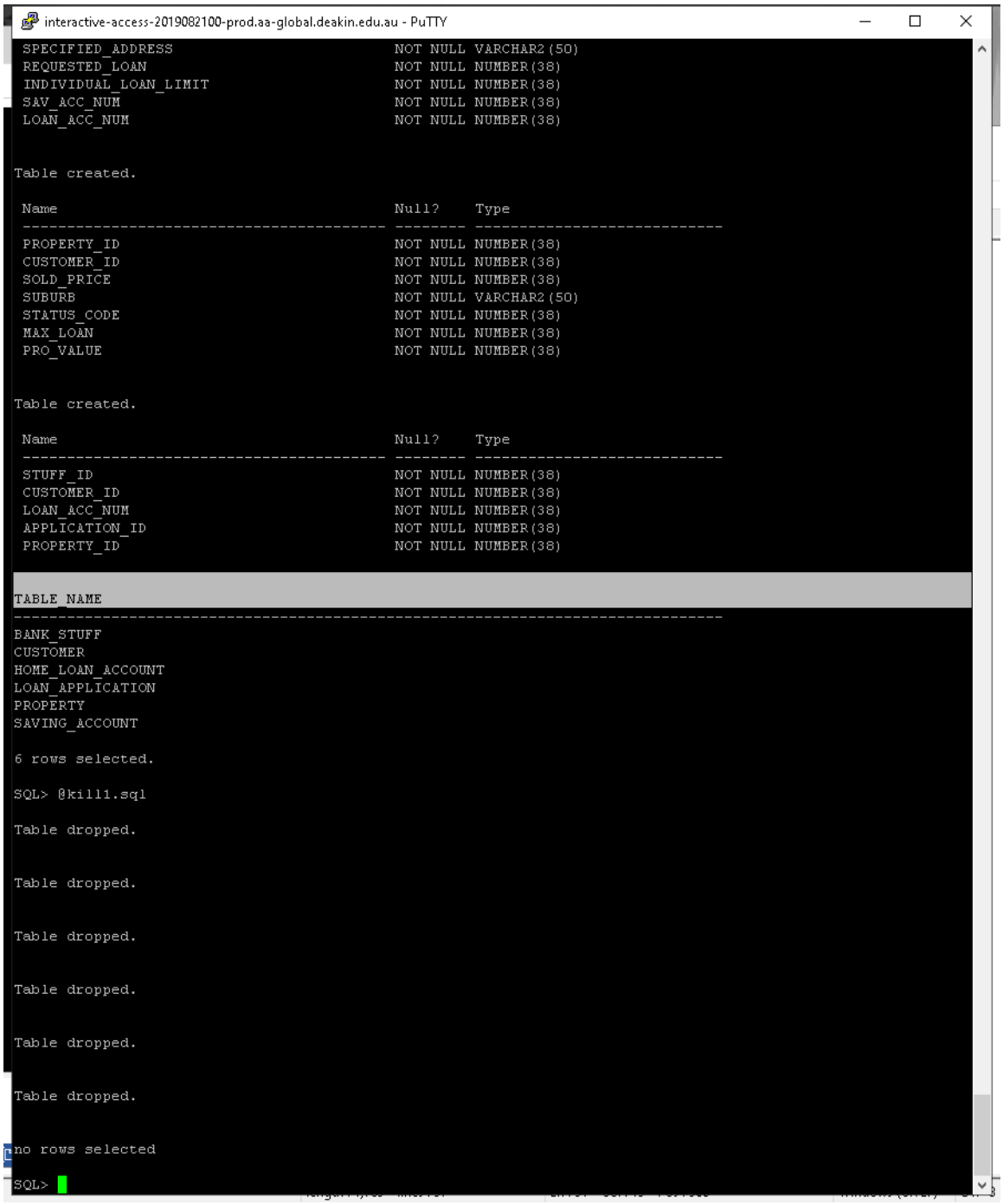
make1.sql

```sql
CREATE TABLE Property (
  Property_ID int NOT NULL,
  Customer_ID int NOT NULL,
  Sold_Price int NOT NULL,
  Suburb varchar(50) NOT NULL,
  Status_code int NOT NULL,
  Max_loan int NOT NULL,
  Pro_value int NOT NULL,
  PRIMARY KEY (Property_ID),
  FOREIGN KEY (Customer_ID) REFERENCES Customer(Customer_ID)
);
DESCRIBE Property;
CREATE TABLE Loan_application (
  Stuff_ID int NOT NULL,
  Customer_ID int NOT NULL,
  loan_acc_num int NOT NULL,
  Application_ID int NOT NULL,
  Property_ID int NOT NULL,
  PRIMARY KEY (Application_ID),
  FOREIGN KEY (Customer_ID) REFERENCES Customer(Customer_ID),
  FOREIGN KEY (loan_acc_num) REFERENCES Home_loan_account(loan_acc_num),
  FOREIGN KEY (Stuff_ID) REFERENCES bank_stuff(Stuff_ID),
  FOREIGN KEY (Property_ID) REFERENCES Property(Property_ID)
);
DESCRIBE Loan_application;
SELECT table_name FROM user_tables;
```

kill1.sql

```sql
DROP TABLE Loan_application;
DROP TABLE Property;
DROP TABLE bank_stuff;
DROP TABLE Customer;
DROP TABLE Saving_account;
DROP TABLE Home_loan_account;
SELECT table_name FROM user_tables;
```


interactive-access-2019082100-prod.aa-global.deakin.edu.au - PuTTY
SQL> @make1.sql
Table created.
Name Null? Type
STUFF_ID NOT NULL NUMBER(38)
DEPARTMENT NOT NULL VARCHAR2(50)
Table created.
Name Null? Type
SAV_ACC_NUM NOT NULL NUMBER(38)
ANNUAL_SALARY NOT NULL NUMBER(38)
Table created.
Name Null? Type
LOAN_ACC_NUM NOT NULL NUMBER(38)
TOTAL_LIMIT NOT NULL NUMBER(38)
TOTAL_REQUESTED_OAN NOT NULL NUMBER(38)
Table created.
Name Null? Type
CUSTOMER_ID NOT NULL NUMBER(38)
SPECIFIED_ADDRESS NOT NULL VARCHAR2(50)
REQUESTED_LOAN NOT NULL NUMBER(38)
INDIVIDUAL_LOAN_LIMIT NOT NULL NUMBER(38)
SAV_ACC_NUM NOT NULL NUMBER(38)
LOAN_ACC_NUM NOT NULL NUMBER(38)
Table created.
Name Null? Type
PROPERTY_ID NOT NULL NUMBER(38)
CUSTOMER_ID NOT NULL NUMBER(38)
SOLD_PRICE NOT NULL NUMBER(38)
SUBURB NOT NULL VARCHAR2(50)
STATUS_CODE NOT NULL NUMBER(38)
MAX_LOAN NOT NULL NUMBER(38)
PRO_VALUE NOT NULL NUMBER(38)
Table created.
Name Null? Type
STUFF_ID NOT NULL NUMBER(38)
CUSTOMER_ID NOT NULL NUMBER(38)
LOAN_ACC_NUM NOT NULL NUMBER(38)
APPLICATION_ID NOT NULL NUMBER(38)
PROPERTY_ID NOT NULL NUMBER(38)
TABLE_NAME

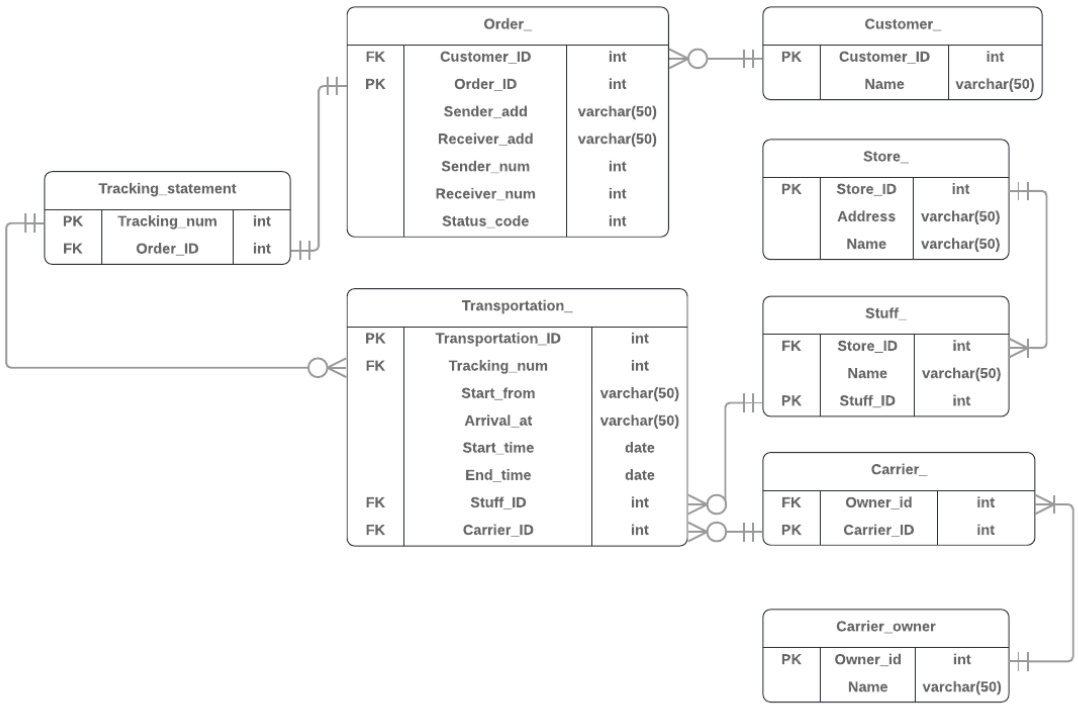

interactive-access-2019082100-prod.aa-global.deakin.edu.au - PuTTY
SPECIFIED_ADDRESS NOT NULL VARCHAR2(50)
REQUESTED_LOAN NOT NULL NUMBER(38)
INDIVIDUAL_LOAN_LIMIT NOT NULL NUMBER(38)
SAV_ACC_NUM NOT NULL NUMBER(38)
LOAN_ACC_NUM NOT NULL NUMBER(38)
Table created.
Name Null? Type
PROPERTY_ID NOT NULL NUMBER(38)
CUSTOMER_ID NOT NULL NUMBER(38)
SOLD_PRICE NOT NULL NUMBER(38)
SUBURB NOT NULL VARCHAR2(50)
STATUS_CODE NOT NULL NUMBER(38)
MAX_LOAN NOT NULL NUMBER(38)
PRO_VALUE NOT NULL NUMBER(38)
Table created.
Name Null? Type
STUFF_ID NOT NULL NUMBER(38)
CUSTOMER_ID NOT NULL NUMBER(38)
LOAN_ACC_NUM NOT NULL NUMBER(38)
APPLICATION_ID NOT NULL NUMBER(38)
PROPERTY_ID NOT NULL NUMBER(38)
TABLE_NAME
BANK_STUFF
CUSTOMER
HOME_LOAN_ACCOUNT
LOAN_APPLICATION
PROPERTY
SAVING_ACCOUNT
6 rows selected.
SQL> @kill1.sql
Table dropped.
Table dropped.
Table dropped.
Table dropped.
Table dropped.
Table dropped.
no rows selected
SQL>

Task 2.1

Let's assume we are making ERD for a delivery company which is similar but much smaller than Australia Post, and following are requirements:

- Each customer will have a customer ID.
- After a customer pay for a delivery, he or she will have a receipt with an order number and a tracking number.
- A tracking number will help with tracking their parcel. A tracking statement should display its tracking number and all the completed transportation activities that made the parcel moves.
- A transportation activity has its ID, starting place and arriving place, starting time and arriving time. When it finishes, a stuff should confirm it and signed, and fill the ID of the vehicle which finished this transportation.
- Only one vehicle be used during one transportation. When there's another vehicle carries the parcel, there should be a different transportation with different transportation ID.
- Each vehicle has their owner, and each owner may have more than one vehicle.
- Each stuff of this company works in a store and has an ID. This company has many stores and each of them has an ID.

Task 2.2



Task 2.3

make2.sql

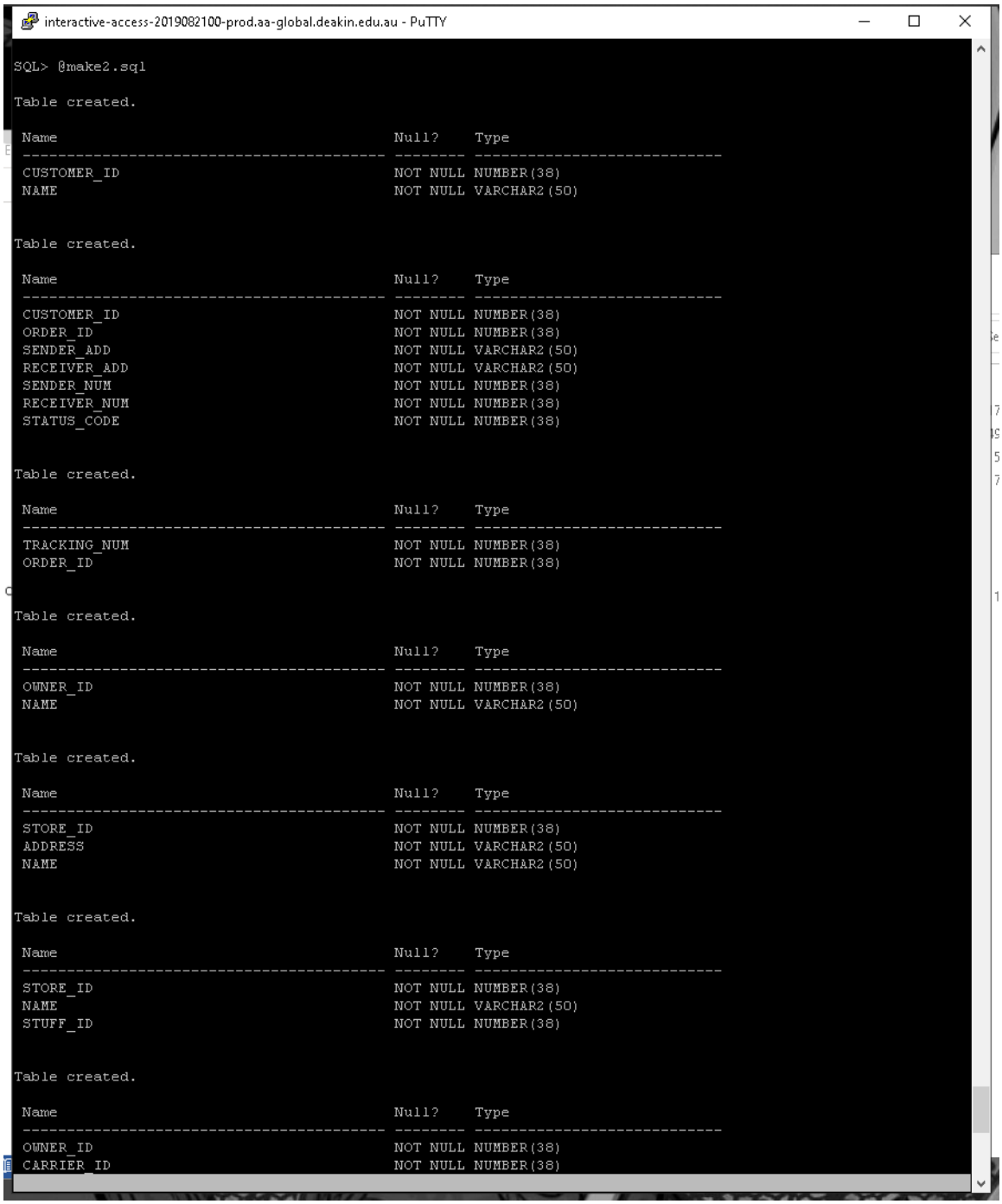
```sql
CREATE TABLE Customer_ (
  Customer_ID int NOT NULL,
  Name varchar(50) NOT NULL,
  PRIMARY KEY (Customer_ID)
);
DESCRIBE Customer_;
CREATE TABLE Order_ (
  Customer_ID int NOT NULL,
  Order_ID int NOT NULL,
  Sender_add varchar(50) NOT NULL,
  Receiver_add varchar(50) NOT NULL,
  Sender_num int NOT NULL,
  Receiver_num int NOT NULL,
  Status_code int NOT NULL,
  PRIMARY KEY (Order_ID),
  FOREIGN KEY (Customer_ID) REFERENCES Customer_(Customer_ID)
);
DESCRIBE Order_;
CREATE TABLE Tracking_statement_ (
  Tracking_num int NOT NULL,
  Order_ID int NOT NULL,
  PRIMARY KEY (Tracking_num),
  FOREIGN KEY (Order_ID) REFERENCES Order_(Order_ID)
);
DESCRIBE Tracking_statement_;
CREATE TABLE Carrier_owner_ (
  Owner_id int NOT NULL,
  Name varchar(50) NOT NULL,
  PRIMARY KEY (Owner_id)
);
DESCRIBE Carrier_owner_;
CREATE TABLE Store_ (
  Store_ID int NOT NULL,
  Address varchar(50) NOT NULL,
  Name varchar(50) NOT NULL,
  PRIMARY KEY (Store_ID)
);
DESCRIBE Store_;
```

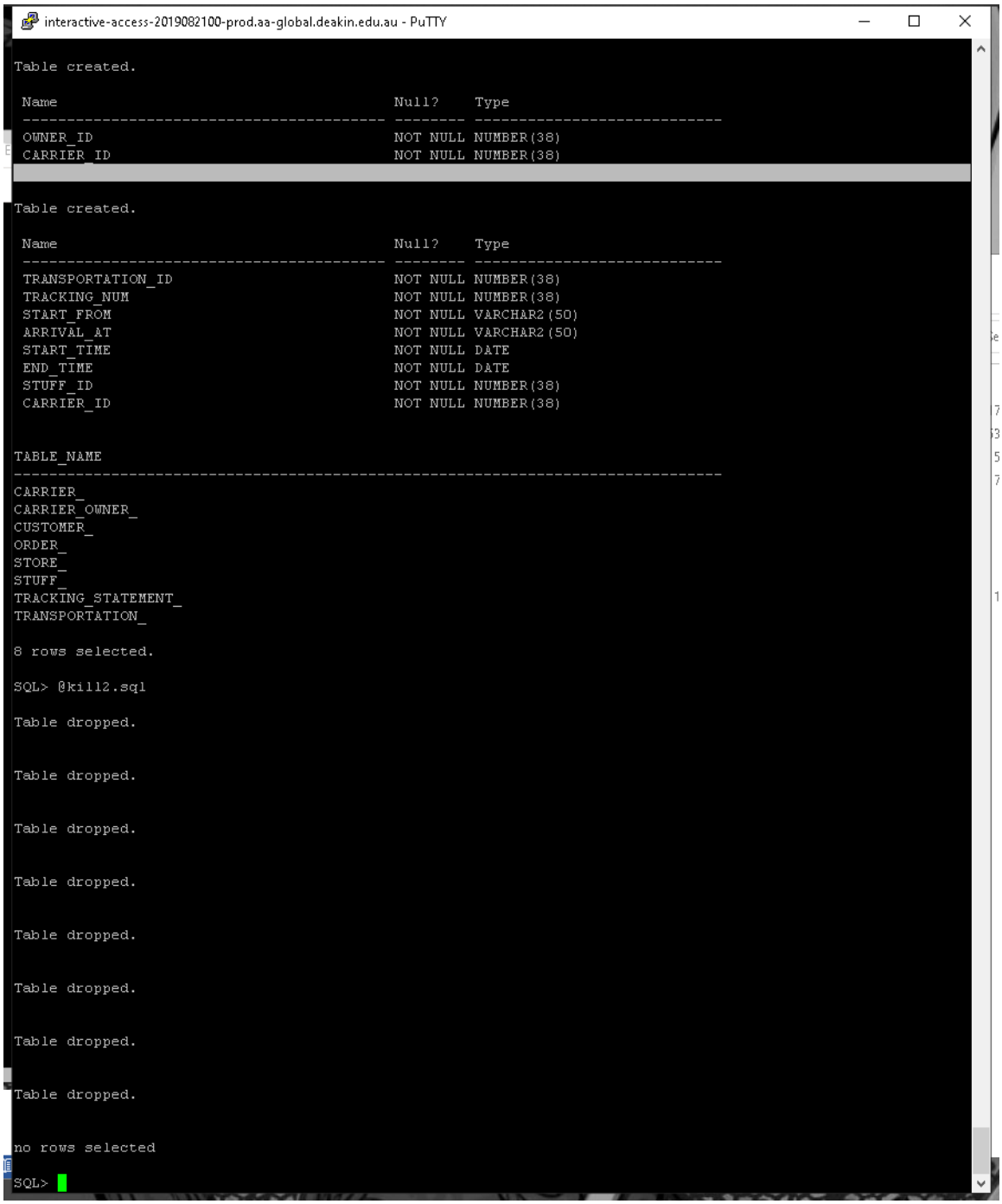
make2.sql

```
39  CREATE TABLE Stuff_ (
40    Store_ID int NOT NULL,
41    Name varchar(50) NOT NULL,
42    Stuff_ID int NOT NULL,
43    PRIMARY KEY (Stuff_ID),
44    FOREIGN KEY (Store_ID) REFERENCES Store_(Store_ID)
45  );
46  DESCRIBE Stuff_;
47  CREATE TABLE Carrier_ (
48    Owner_id int NOT NULL,
49    Carrier_ID int NOT NULL,
50    PRIMARY KEY (Carrier_ID),
51    FOREIGN KEY (Owner_id) REFERENCES Carrier_owner_(Owner_id)
52  );
53  DESCRIBE Carrier_;
54  CREATE TABLE Transportation_ (
55    Transportation_ID int NOT NULL,
56    Tracking_num int NOT NULL,
57    Start_from varchar(50) NOT NULL,
58    Arrival_at varchar(50) NOT NULL,
59    Start_time date NOT NULL,
60    End_time date NOT NULL,
61    Stuff_ID int NOT NULL,
62    Carrier_ID int NOT NULL,
63    PRIMARY KEY (Transportation_ID),
64    FOREIGN KEY (Tracking_num) REFERENCES Tracking_statement_(Tracking_num),
65    FOREIGN KEY (Stuff_ID) REFERENCES Stuff_(Stuff_ID),
66    FOREIGN KEY (Carrier_ID) REFERENCES Carrier_(Carrier_ID)
67  );
68  DESCRIBE Transportation_;
69  SELECT table_name FROM user_tables;
```

kill2.sql

```
1  DROP TABLE Transportation_;
2  DROP TABLE Carrier_;
3  DROP TABLE Stuff_;
4  DROP TABLE Store_;
5  DROP TABLE Carrier_owner_;
6  DROP TABLE Tracking_statement_;
7  DROP TABLE Order_;
8  DROP TABLE Customer_;
9  SELECT table_name FROM user_tables;
```


interactive-access-2019082100-prod.aa-global.deakin.edu.au - PuTTY
SQL> @make2.sql
Table created.
Name Null? Type
CUSTOMER_ID NOT NULL NUMBER(38)
NAME NOT NULL VARCHAR2(50)
Table created.
Name Null? Type
CUSTOMER_ID NOT NULL NUMBER(38)
ORDER_ID NOT NULL NUMBER(38)
SENDER_ADD NOT NULL VARCHAR2(50)
RECEIVER_ADD NOT NULL VARCHAR2(50)
SENDER_NUM NOT NULL NUMBER(38)
RECEIVER_NUM NOT NULL NUMBER(38)
STATUS_CODE NOT NULL NUMBER(38)
Table created.
Name Null? Type
TRACKING_NUM NOT NULL NUMBER(38)
ORDER_ID NOT NULL NUMBER(38)
Table created.
Name Null? Type
OWNER_ID NOT NULL NUMBER(38)
NAME NOT NULL VARCHAR2(50)
Table created.
Name Null? Type
STORE_ID NOT NULL NUMBER(38)
ADDRESS NOT NULL VARCHAR2(50)
NAME NOT NULL VARCHAR2(50)
Table created.
Name Null? Type
STORE_ID NOT NULL NUMBER(38)
NAME NOT NULL VARCHAR2(50)
STUFF_ID NOT NULL NUMBER(38)
Table created.
Name Null? Type
OWNER_ID NOT NULL NUMBER(38)
CARRIER_ID NOT NULL NUMBER(38)


interactive-access-2019082100-prod.aa-global.deakin.edu.au - PuTTY
Table created.
Name Null? Type
OWNER_ID NOT NULL NUMBER(38)
CARRIER_ID NOT NULL NUMBER(38)
Table created.
Name Null? Type
TRANSPORTATION_ID NOT NULL NUMBER(38)
TRACKING_NUM NOT NULL NUMBER(38)
START_FROM NOT NULL VARCHAR2(50)
ARRIVAL_AT NOT NULL VARCHAR2(50)
START_TIME NOT NULL DATE
END_TIME NOT NULL DATE
STUFF_ID NOT NULL NUMBER(38)
CARRIER_ID NOT NULL NUMBER(38)
TABLE_NAME
CARRIER_
CARRIER_OWNER_
CUSTOMER_
ORDER_
STORE_
STUFF_
TRACKING_STATEMENT_
TRANSPORTATION_
8 rows selected.
SQL> @kill2.sql
Table dropped.
Table dropped.
Table dropped.
Table dropped.
Table dropped.
Table dropped.
Table dropped.
Table dropped.
no rows selected
SQL>