Southern Alberta institute of Technology

Title:

Workshop 4 Create Tables & Load Data

by

**Femi Adams**

Winter 2023

**Dba User Creation**

SQL> create user tea\_owner identified by tea\_owner;

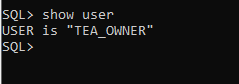
SQL> grant dba to tea\_owner;

SQL> connect tea\_owner/tea\_owner

Text

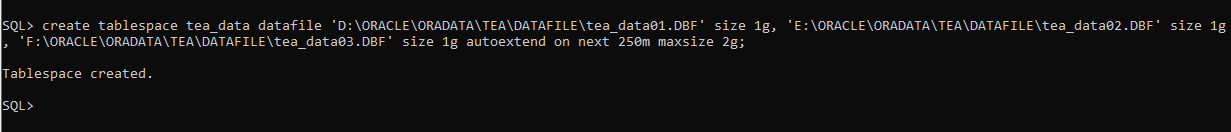
Description automatically generated

SQL> show user

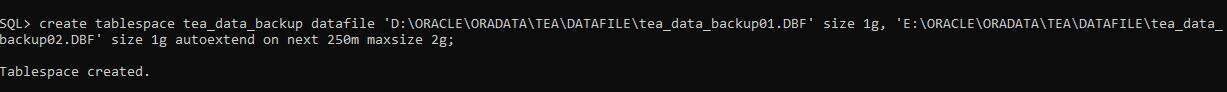


**Tablespace Creation**

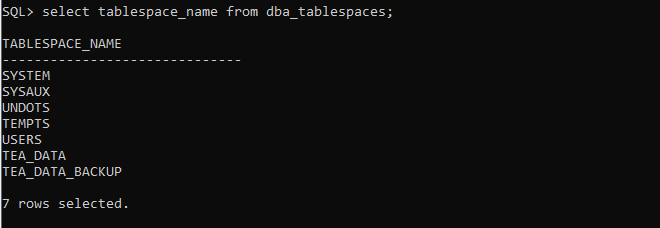
SQL> create tablespace tea\_data datafile 'D:\ORACLE\ORADATA\TEA\DATAFILE\tea\_data01.DBF' size 1g, 'E:\ORACLE\ORADATA\TEA\DATAFILE\tea\_data02.DBF' size 1g, 'F:\ORACLE\ORADATA\TEA\DATAFILE\tea\_data03.DBF' size 1g autoextend on next 250m maxsize 2g;



create tablespace tea\_data\_backup datafile 'D:\ORACLE\ORADATA\TEA\DATAFILE\tea\_data\_backup01.DBF' size 1g, 'E:\ORACLE\ORADATA\TEA\DATAFILE\tea\_data\_backup02.DBF' size 1g autoextend on next 250m maxsize 2g;



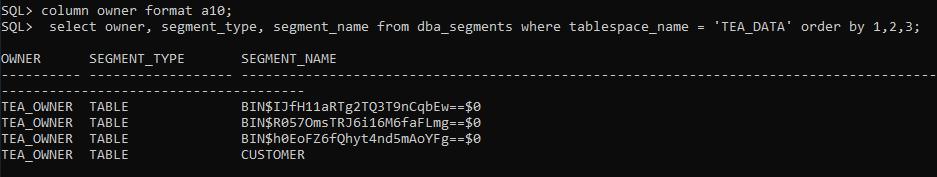
select tablespace\_name from dba\_tablespaces;



SQL> alter database default tablespace tea\_data;



SQL> select owner, segment\_type, segment\_name from dba\_segments where tablespace\_name = 'TEA\_DATA' order by 1,2,3;



**External Table Creation**

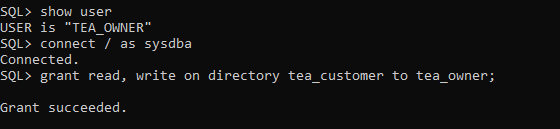
**Customer\_load External Table**

Lookup table on original excel data

SQL> create directory tea\_customer as 'C:\Temp\Loader';



SQL> grant read, write on directory tea\_customer to tea\_owner;



SQL> @C:\connTemp\Loader\teacustomertable.sql

**Create table Customer load file**

create table customer\_load

(

customer\_id number(6),

first\_name varchar2(50),

last\_name varchar2(50),

agent\_code varchar2(3),

email varchar2(50),

home\_phone varchar2(15),

business\_phone varchar2(15),

date\_of\_birth date,

address varchar2(50),

city varchar2(25),

postal\_code varchar2(7),

province varchar2(7),

country varchar2(15)

)

organization external

(

type oracle\_loader

default directory tea\_customer

access parameters

(

records delimited by newline

fields terminated by ","

optionally enclosed by '"'

(

customer\_id CHAR,

first\_name CHAR,

last\_name CHAR,

agent\_code CHAR,

email CHAR,

home\_phone CHAR,

business\_phone CHAR,

date\_of\_birth date “mm/dd/yyyy”,

address CHAR,

city CHAR,

postal\_code CHAR,

province CHAR,

country CHAR

)

)

location ('teacustomer.csv')

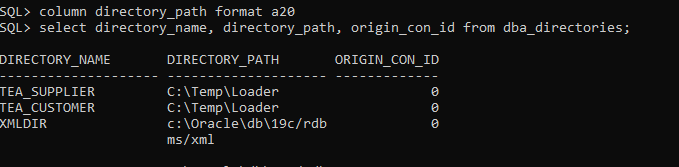
)

/

**Supplier\_load External Table**

This table is a look up table for the original data in the excel sheet

SQL> create directory tea\_supplier as 'C:\Temp\Loader';

SQL> select directory\_name, directory\_path, origin\_con\_id from dba\_directories;

SQL> @C:\Temp\Loader\teasuppliertable.sql

Create table supplier load file

create table supplier\_load

(

product\_supplier number(6),

product\_category number(6),

supplier\_office number(1),

product\_description varchar2(50),

contact\_name varchar2(50),

company varchar2(80),

address1 varchar2(50),

address2 varchar2(50),

city varchar2(25),

province varchar2(2),

postal\_code varchar2(7),

country varchar2(15),

phone\_number number(12),

fax number(12),

email varchar2(50),

website varchar2(50),

represents varchar2(25),

affiliation varchar2(25)

)

organization external

(

type oracle\_loader

default directory tea\_supplier

access parameters

(

records delimited by newline

fields terminated by ","

(

product\_supplier CHAR,

product\_category CHAR,

supplier\_office CHAR,

product\_description CHAR,

contact\_name CHAR,

company CHAR,

address1 CHAR,

address2 CHAR,

city CHAR,

province CHAR,

postal\_code CHAR,

country CHAR,

phone\_number CHAR,

fax CHAR,

email CHAR,

website CHAR,

represents CHAR,

affiliation CHAR

)

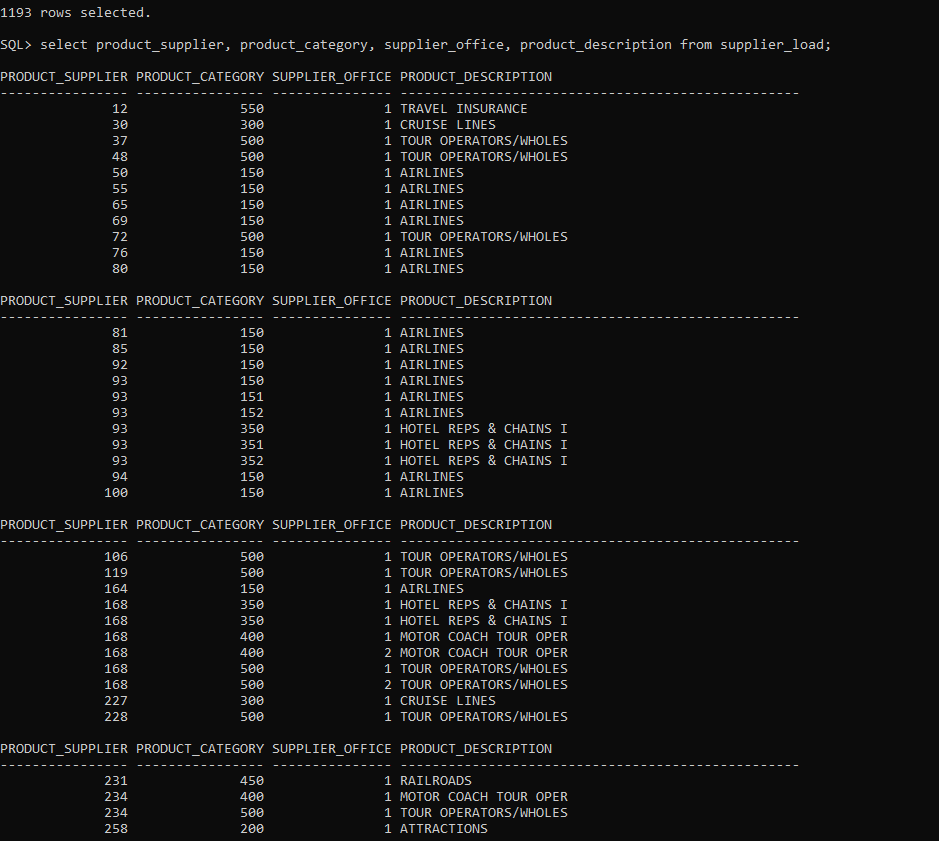
)

location ('teasupplier.csv')

)

/

SQL> select \* from supplier\_load;



**Main Supplier External Table**

This is a table for the suppliers that represents other suppliers

SQL> @C:\Temp\Loader\teamainsuppliertable.sql

Create main supplier load file

Main\_Supplier\_Load Table

create table main\_supplier\_load

(

product\_supplier number(6),

product\_category number(6),

supplier\_office number(1),

product\_description varchar2(50),

contact\_name varchar2(50),

company varchar2(80),

address1 varchar2(50),

address2 varchar2(50),

city varchar2(25),

province varchar2(2),

postal\_code varchar2(7),

country varchar2(15),

phone\_number number(12),

fax number(12),

email varchar2(50),

website varchar2(50),

represents varchar2(25),

affiliation varchar2(25)

)

organization external

(

type oracle\_loader

default directory tea\_supplier

access parameters

(

records delimited by newline

fields terminated by ","

(

product\_supplier CHAR,

product\_category CHAR,

supplier\_office CHAR,

product\_description CHAR,

contact\_name CHAR,

company CHAR,

address1 CHAR,

address2 CHAR,

city CHAR,

province CHAR,

postal\_code CHAR,

country CHAR,

phone\_number CHAR,

fax CHAR,

email CHAR,

website CHAR,

represents CHAR,

affiliation CHAR

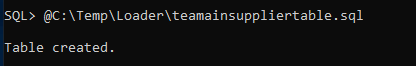
)

)

location ('teamainsupplier.csv')

)

REJECT LIMIT UNLIMITED;



**Company External Table**

This table shows a list of all the suppliers, their names, the supplier\_id that represents them and the supplier group they belong to. Company\_id is a unique value that is not in the original data, it represents a value of all suppliers in the table. Supplier\_rep is a foreign key of company\_id in the same table. Supplier\_rep is the supplier\_id that other suppliers belong to.

SQL> @C:\Temp\Loader\teacompany.sql

Create company load file

create table company\_load

(

company\_id number(6),

company\_name varchar2(80),

supplier\_id number(6),

supplier\_rep number(6)

)

organization external

(

type oracle\_loader

default directory tea\_supplier

access parameters

(

records delimited by newline

fields terminated by ","

OPTIONALLY ENCLOSED BY '"'

(

company\_id CHAR,

company\_name CHAR,

supplier\_id CHAR,

supplier\_rep CHAR

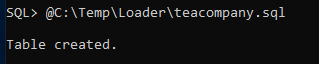
)

)

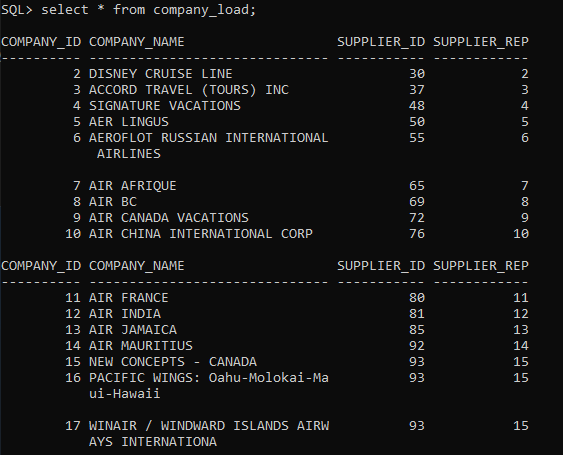
location ('teacompany.csv')

)

REJECT LIMIT UNLIMITED;



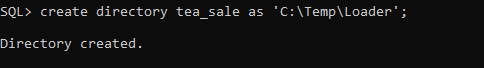
SQL> select \* from company\_load;



**Sale Load External Table**

Lookup table on original excel data

SQL> create directory tea\_sale as 'C:\Temp\Loader';



SQL> @ C:\Temp\Loader\teasaletable.sql

Create sale load file

/\*=====Load Sale External Data=====\*/

CREATE TABLE sale\_load

(

sale\_date date,

customer\_id number(8),

itinerary\_number number(8),

agent\_code varchar2(3),

booking\_number varchar2(15),

product\_category number(3),

product\_supplier number(8),

supplier\_office number(1),

trip\_start date,

trip\_end date,

class varchar2(5),

number\_of\_traveller number(3),

product varchar2(25),

description varchar2(60),

destination varchar2(25),

destination\_id varchar2(5),

credit\_card varchar2(15),

expiry\_date date,

card\_number number(20),

billing\_date date,

bill\_description varchar2(15),

base\_price number(11,2),

total\_price number(11,2),

billed\_amount number(11,2),

agency\_fee\_code varchar2(3),

agency\_fee\_amount number(6,2),

agency\_commission number(8,2)

)

ORGANIZATION EXTERNAL

(

TYPE ORACLE\_LOADER

DEFAULT DIRECTORY tea\_sale

ACCESS PARAMETERS

(

RECORDS DELIMITED BY NEWLINE

FIELDS TERMINATED BY ","

OPTIONALLY ENCLOSED BY '"'

(

sale\_date date "mm/dd/yyyy",

customer\_id char,

itinerary\_number char,

agent\_code char,

booking\_number char,

product\_category char,

product\_supplier char,

supplier\_office char,

trip\_start date "mm/dd/yyyy",

trip\_end date "mm/dd/yyyy",

class char,

number\_of\_traveller char,

product char,

description char,

destination char,

destination\_id char,

credit\_card char,

expiry\_date date "mm/dd/yyyy",

card\_number char,

billing\_date date "mm/dd/yyyy",

bill\_description char,

base\_price char,

total\_price char,

billed\_amount char,

agency\_fee\_code char,

agency\_fee\_amount char,

agency\_commission char

)

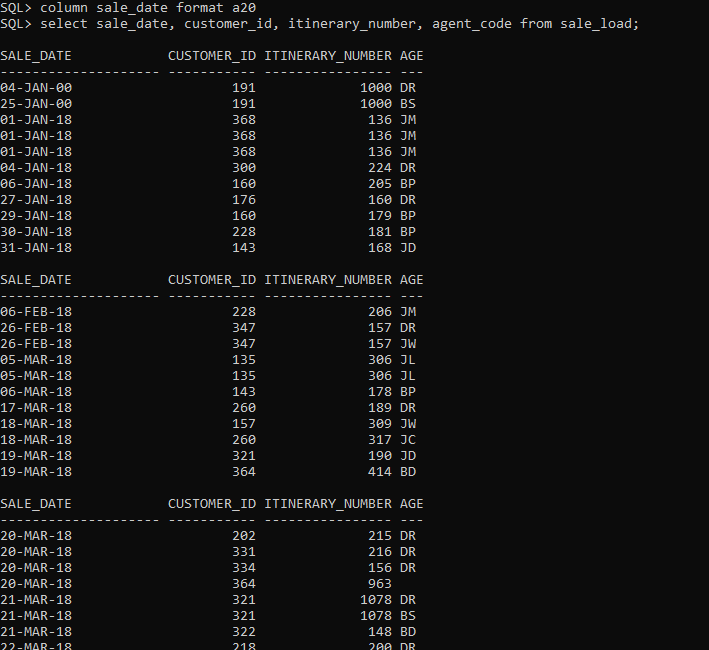
)

LOCATION ('teasale.csv')

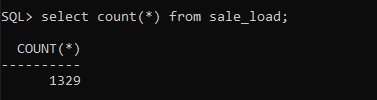
)

REJECT LIMIT UNLIMITED;

SQL> select \* from sale\_load;



SQL> select count(\*) from sale\_load;



**Product External Table**

SQL> @C:\Temp\Loader\teaproducttable.sql

Create product load file

/\*=====Load Product External Data=====\*/

CREATE TABLE product\_load

(

product\_id number(6),

product\_name varchar2(50)

)

ORGANIZATION EXTERNAL

(

TYPE ORACLE\_LOADER

DEFAULT DIRECTORY tea\_sale

ACCESS PARAMETERS

(

RECORDS DELIMITED BY NEWLINE

FIELDS TERMINATED BY ","

OPTIONALLY ENCLOSED BY '"'

(

product\_id char,

product\_name char

)

)

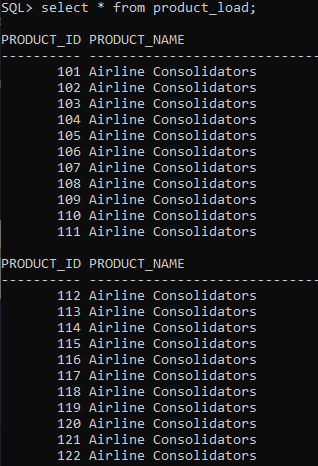
LOCATION ('teaproduct.csv')

)

REJECT LIMIT UNLIMITED;



SQL> select \* from product\_load;



**Payment External Table**

SQL> @C:\Temp\Loader\teapaymenttable.sql

Create payment load file

/\*=====Load Product External Data=====\*/

CREATE TABLE payment\_load

(

payment\_id number(6),

cust\_credit\_card\_id number(6),

payment\_type varchar2(15)

)

ORGANIZATION EXTERNAL

(

TYPE ORACLE\_LOADER

DEFAULT DIRECTORY tea\_sale

ACCESS PARAMETERS

(

RECORDS DELIMITED BY NEWLINE

FIELDS TERMINATED BY ","

OPTIONALLY ENCLOSED BY '"'

(

payment\_id char,

cust\_credit\_card\_id char,

payment\_type char

)

)

LOCATION ('teapayment.csv')

)

REJECT LIMIT UNLIMITED;



**Employee External Table**

SQL> @C:\Temp\Loader\teaemployeetable.sql

Create employee load file

create table employee\_load

(

employee\_id number(6),

first\_name varchar2(50),

last\_name varchar2(50),

department\_id number(3),

address varchar2(256),

city varchar2(50),

postal\_code varchar2(10),

phone\_number varchar(16),

email varchar2(50)

)

organization external

(

type oracle\_loader

default directory tea\_customer

access parameters

(

records delimited by newline

fields terminated by ","

optionally enclosed by '"'

(

employee\_id char,

first\_name char,

last\_name char,

department\_id char,

address char,

city char,

postal\_code char,

phone\_number char,

email char

)

)

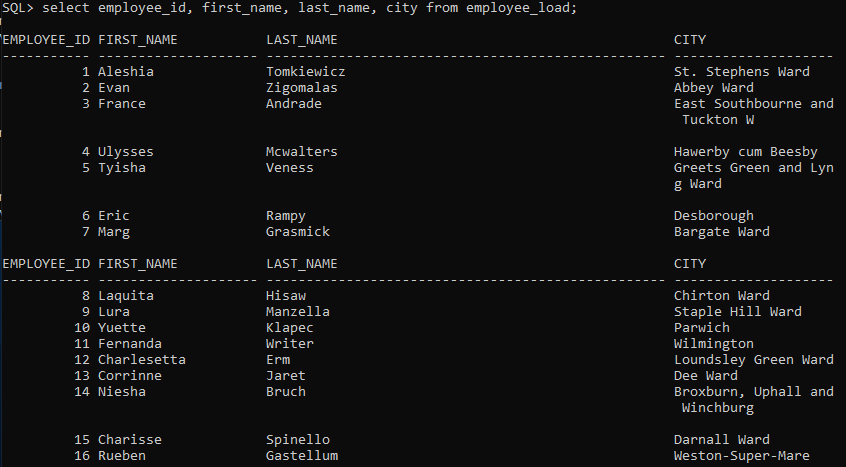
location ('teaemployee.csv')

)

REJECT LIMIT UNLIMITED;



SQL> select employee\_id, first\_name, last\_name, city from employee\_load;



**All Sale External Table**

SQL> @C:\Temp\Loader\teaallsaletable.sql

Create all sale load file

/\*=====Load All Sale External Data=====\*/

CREATE TABLE all\_sale\_load

(

sale\_id number(6),

employee\_id number(6),

customer\_id number(6),

itinerary\_number number(8),

product\_id number(6),

class varchar2(5),

supplier\_id number(6),

payment\_id number(6),

booking\_number varchar2(15),

destination\_id varchar2(5),

sale\_date date

)

ORGANIZATION EXTERNAL

(

TYPE ORACLE\_LOADER

DEFAULT DIRECTORY tea\_sale

ACCESS PARAMETERS

(

RECORDS DELIMITED BY NEWLINE

FIELDS TERMINATED BY ","

OPTIONALLY ENCLOSED BY '"'

(

sale\_id char,

employee\_id char,

customer\_id char,

itinerary\_number char,

product\_id char,

class char,

supplier\_id char,

payment\_id char,

booking\_number char,

destination\_id char,

sale\_date date "mm/dd/yyyy"

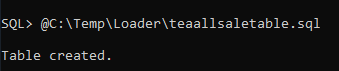
)

)

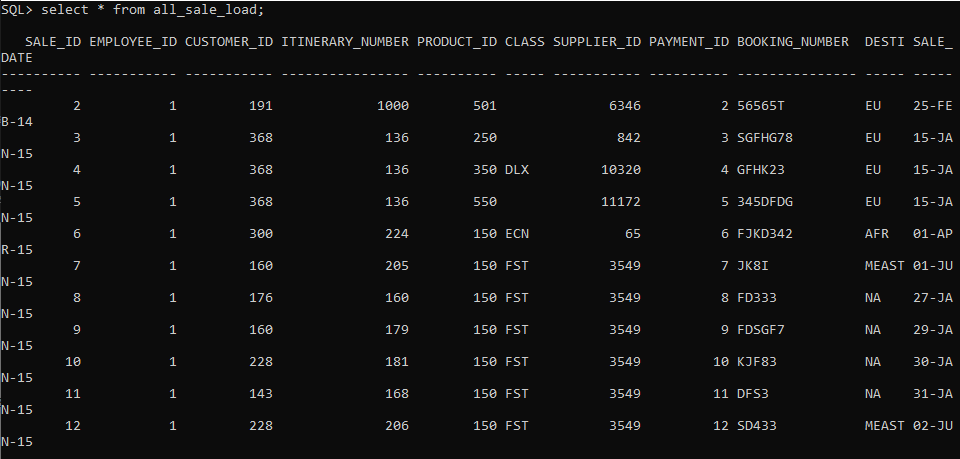
LOCATION ('teaallsale.csv')

)

REJECT LIMIT UNLIMITED;



SQL> select \* from all\_sale\_load;



**Booking Number External Table**

/\*=====Load Booking Number External Data=====\*/

CREATE TABLE booking\_number\_load

(

booking\_number varchar2(15)

)

ORGANIZATION EXTERNAL

(

TYPE ORACLE\_LOADER

DEFAULT DIRECTORY tea\_load

ACCESS PARAMETERS

(

RECORDS DELIMITED BY NEWLINE

FIELDS TERMINATED BY ","

OPTIONALLY ENCLOSED BY '"'

(

booking\_number char

)

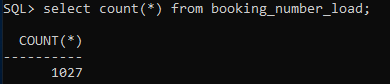
)

LOCATION ('teabookingnumber.csv')

)

REJECT LIMIT UNLIMITED;

SQL> @C:\Temp\Loader\teabookingnumber.sql



**Itinerary Number External Table**

/\*=====Load Itinerary Number External Data=====\*/

CREATE TABLE itinerary\_number\_load

(

itinerary\_number number

)

ORGANIZATION EXTERNAL

(

TYPE ORACLE\_LOADER

DEFAULT DIRECTORY tea\_load

ACCESS PARAMETERS

(

RECORDS DELIMITED BY NEWLINE

FIELDS TERMINATED BY ","

OPTIONALLY ENCLOSED BY '"'

(

itinerary\_number char

)

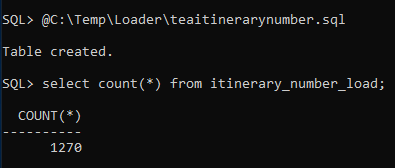
)

LOCATION ('teaitinerarynumber.csv')

)

REJECT LIMIT UNLIMITED;

SQL> @C:\Temp\Loader\teaitinerarynumber.sql

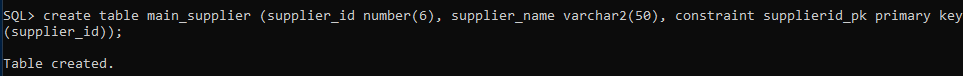


**Table Creation**

**Main Supplier Table**

This is a table for the suppliers that represents other suppliers

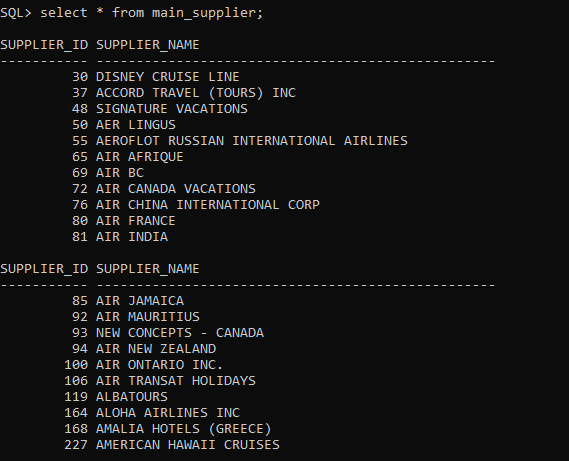
SQL> create table main\_supplier (supplier\_id number(6), supplier\_name varchar2(50), constraint supplierid\_pk primary key(supplier\_id));



SQL> insert into main\_supplier select product\_supplier, company from main\_supplier\_load;



SQL> select \* from main\_supplier;

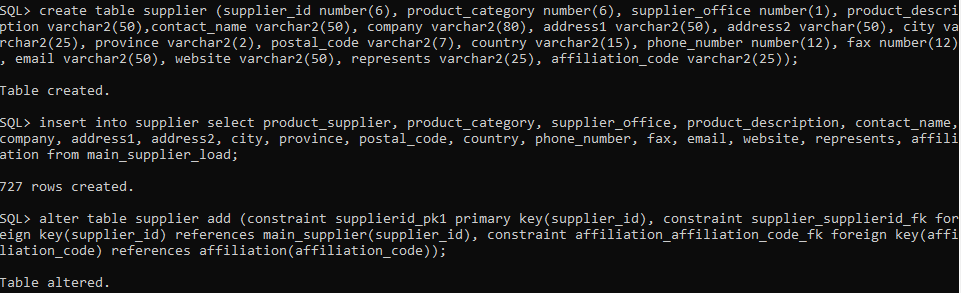


**Supplier Table**

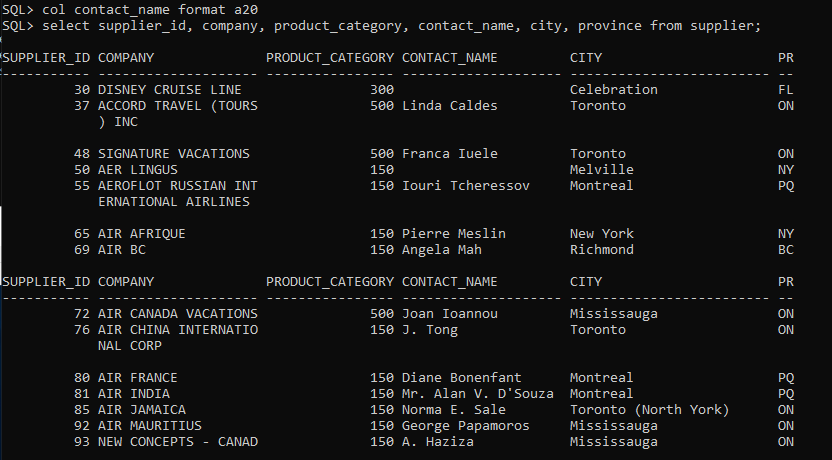
SQL> create table supplier (supplier\_id number(6), product\_category number(6), supplier\_office number(1), product\_description varchar2(50),contact\_name varchar2(50), company varchar2(80), address1 varchar2(50), address2 varchar(50), city varchar2(25), province varchar2(2), postal\_code varchar2(7), country varchar2(15), phone\_number number(12), fax number(12), email varchar2(50), website varchar2(50), represents varchar2(25), affiliation\_code varchar2(25));

SQL> insert into supplier select product\_supplier, product\_category, supplier\_office, product\_description, contact\_name,company, address1, address2, city, province, postal\_code, country, phone\_number, fax, email, website, represents, affiliation from main\_supplier\_load;

SQL> alter table supplier add (constraint supplierid\_pk1 primary key(supplier\_id), constraint supplier\_supplierid\_fk foreign key(supplier\_id) references main\_supplier(supplier\_id), constraint affiliation\_affiliation\_code\_fk foreign key(affiliation\_code) references affiliation(affiliation\_code));

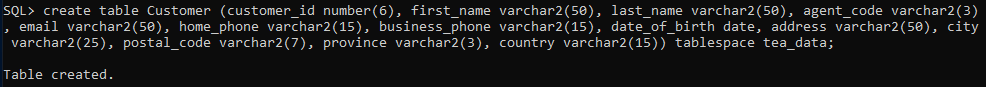


SQL> select supplier\_id, company, product\_category, contact\_name, city, province from supplier;

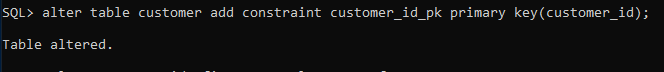


**Customer Table**

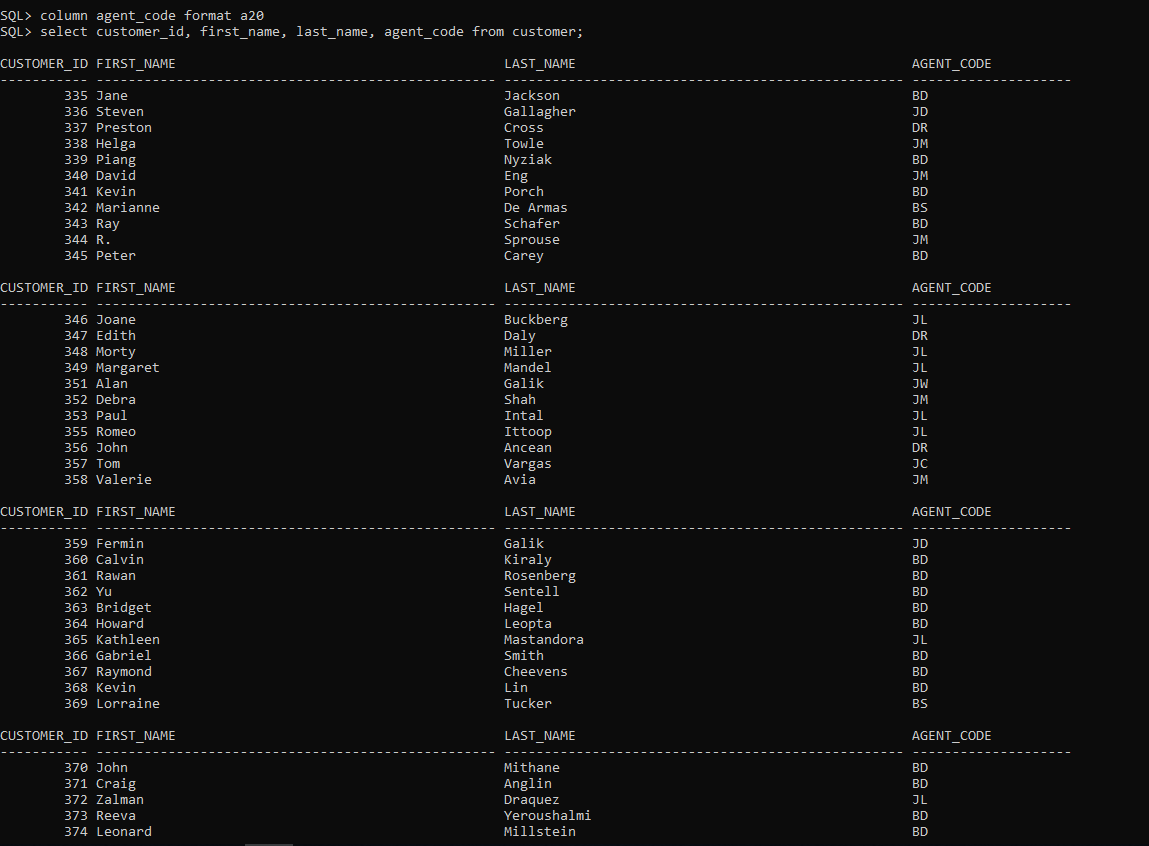
SQL> create table Customer (customer\_id number(6), first\_name varchar2(50), last\_name varchar2(50), agent\_code varchar2(3), email varchar2(50), home\_phone varchar2(15), business\_phone varchar2(15), date\_of\_birth date, address varchar2(50), city varchar2(25), postal\_code varchar2(7), province varchar2(3), country varchar2(15)) tablespace tea\_data;



SQL> alter table customer add constraint customer\_id\_pk primary key(customer\_id);

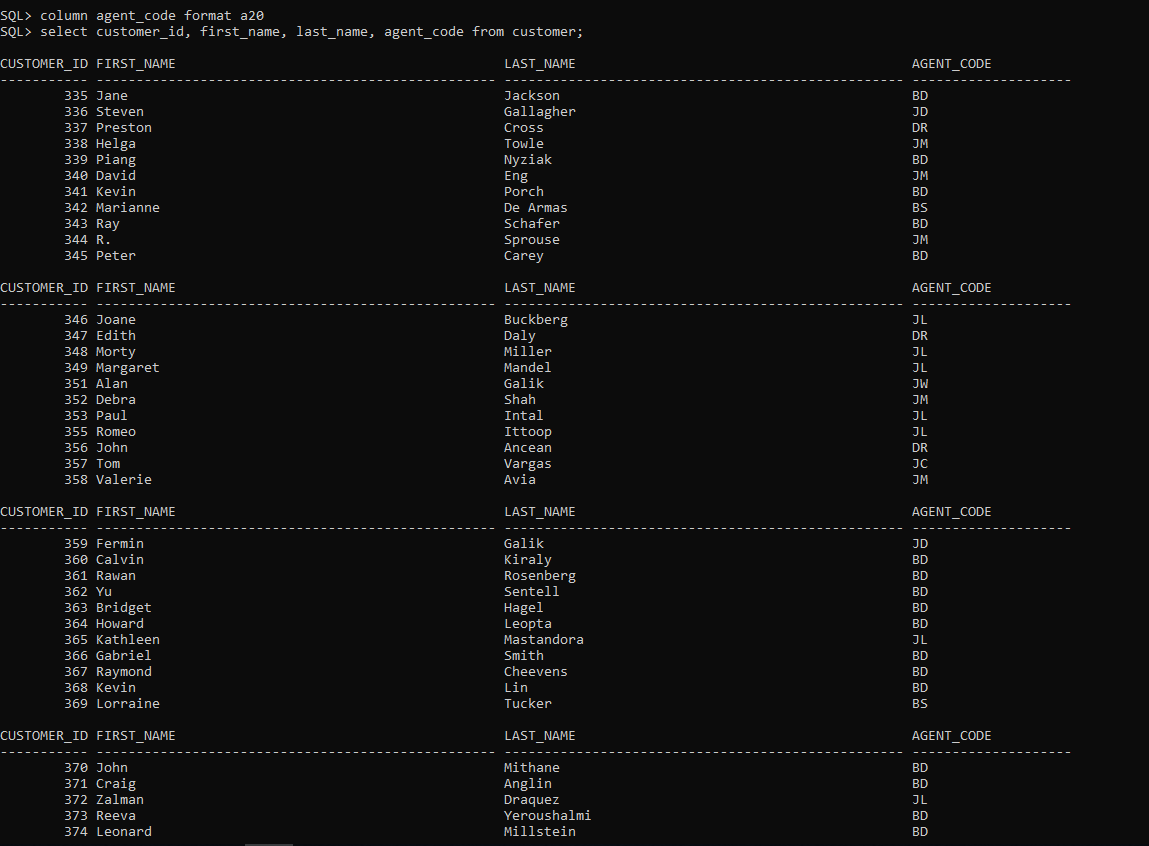


SQL> select \* from customer\_load;

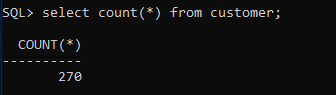


SQL> insert into customer select customer\_id, first\_name, last\_name, agent\_code, email, home\_phone, business\_phone, date\_of\_birth, address, city, postal\_code, province, country from customer\_load;

SQL> select \* from customer;

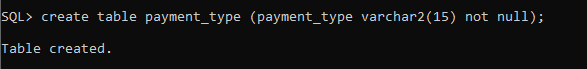


SQL> select count(\*) from customer;



**Payment Type Table**

SQL> create table payment\_type (payment\_type varchar2(15) not null);

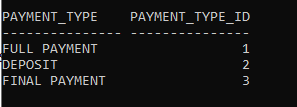


SQL> insert into payment\_type select distinct bill\_description from sale\_load where bill\_description is not null;

SQL> alter table payment\_type add payment\_type\_id integer generated by default on null as identity;

SQL> alter table payment\_type add constraint payment\_type\_pk primary key(payment\_type);

SQL> select \* from payment\_type;



**Payment Method Table**

SQL> create table payment\_method (payment\_method varchar2(15));

SQL> insert into payment\_method values ('cash');

SQL> insert into payment\_method values ('cheque');

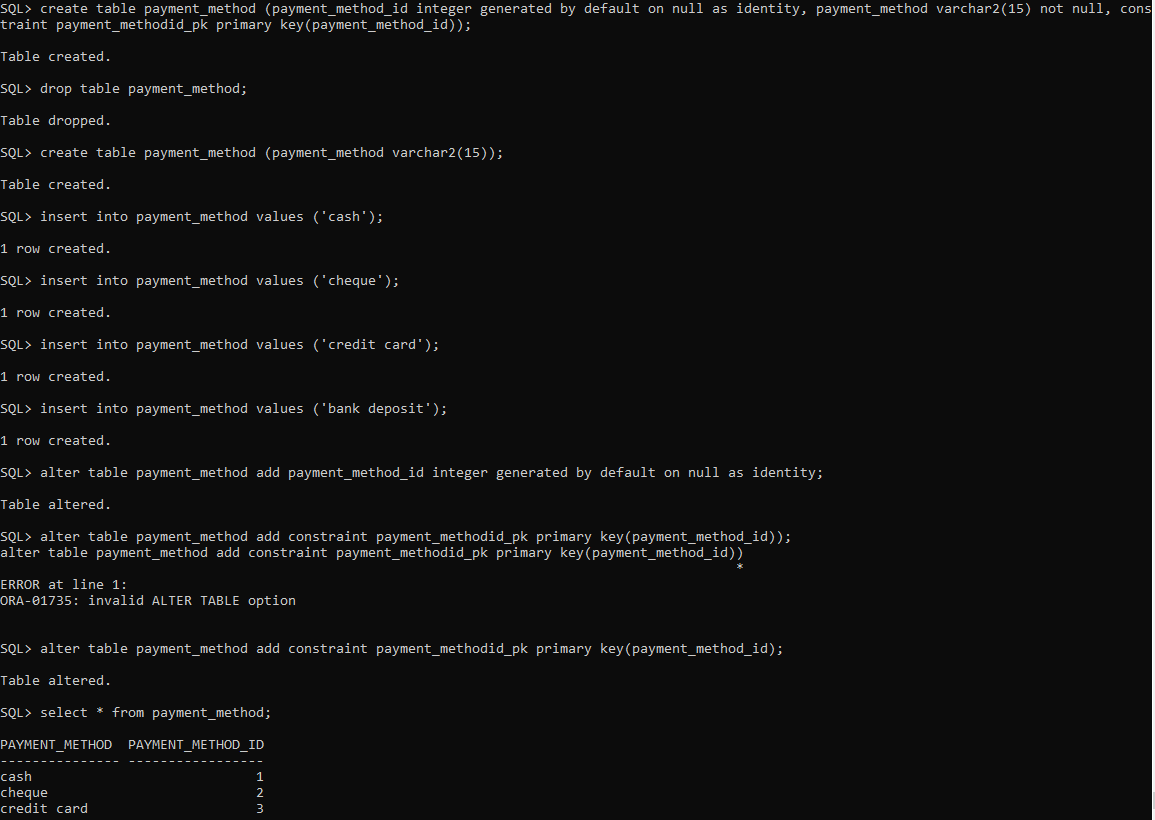
SQL> insert into payment\_method values ('credit card');

SQL> insert into payment\_method values ('bank deposit');

SQL> alter table payment\_method add payment\_method\_id integer generated by default on null as identity;

SQL> alter table payment\_method add constraint payment\_methodid\_pk primary key(payment\_method\_id);

SQL> select \* from payment\_method;



**Cust Credit Card Table**

SQL> create table cust\_credit\_card (customer\_id number(6), card\_type varchar2(15), card\_number number(16), expiry\_date date);

SQL> insert into cust\_credit\_card select customer\_id, credit\_card, card\_number, expiry\_date from sale\_load;

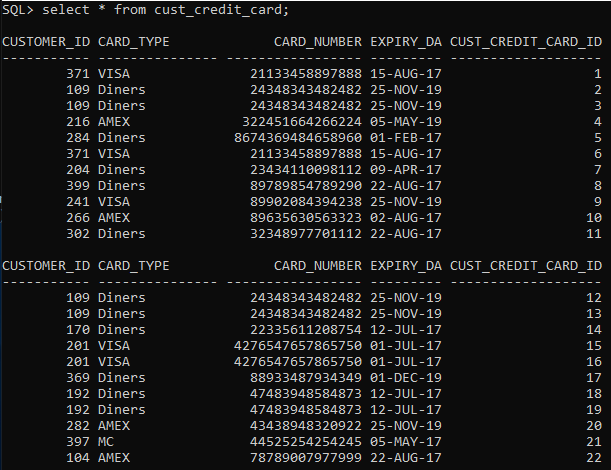
SQL> alter table cust\_credit\_card add constraint customer\_customerid\_fk1 foreign key(customer\_id) references customer(customer\_id);

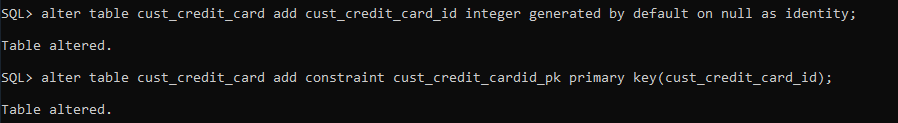
SQL> alter table cust\_credit\_card add cust\_credit\_card\_id integer generated by default on null as identity;

SQL> alter table cust\_credit\_card add constraint cust\_credit\_cardid\_pk primary key(cust\_credit\_card\_id);







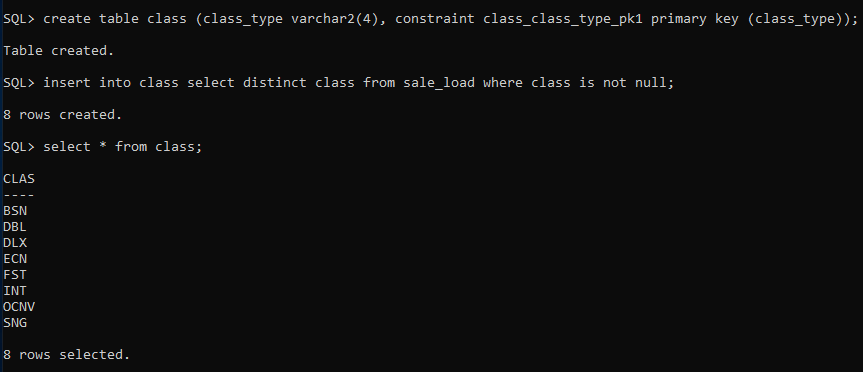


**Class Table**

SQL> create table class (class\_type varchar2(4), constraint class\_class\_type\_pk1 primary key (class\_type));

SQL> insert into class select distinct class from sale\_load where class is not null;

SQL> select \* from class;

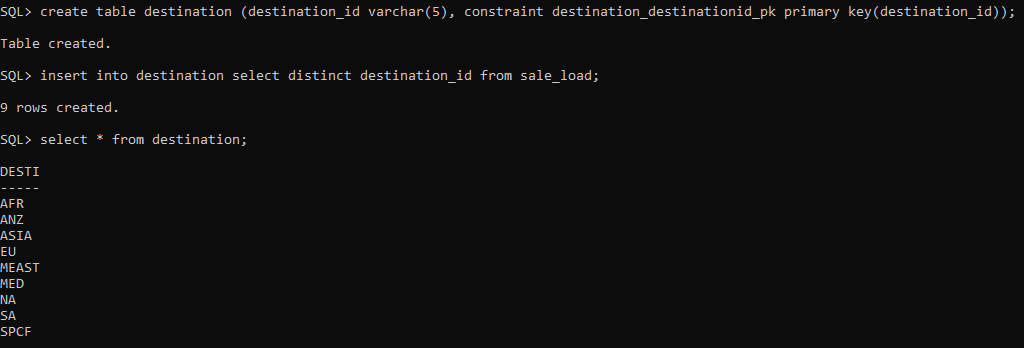


**Destination Table**

SQL> create table destination (destination\_id varchar(5), constraint destination\_destinationid\_pk primary key(destination\_id));

SQL> insert into destination select distinct destination\_id from sale\_load;

SQL> select \* from destination;

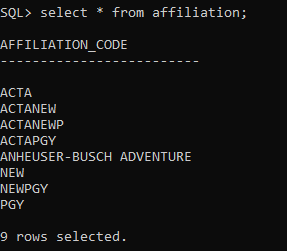


**Affiliation Table**

SQL> create table affiliation (affiliation\_code varchar2(25), constraint affiliation\_affiliationcode\_pk primary key(affiliation\_code));

SQL> insert into affiliation select distinct affiliation from supplier\_load where affiliation is not null;

SQL> select \* from affiliation;



**Company Table**

SQL> create table company (company\_id number(6), company\_name varchar2(80), supplier\_id number(6), supplier\_rep number(6));

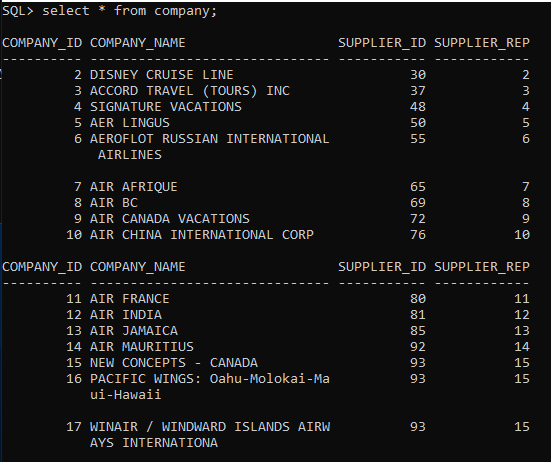
SQL> insert into company select company\_id, company\_name, supplier\_id, supplier\_rep from company\_load;

SQL> alter table company add constraint companyid\_pk primary key(company\_id);

SQL> alter table company add constraint main\_supplier\_supplierid\_fk foreign key(supplier\_id) references main\_supplier(supplier\_id);

SQL> alter table company add constraint company\_supplier\_rep\_fk foreign key(supplier\_rep) references company(company\_id);

SQL> select \* from company;

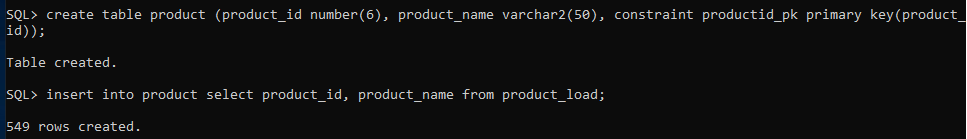


**Product Table**

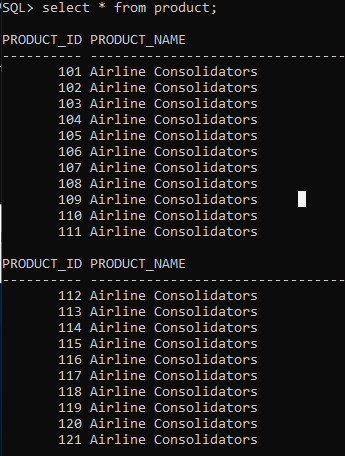
SQL> create table product (product\_id number(6), product\_name varchar2(50), constraint productid\_pk primary key(product\_id));

SQL> insert into product select product\_id, product\_name from product\_load;

insert into product (product\_id) values (35);



SQL> select \* from product;



**Payment Table**

SQL> create table payment (payment\_id integer generated by default on null as identity, cust\_credit\_card\_id number(6), payment\_type varchar2(15));

SQL> insert into payment select payment\_id, cust\_credit\_card\_id, payment\_type from payment\_load;

SQL> alter table payment add (constraint paymentid\_pk primary key(payment\_id), constraint payment\_type\_payment\_type\_fk foreign key(payment\_type) references payment\_type(payment\_type));

insert into payment values (591, 591,'');

insert into payment values (3,3,'');

insert into payment values (4,4,'');

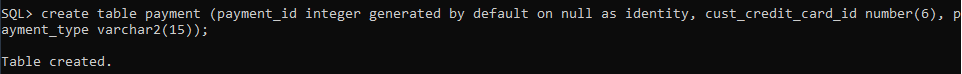
insert into payment values (369,369,'');

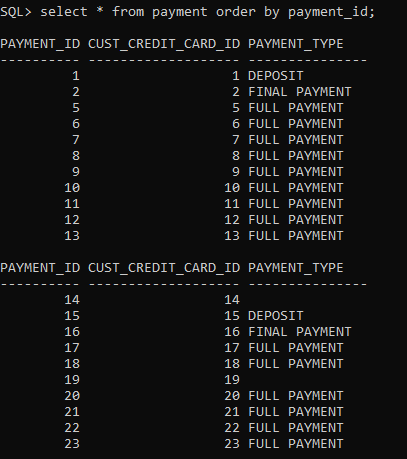
insert into payment values (510,510,'');

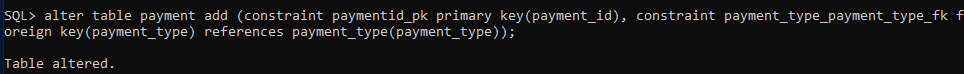
alter table payment add payment\_method\_id number;

alter table payment add constraint payment\_payment\_method\_id\_fk foreign key (payment\_method\_id) references payment\_method(payment\_method\_id);









**Commission Table**

Did not add FK on supplier\_id because it violates parent key. This is due to the data assigned to us, originally meant to have FK.

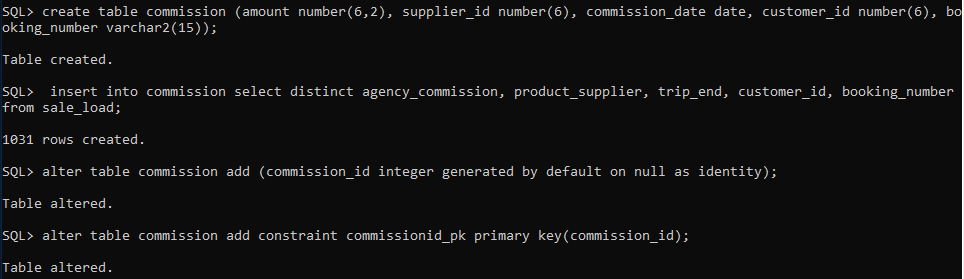
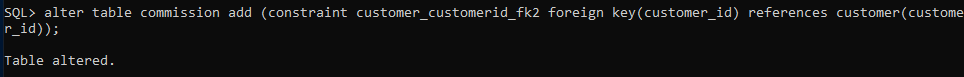
SQL> create table commission (amount number(6,2), supplier\_id number(6), commission\_date date, customer\_id number(6), booking\_number varchar2(15));

SQL> insert into commission select distinct agency\_commission, product\_supplier, trip\_end, customer\_id, booking\_number from sale\_load;

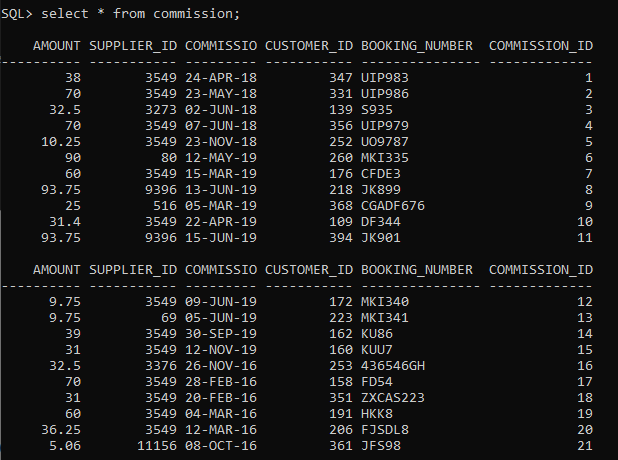
SQL> alter table commission add (commission\_id integer generated by default on null as identity);

SQL> alter table commission add constraint commissionid\_pk primary key(commission\_id);

SQL> alter table commission add (constraint customer\_customerid\_fk2 foreign key(customer\_id) references customer(customer\_id));

SQL> select \* from commission;



**Itinerary Number Table**

create table itinerary\_number (itinerary\_number number);

alter table itinerary\_number add constraint itinerary\_number\_pk primary key (itinerary\_number);

**Itinerary Table**

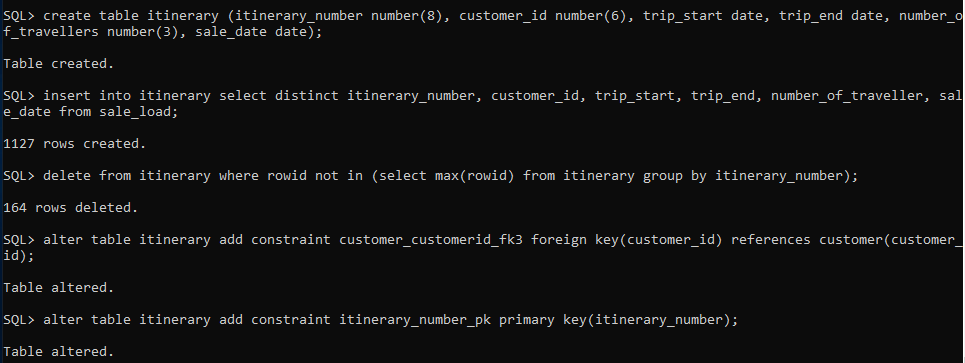
SQL> create table itinerary (itinerary\_number number(8), customer\_id number(6), trip\_start date, trip\_end date, number\_of\_travellers number(3), sale\_date date);

SQL> insert into itinerary select itinerary\_number, customer\_id, trip\_start, trip\_end, number\_of\_traveller, sale\_date from sale\_load;

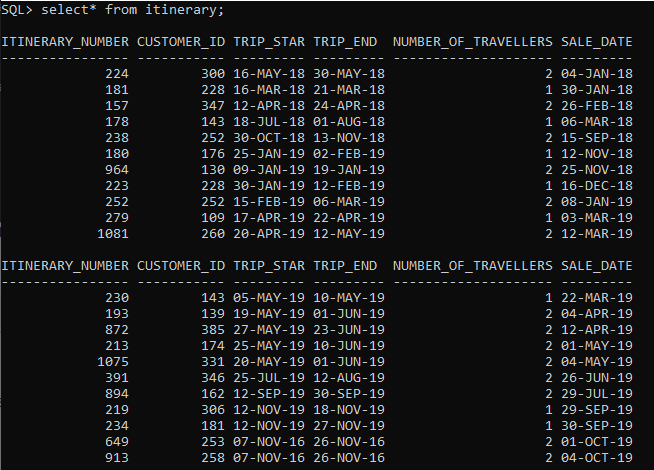
SQL> alter table itinerary add constraint customer\_customerid\_fk3 foreign key(customer\_id) references customer(customer\_id);

Not Needed [delete from itinerary where rowid not in (select max(rowid) from itinerary group by itinerary\_number);] used to delete duplicate rows.

SQL> alter table itinerary add constraint itinerarynumber\_itinerary\_number\_fk foreign key(itinerary\_number) references itinerary\_number(itinerary\_number);



SQL> select \* from itinerary order by itinerary\_number;

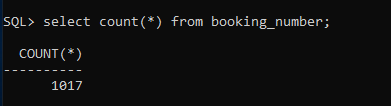


**Booking Number Table**

create table booking\_number (booking\_number varchar(256));

alter table booking\_number add constraint booking\_number\_pk primary key(booking\_number);

insert into booking\_number select booking\_number from booking\_number\_load;



**Booking Table**

SQL> create table booking (booking\_number varchar2(15), itinerary\_number number(8), product\_id number(3), payment\_id number(6), billing\_date date, base\_price number(11,2), billed\_amount number(11,2), total\_price number(11,2), bill\_description varchar2(15), agency\_fee\_amount number(6,2), agency\_fee\_code varchar2(3));

SQL> insert into booking select distinct booking\_number, itinerary\_number, product\_id, payment\_id, billing\_date, base\_price, billed\_amount, total\_price, bill\_description, agency\_fee\_amount, agency\_fee\_code from booking\_load;

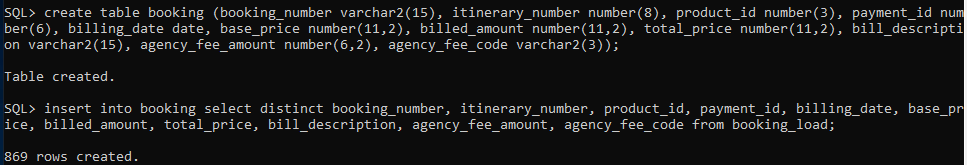
Not needed [SQL> delete from booking where rowid not in (select max(rowid) from booking group by booking\_number); (used to delete duplicate rows)]

SQL> alter table booking add constraint booking\_number\_bookingnumber\_fk foreign key (booking\_number) references booking\_number(booking\_number);

SQL> alter table booking add (constraint product\_productid\_fk3 foreign key(product\_id) references product(product\_id));

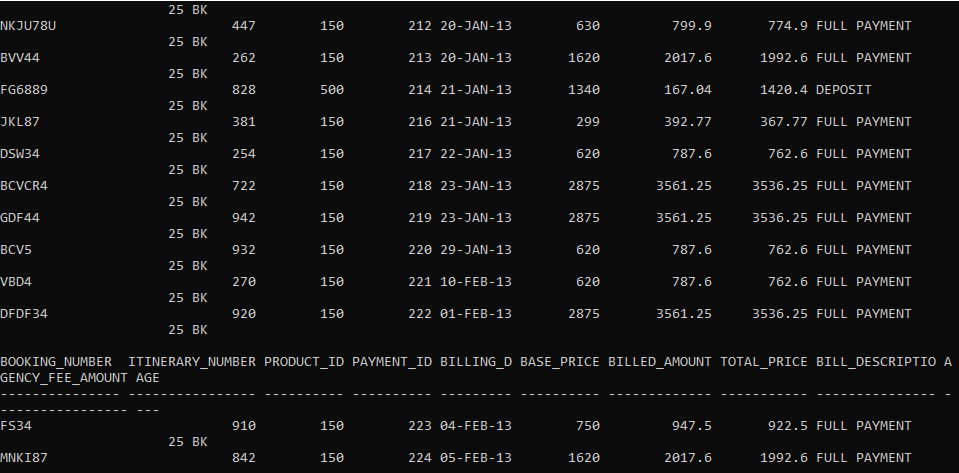
alter table booking add constraint payment\_paymentid\_fk foreign key (payment\_id) references payment (payment\_id);

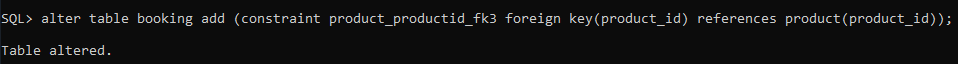
commit;





SQL> select \* from booking;





**Department Table**

SQL> create table department (department\_id number(6), department\_name varchar(50), constraint departmentid\_pk primary key(department\_id));

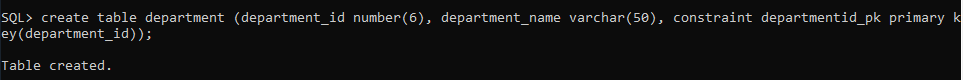
SQL> insert into department values (2, 'Airline Booking Department');

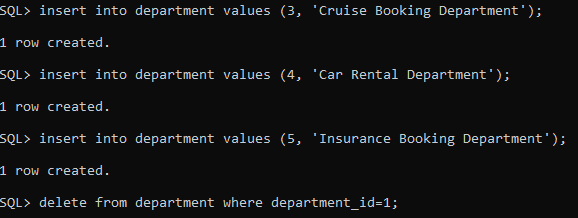
SQL> insert into department values (3, 'Cruise Booking Department');

SQL> insert into department values (4, 'Car Rental Department');

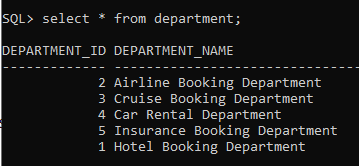
SQL> insert into department values (5, 'Insurance Booking Department');

SQL> insert into department values (1, 'Hotel Booking Department');





SQL> select \* from department;

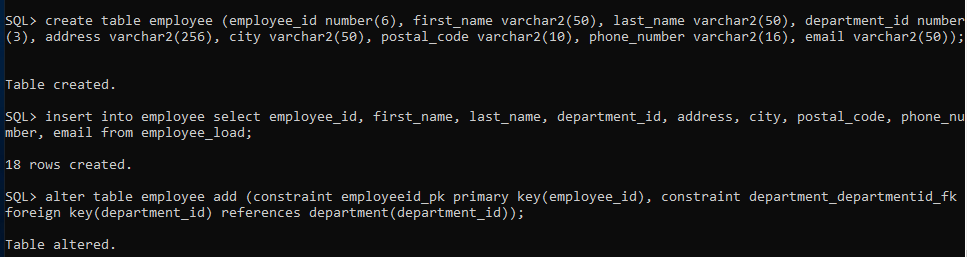


**Employee Table**

SQL> create table employee (employee\_id number(6), first\_name varchar2(50), last\_name varchar2(50), department\_id number(3), address varchar2(256), city varchar2(50), postal\_code varchar2(10), phone\_number varchar2(16), email varchar2(50));

SQL> insert into employee select employee\_id, first\_name, last\_name, department\_id, address, city, postal\_code, phone\_number, email from employee\_load;

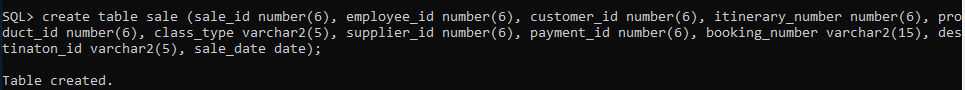
SQL> alter table employee add (constraint employeeid\_pk primary key(employee\_id), constraint department\_departmentid\_fk foreign key(department\_id) references department(department\_id));

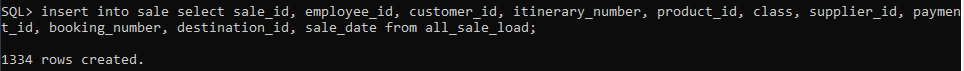


**Sale Table**

SQL> create table sale (sale\_id number(6), employee\_id number(6), customer\_id number(6), itinerary\_number number(6), product\_id number(6), class\_type varchar2(5), supplier\_id number(6), payment\_id number(6), booking\_number varchar2(15), destinaton\_id varchar2(5), sale\_date date);

SQL> insert into sale select sale\_id, employee\_id, customer\_id, itinerary\_number, product\_id, class, supplier\_id, payment\_id, booking\_number, destination\_id, sale\_date from all\_sale\_load;





SQL> alter table sale add constraint saleid\_pk primary key(sale\_id);

SQL> alter table sale add constraint employee\_employeeid\_fk foreign key(employee\_id) references employee(employee\_id);

SQL> alter table sale add constraint customer\_customerid\_fk foreign key(customer\_id) references customer(customer\_id);

SQL> alter table sale add constraint class\_class\_type\_fk foreign key(class\_type) references class(class\_type);

SQL> alter table sale rename column destinaton\_id to destination\_id;

SQL> alter table sale add constraint destination\_destinationid\_fk foreign key(destination\_id) references destination(destination\_id);

alter table sale add constraint payment\_paymentid\_fk1 foreign key (payment\_id) references payment(payment\_id);

delete from sale where supplier\_id = 150;

delete from sale where supplier\_id = 12;

select sale\_id from sale where supplier\_id=150;

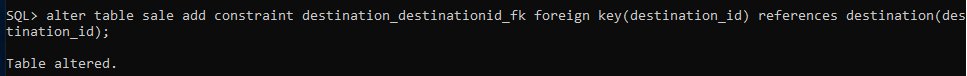
select sale\_id from sale where supplier\_id=12;

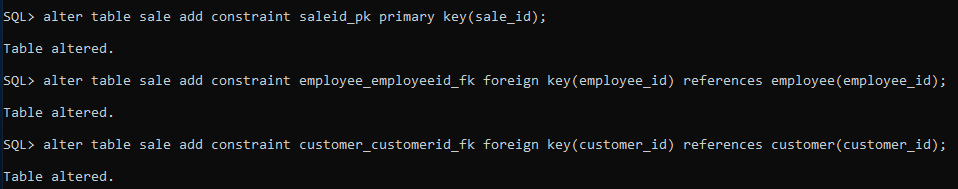
delete from tax\_quote where sale\_id=60;

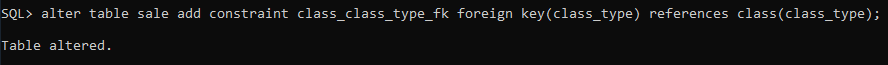
alter table sale add constraint MAIN\_SUPPLIER\_SUPPLIERID\_FK1 foreign key(supplier\_id) references main\_supplier (supplier\_id);

alter table sale add constraint booking\_booking\_number\_fk foreign key(booking\_number) references booking\_number(booking\_number);

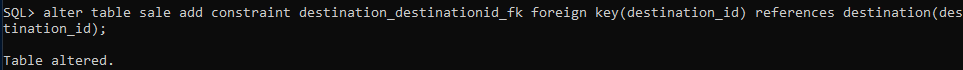
alter table sale add constraint product\_productid\_fk1 foreign key (product\_id) references product (product\_id);

alter table sale add constraint itinerarynumber\_itinerary\_number\_fk1 foreign key(itinerary\_number) references itinerary\_number(itinerary\_number);

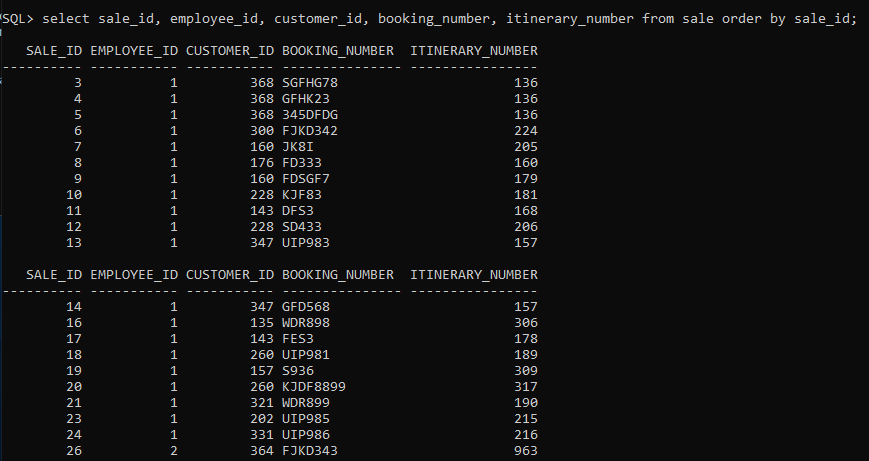








SQL> select sale\_id, employee\_id, customer\_id, booking\_number, itinerary\_number from sale order by sale\_id;



**Tax Quote Table**

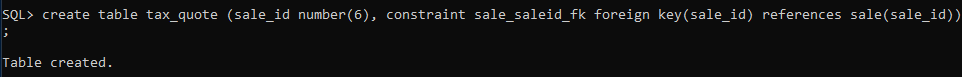
Tax quoted to agents by the supplier for different products.

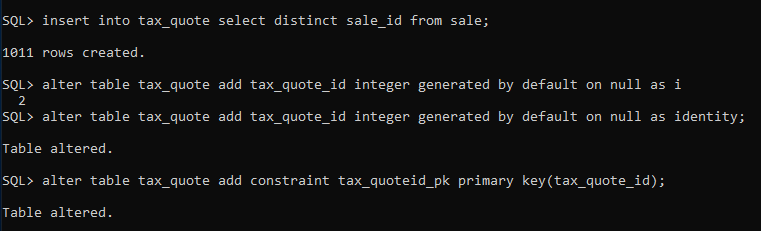
SQL> create table tax\_quote (sale\_id number(6), constraint sale\_saleid\_fk foreign key(sale\_id) references sale(sale\_id));drop

SQL> alter table tax\_quote add constraint tax\_quoteid\_pk primary key(tax\_quote\_id);

SQL> alter table tax\_quote add tax\_quote\_id integer generated by default on null as identity;

Commit;





SQL> select \* from tax\_quote;

