1. **Sample invoice/itinerary script**

**Using joins**

SET COLSEP " | "

SQL> select c.customer\_id, c.first\_name, c.last\_name, c.address, c.postal\_code, i.itinerary\_number, i.trip\_start, i.trip\_end, i.number\_of\_travellers, d.destination\_id, ms.supplier\_name, b.bill\_description, b.booking\_number, b.billed\_amount, b.total\_price, cc.card\_number from sale s

join customer c on s.customer\_id=c.customer\_id

join itinerary i on s.itinerary\_number=i.itinerary\_number

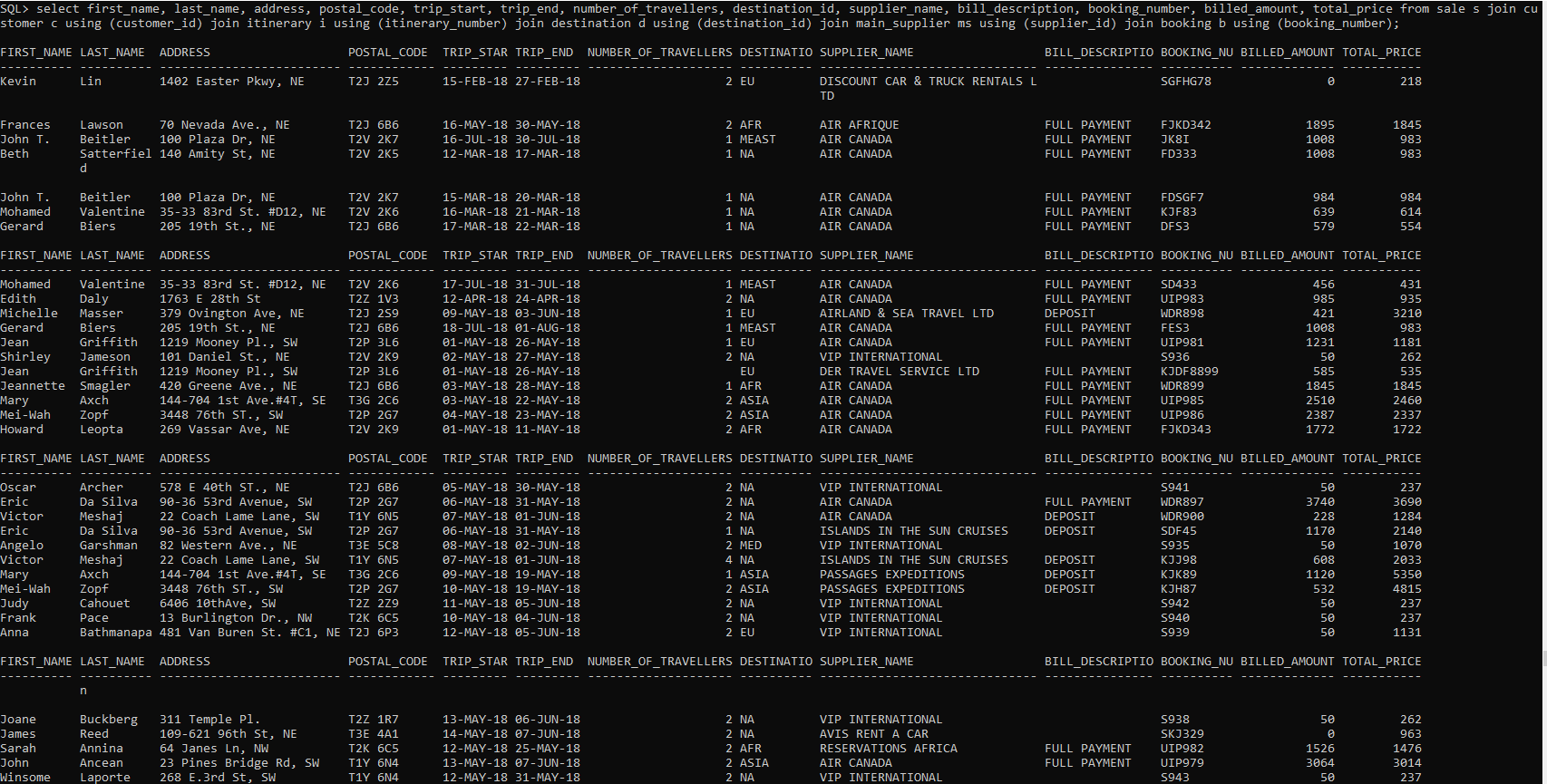
join destination d on s.destination\_id=d.destination\_id

join main\_supplier ms on s.supplier\_id=ms.supplier\_id

join booking b on s.booking\_number=b.booking\_number

join payment p on s.payment\_id=p.payment\_id

join cust\_credit\_card cc on p.cust\_credit\_card\_id=cc.cust\_credit\_card\_id;



**Stored procedure**

CREATE OR REPLACE PROCEDURE sp\_select\_customer (p\_cust\_details OUT SYS\_REFCURSOR)

IS

BEGIN

OPEN p\_cust\_details for SELECT c.customer\_id, c.first\_name, c.last\_name, c.address, c.postal\_code,

i.itinerary\_number, i.trip\_start, i.trip\_end, i.number\_of\_travellers,

d.destination\_id, ms.supplier\_name, b.bill\_description, b.booking\_number,

b.billed\_amount, b.total\_price, cc.card\_number

FROM sale s

JOIN customer c ON s.customer\_id=c.customer\_id

JOIN itinerary i ON s.itinerary\_number=i.itinerary\_number

JOIN destination d ON s.destination\_id=d.destination\_id

JOIN main\_supplier ms ON s.supplier\_id=ms.supplier\_id

JOIN booking b ON s.booking\_number=b.booking\_number

JOIN payment p ON s.payment\_id=p.payment\_id

JOIN cust\_credit\_card cc ON p.cust\_credit\_card\_id=cc.cust\_credit\_card\_id;

END;

/

CREATE OR REPLACE PROCEDURE get\_sale\_details

IS

-- Define cursor for the query

CURSOR sale\_cursor IS

SELECT c.customer\_id, c.first\_name, c.last\_name, c.address, c.postal\_code, i.itinerary\_number, i.trip\_start, i.trip\_end, i.number\_of\_travellers, d.destination\_id, ms.supplier\_name, b.bill\_description, b.booking\_number, b.billed\_amount, b.total\_price, cc.card\_number

FROM sale s

JOIN customer c ON s.customer\_id=c.customer\_id

JOIN itinerary i ON s.itinerary\_number=i.itinerary\_number

JOIN destination d ON s.destination\_id=d.destination\_id

JOIN main\_supplier ms ON s.supplier\_id=ms.supplier\_id

JOIN booking b ON s.booking\_number=b.booking\_number

JOIN payment p ON s.payment\_id=p.payment\_id

JOIN cust\_credit\_card cc ON p.cust\_credit\_card\_id=cc.cust\_credit\_card\_id;

-- Declare variables for the cursor

v\_customer\_id sale.customer\_id%TYPE;

v\_first\_name customer.first\_name%TYPE;

v\_last\_name customer.last\_name%TYPE;

v\_address customer.address%TYPE;

v\_postal\_code customer.postal\_code%TYPE;

v\_itinerary\_number itinerary.itinerary\_number%TYPE;

v\_trip\_start itinerary.trip\_start%TYPE;

v\_trip\_end itinerary.trip\_end%TYPE;

v\_number\_of\_travellers itinerary.number\_of\_travellers%TYPE;

v\_destination\_id destination.destination\_id%TYPE;

v\_supplier\_name main\_supplier.supplier\_name%TYPE;

v\_bill\_description booking.bill\_description%TYPE;

v\_booking\_number booking.booking\_number%TYPE;

v\_billed\_amount booking.billed\_amount%TYPE;

v\_total\_price booking.total\_price%TYPE;

v\_card\_number cust\_credit\_card.card\_number%TYPE;

BEGIN

-- Open the cursor

OPEN sale\_cursor;

-- Loop through the cursor and output the results

LOOP

FETCH sale\_cursor INTO v\_customer\_id, v\_first\_name, v\_last\_name, v\_address, v\_postal\_code, v\_itinerary\_number, v\_trip\_start, v\_trip\_end, v\_number\_of\_travellers, v\_destination\_id, v\_supplier\_name, v\_bill\_description, v\_booking\_number, v\_billed\_amount, v\_total\_price, v\_card\_number;

EXIT WHEN sale\_cursor%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE(v\_customer\_id || ', ' || v\_first\_name || ' ' || v\_last\_name || ', ' || v\_address || ', ' || v\_postal\_code || ', ' || v\_itinerary\_number || ', ' || v\_trip\_start || ', ' || v\_trip\_end || ', ' || v\_number\_of\_travellers || ', ' || v\_destination\_id || ', ' || v\_supplier\_name || ', ' || v\_bill\_description || ', ' || v\_booking\_number || ', ' || v\_billed\_amount || ', ' || v\_total\_price || ', ' || v\_card\_number);

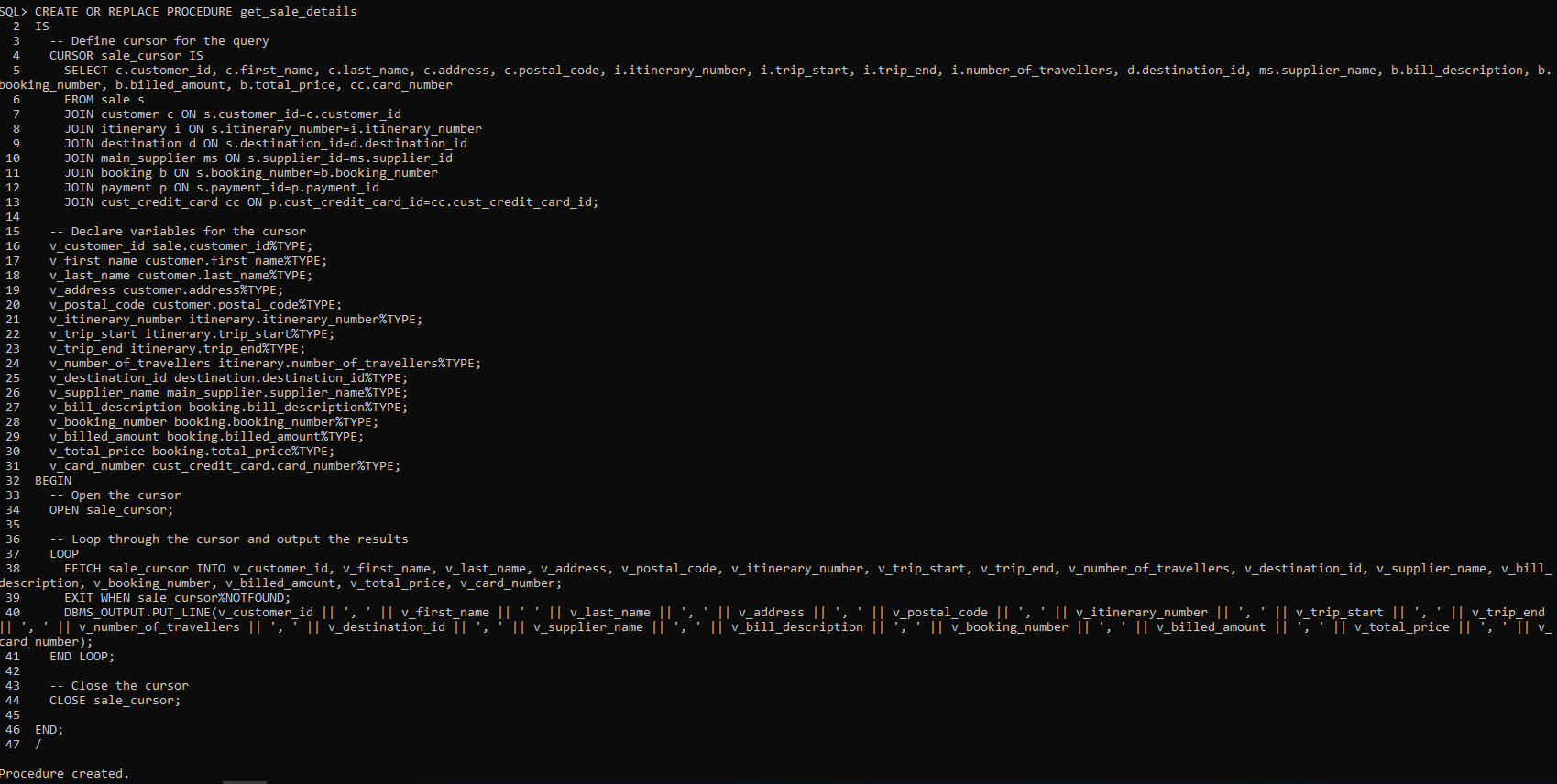
END LOOP;

-- Close the cursor

CLOSE sale\_cursor;

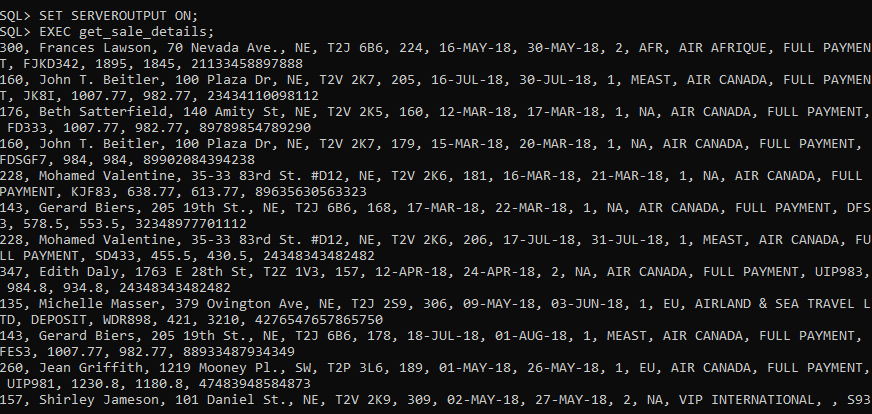
END;

/



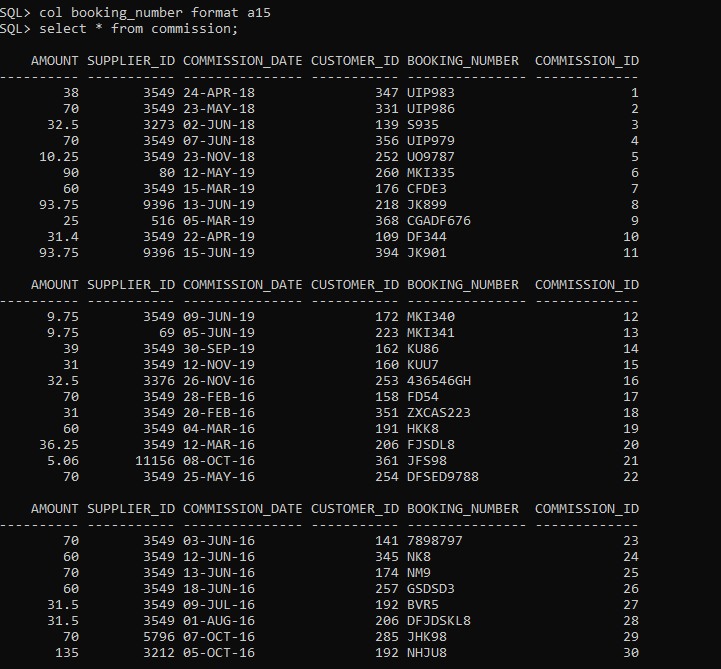
SQL> SET SERVEROUTPUT ON;

SQL> EXEC get\_sale\_details;



1. **Commission report script**

select \* from commission;



1. **Entry of a new customer**

SQL> CREATE SEQUENCE seq\_customers START WITH 403 INCREMENT BY 1 NOCACHE NOCYCLE;

ACCEPT first\_name PROMPT "Enter first name: "

ACCEPT last\_name PROMPT 'Enter last name: '

ACCEPT email PROMPT 'Enter email: '

ACCEPT home\_phone PROMPT 'Enter phone number: '

ACCEPT business\_phone PROMPT 'Enter business phone: '

ACCEPT date\_of\_birth PROMPT 'Enter DOB (Example 24-DEC-1998: '

ACCEPT address PROMPT 'Enter address: '

ACCEPT city PROMPT 'Enter city: '

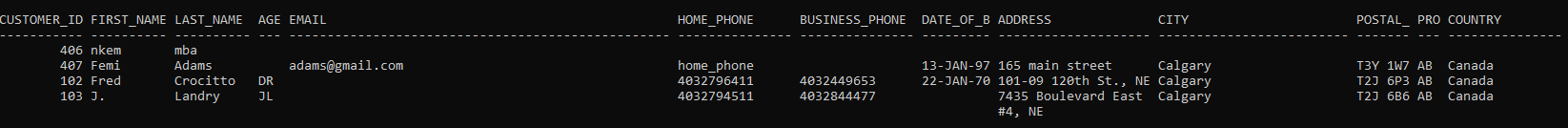
ACCEPT postal\_code PROMPT 'Enter postal code: '

ACCEPT province PROMPT 'Enter province: '

ACCEPT country PROMPT 'Enter country: '

INSERT INTO customer (customer\_id, first\_name, last\_name, email, home\_phone, business\_phone, date\_of\_birth, address, city, postal\_code, province, country) VALUES (seq\_customers.NEXTVAL, '&first\_name', '&last\_name', '&email', 'home\_phone', '&business\_phone', '&date\_of\_birth', '&address', '&city', '&postal\_code', '&province', '&country');

@C:\Temp\addcustomer.sql



select \* from customer where first\_name = 'Femi';

