



UTM
UNIVERSITI TEKNOLOGI MALAYSIA

FACULTY OF COMPUTING
UTM Johor Bahru

SECD2523
SECTION 10

LECTURER: ROZILAWATI BINTI DOLLAH @ MD ZAIN

LAB EXERCISE 4

NAME: NUR AQILA NATASHA BINTI HAZIDI
MATRIC NUMBERS : A22EC0243

Section 6 Lesson 9 Exercise 1: Joining Tables Using JOIN Write SELECT Statements Using Data From Multiple Tables Using Equijoins and Non-Equijoins (S6L9 Objective 1)

In this exercise you will write SELECT statements to access data from more than one table.

Part 1: Creating Natural Joins.

1. Display all of the information about sales representatives and their addresses using a natural join.

The screenshot shows the APEX SQL Workshop interface. The SQL Commands panel contains the following query:

```
1 SELECT *
2 FROM sales_representatives
3 NATURAL JOIN sales_rep_addresses;
```

The Results panel displays the output of the query, showing all columns from both tables. The data is as follows:

ID	EMAIL	FIRST_NAME	LAST_NAME	PHONE_NUMBER	COMMISSION_RATE	SUPERVISOR_ID	ADDRESS_LINE_1	ADDRESS_LINE_2	CITY	ZIP_CODE
sr01	chray@obl.com	Charles	Raymond	0134598761	10	sr01	12 Cherry Lane	Denton	Detroit	DT48211
sr03	bspeed@obl.com	Barry	Speed	0134598763	5	sr01	12 Junction Row	Skinflats	Detroit	DT52564
sr02	vwright@obl.com	Victoria	Wright	0134598762	5	sr01	87 Blossom Hill	Uptown	Detroit	DT52314

2. Adapt the query from the previous question to only show the id, first name, last name, address line 1, address line 2, city, email and phone_number for the sales representatives.

The screenshot shows the APEX SQL Workshop interface. The SQL Commands panel contains the following query:

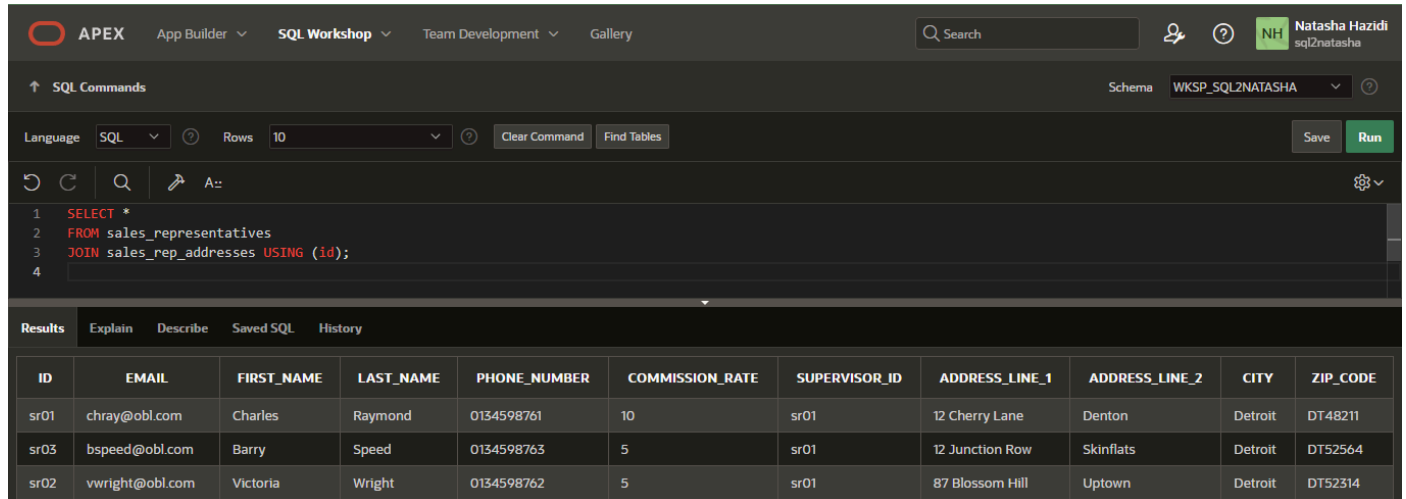
```
1 SELECT
2     id,
3     first_name, last_name, address_line_1, address_line_2, city, email, phone_number
4 FROM
5     sales_representatives
6 NATURAL JOIN
7     sales_rep_addresses ;
```

The Results panel displays the output of the query, showing only the specified columns. The data is as follows:

ID	FIRST_NAME	LAST_NAME	ADDRESS_LINE_1	ADDRESS_LINE_2	CITY	EMAIL	PHONE_NUMBER
sr01	Charles	Raymond	12 Cherry Lane	Denton	Detroit	chray@obl.com	0134598761
sr03	Barry	Speed	12 Junction Row	Skinflats	Detroit	bspeed@obl.com	0134598763
sr02	Victoria	Wright	87 Blossom Hill	Uptown	Detroit	vwright@obl.com	0134598762

Part 2: Creating Joins with the USING Clause

1. Adapt the previous query answer to use the USING clause instead of a natural join.



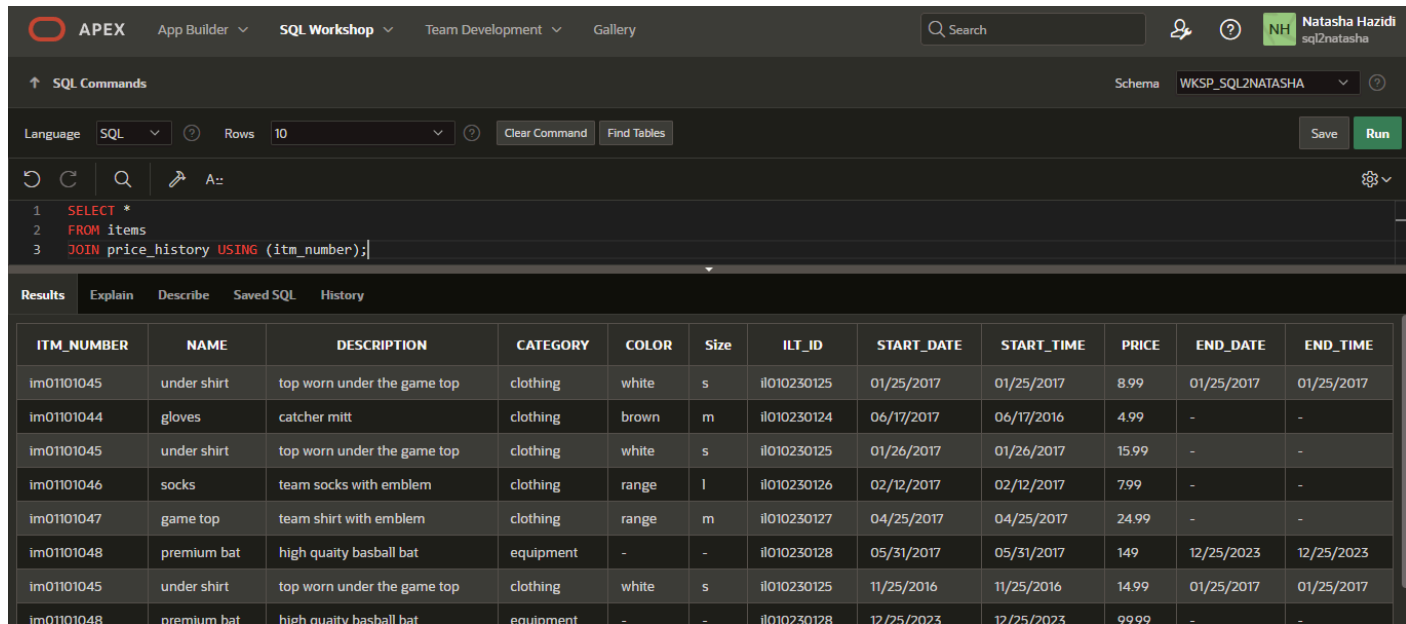
The screenshot shows the APEX SQL Workshop interface. The SQL command area contains the following query:

```
1 SELECT *
2 FROM sales_representatives
3 JOIN sales_rep_addresses USING (id);
4
```

The Results tab displays the following data:

ID	EMAIL	FIRST_NAME	LAST_NAME	PHONE_NUMBER	COMMISSION_RATE	SUPERVISOR_ID	ADDRESS_LINE_1	ADDRESS_LINE_2	CITY	ZIP_CODE
sr01	chray@obl.com	Charles	Raymond	0134598761	10	sr01	12 Cherry Lane	Denton	Detroit	DT48211
sr03	bspeed@obl.com	Barry	Speed	0134598763	5	sr01	12 Junction Row	Skinflats	Detroit	DT52564
sr02	vwright@obl.com	Victoria	Wright	0134598762	5	sr01	87 Blossom Hill	Uptown	Detroit	DT52314

2. Display all of the information about items and their price history by joining the items and price_history tables



The screenshot shows the APEX SQL Workshop interface. The SQL command area contains the following query:

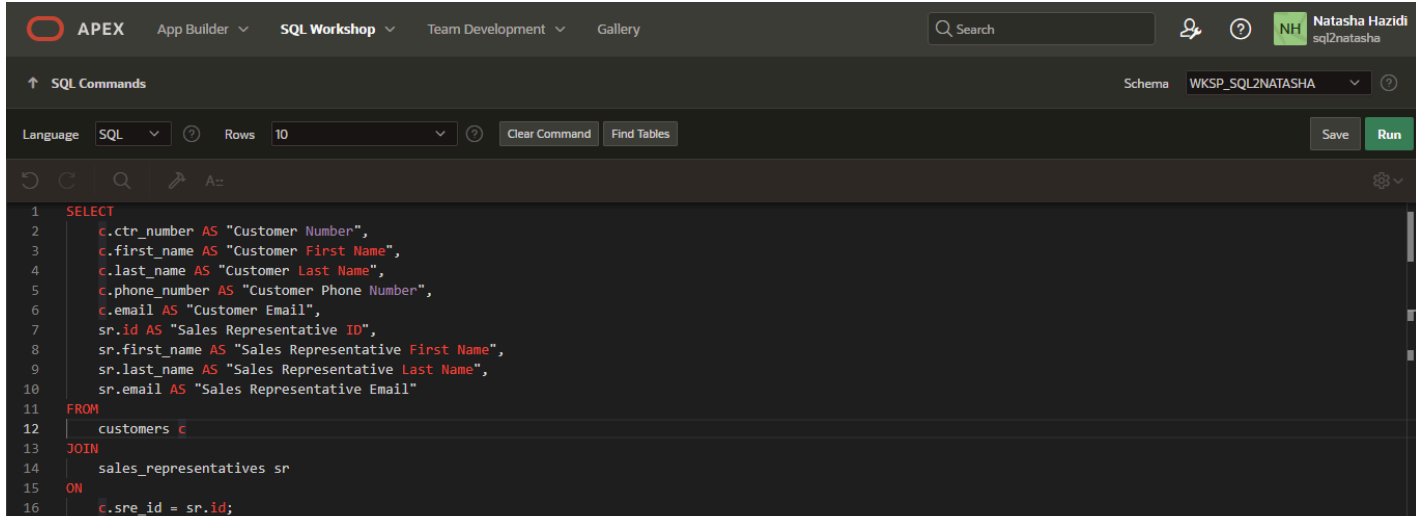
```
1 SELECT *
2 FROM items
3 JOIN price_history USING (itm_number);
```

The Results tab displays the following data:

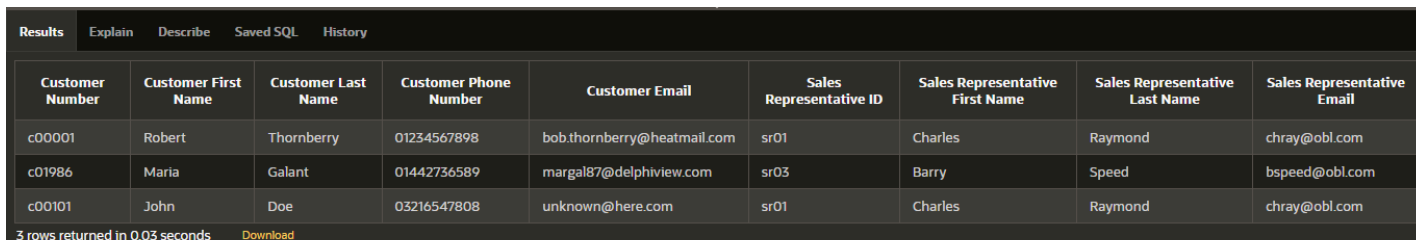
ITM_NUMBER	NAME	DESCRIPTION	CATEGORY	COLOR	Size	ILT_ID	START_DATE	START_TIME	PRICE	END_DATE	END_TIME
im01101045	under shirt	top worn under the game top	clothing	white	s	il010230125	01/25/2017	01/25/2017	8.99	01/25/2017	01/25/2017
im01101044	gloves	catcher mitt	clothing	brown	m	il010230124	06/17/2017	06/17/2016	4.99	-	-
im01101045	under shirt	top worn under the game top	clothing	white	s	il010230125	01/26/2017	01/26/2017	15.99	-	-
im01101046	socks	team socks with emblem	clothing	range	l	il010230126	02/12/2017	02/12/2017	7.99	-	-
im01101047	game top	team shirt with emblem	clothing	range	m	il010230127	04/25/2017	04/25/2017	24.99	-	-
im01101048	premium bat	high quality baseball bat	equipment	-	-	il010230128	05/31/2017	05/31/2017	149	12/25/2023	12/25/2023
im01101045	under shirt	top worn under the game top	clothing	white	s	il010230125	11/25/2016	11/25/2016	14.99	01/25/2017	01/25/2017
im01101048	premium bat	high quality baseball bat	equipment	-	-	il010230128	12/25/2023	12/25/2023	99.99	-	-

Part 3: Creating Joins with the ON Clause

1. Use an ON clause to join the customer and sales representative table so that you display the customer number, customer first name, customer last name, customer phone number, customer email, sales representative id, sales representative first name, sales representative last name and sales representative email. You will need to use a table alias in your answer as both tables have columns with the same name.



```
1 SELECT
2     c.ctr_number AS "Customer Number",
3     c.first_name AS "Customer First Name",
4     c.last_name AS "Customer Last Name",
5     c.phone_number AS "Customer Phone Number",
6     c.email AS "Customer Email",
7     sr.id AS "Sales Representative ID",
8     sr.first_name AS "Sales Representative First Name",
9     sr.last_name AS "Sales Representative Last Name",
10    sr.email AS "Sales Representative Email"
11 FROM
12     customers c
13 JOIN
14     sales_representatives sr
15 ON
16     c.sre_id = sr.id;
```

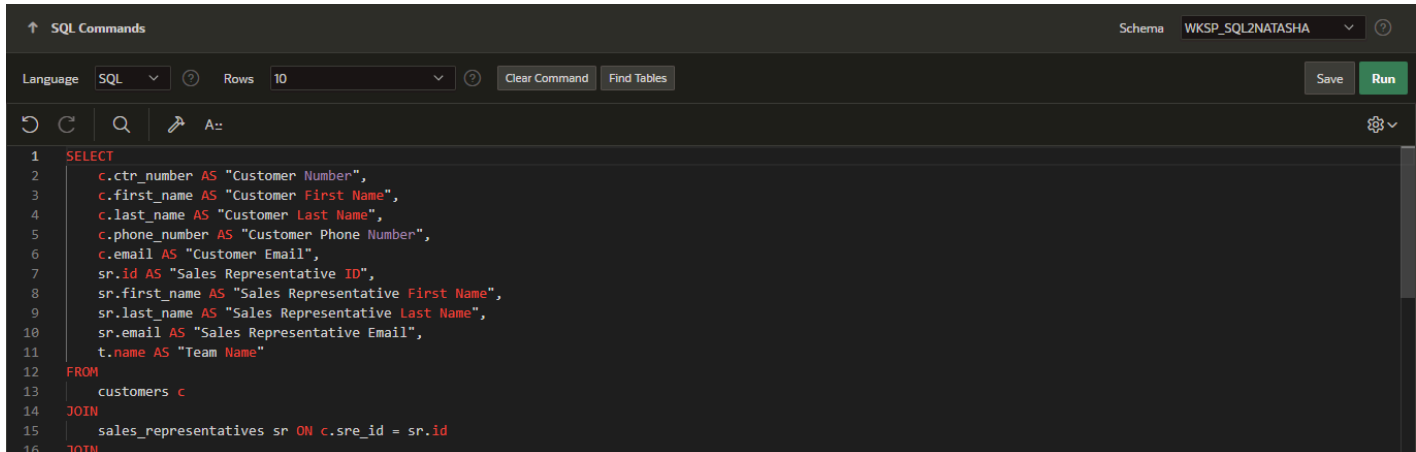


Customer Number	Customer First Name	Customer Last Name	Customer Phone Number	Customer Email	Sales Representative ID	Sales Representative First Name	Sales Representative Last Name	Sales Representative Email
c00001	Robert	Thornberry	01234567898	bob.thornberry@heatmail.com	sr01	Charles	Raymond	chray@obl.com
c01986	Maria	Galant	01442736589	margal87@delphiview.com	sr03	Barry	Speed	bspeed@obl.com
c00101	John	Doe	03216547808	unknown@here.com	sr01	Charles	Raymond	chray@obl.com

3 rows returned in 0.03 seconds

Part 4- Creating Three-Way Joins with the ON Clause

1. Using the answer to Task 3 add a join that will allow the team name that the customer represents to be included in the results.



```
1 SELECT
2     c.ctr_number AS "Customer Number",
3     c.first_name AS "Customer First Name",
4     c.last_name AS "Customer Last Name",
5     c.phone_number AS "Customer Phone Number",
6     c.email AS "Customer Email",
7     sr.id AS "Sales Representative ID",
8     sr.first_name AS "Sales Representative First Name",
9     sr.last_name AS "Sales Representative Last Name",
10    sr.email AS "Sales Representative Email",
11    t.name AS "Team Name"
12 FROM
13     customers c
14 JOIN
15     sales_representatives sr ON c.sre_id = sr.id
16 JOIN
```

SQL Commands									
Schema: WKSP_SQL2NATASHA									
Language: SQL Rows: 10									
<pre> 12 FROM 13 customers c 14 JOIN 15 sales_representatives sr ON c.sre_id = sr.id 16 JOIN 17 teams t ON c.tem_id = t.id; 18 </pre>									
Results									
Customer Number	Customer First Name	Customer Last Name	Customer Phone Number	Customer Email	Sales Representative ID	Sales Representative First Name	Sales Representative Last Name	Sales Representative Email	Team Name
c00001	Robert	Thornberry	01234567898	bob.thornberry@heatmail.com	sr01	Charles	Raymond	chray@obl.com	Rockets
c01986	Maria	Galant	01442736589	margal87@delphiview.com	sr03	Barry	Speed	bspeed@obl.com	Rovers
c00101	John	Doe	03216547808	unknown@here.com	sr01	Charles	Raymond	chray@obl.com	Celtics

Part 5: Applying Additional Conditions to a Join

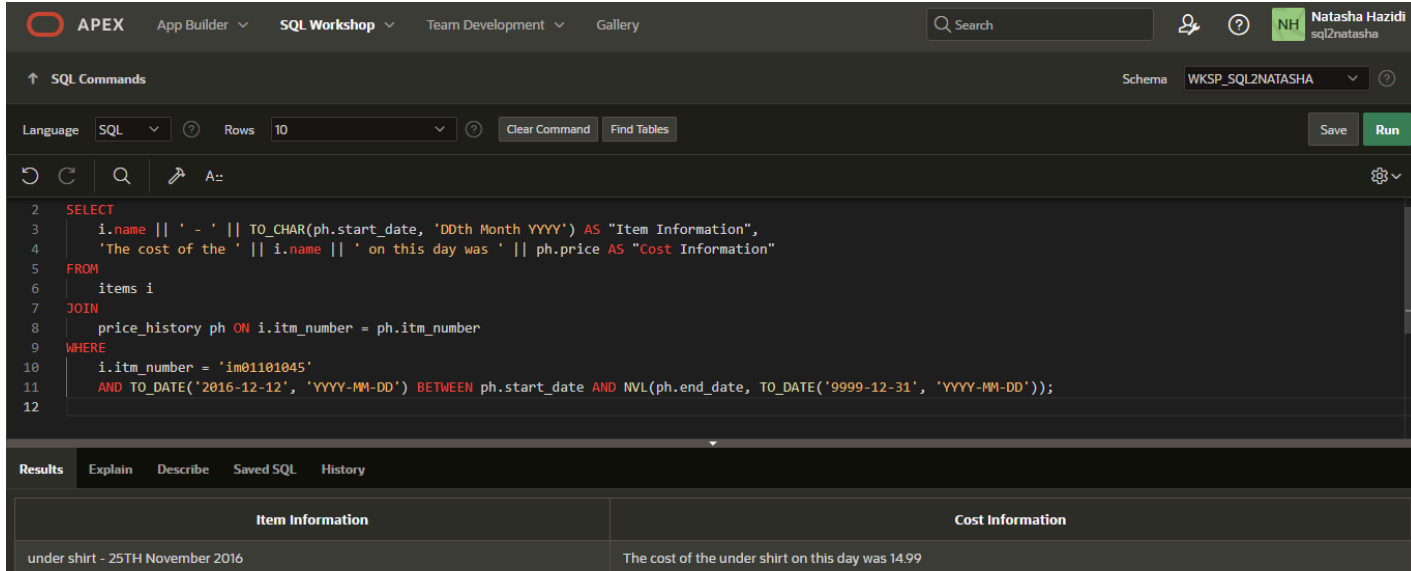
1. Using the answer to Task 4 add an additional condition to only show the results for the customer that has the number - c00001.

SQL Commands									
Schema: WKSP_SQL2NATASHA									
Language: SQL Rows: 10									
<pre> 1 SELECT 2 c.ctr_number AS "Customer Number", 3 c.first_name AS "Customer First Name", 4 c.last_name AS "Customer Last Name", 5 c.phone_number AS "Customer Phone Number", 6 c.email AS "Customer Email", 7 sr.id AS "Sales Representative ID", 8 sr.first_name AS "Sales Representative First Name", 9 sr.last_name AS "Sales Representative Last Name", 10 sr.email AS "Sales Representative Email", 11 t.name AS "Team Name" 12 FROM 13 customers c 14 JOIN 15 sales_representatives sr ON c.sre_id = sr.id 16 JOIN 17 teams t ON c.tem_id = t.id 18 WHERE 19 c.ctr_number = 'c00001'; </pre>									
Results									
Customer Number	Customer First Name	Customer Last Name	Customer Phone Number	Customer Email	Sales Representative ID	Sales Representative First Name	Sales Representative Last Name	Sales Representative Email	Team Name
c00001	Robert	Thornberry	01234567898	bob.thornberry@heatmail.com	sr01	Charles	Raymond	chray@obl.com	Rockets

Part 6: Retrieving Records with Nonequiijoins

1. Write a query that will display name and cost of the item with the number im01101045 on the 12th of December 2016. The output of the query should look like this:

The cost of the under shirt on this day was 14.99



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', and 'Gallery'. A search bar and user profile 'Natasha Hazidi' are on the right. The 'SQL Commands' tab is active, showing a query in the editor. The query is a SELECT statement that concatenates item names with their prices and dates, formatted as 'Item Information' and 'Cost Information'. It filters for item number 'im01101045' and the date '2016-12-12'. The 'Run' button is highlighted in green. Below the editor, the 'Results' tab is selected, displaying a table with two columns: 'Item Information' and 'Cost Information'. The table contains one row of data.

```
2 SELECT
3     i.name || ' - ' || TO_CHAR(ph.start_date, 'DDth Month YYYY') AS "Item Information",
4     'The cost of the ' || i.name || ' on this day was ' || ph.price AS "Cost Information"
5 FROM
6     items i
7 JOIN
8     price_history ph ON i.itm_number = ph.itm_number
9 WHERE
10    i.itm_number = 'im01101045'
11    AND TO_DATE('2016-12-12', 'YYYY-MM-DD') BETWEEN ph.start_date AND NVL(ph.end_date, TO_DATE('9999-12-31', 'YYYY-MM-DD'));
12
```

Item Information	Cost Information
under shirt - 25TH November 2016	The cost of the under shirt on this day was 14.99

Part 1 : Use a Self-Join to Join a Table to Itself (S6L9 Objective 2)

App Builder

SQL Workshop

Team Development

Gallery

NH

Natasha Hazidi

sqlZnataasha

↑

SQL Commands

Schema

WKSP_SQLZNATASHA

Language

SQL

Rows

10

Clear Command

Find Tables

Save

Run

↶

↷

🔍

🔧

A..

2

3

4

5

6

7

8

9

```

SELECT
  rep.first_name || ' ' || rep.last_name AS "Rep",
  sup.first_name || ' ' || sup.last_name AS "Supervisor"
FROM
  sales_representatives rep
LEFT JOIN
  sales_representatives sup ON rep.supervisor_id = sup.id;

```

Results

Explain

Describe

Saved SQL

History

Rep	Supervisor
Charles Raymond	Charles Raymond
Barry Speed	Charles Raymond
Victoria Wright	Charles Raymond

1. Write a query that will display all of the team and customer information even if there is no match with the table on the left (team).

[illegible]

Part 3 : Generating a Cartesian Product (S6L9 Objective 4)

1. Create a Cartesian product between the customer and sales representative tables.

APEX

App Builder

SQL Workshop

Team Development

Gallery

Search

NH

Natasha Hazidi

sql2natasha

SQL Commands

Schema

WKSP_SQL2NATASHA

Language

SQL

Rows

10

Clear Command

Find Tables

Save

Run

A::

1

SELECT*

2

FROM customers

3

CROSS JOIN sales_representatives;

Results

Explain

Describe

Saved SQL

History

CTR_NUMBER	EMAIL	FIRST_NAME	LAST_NAME	PHONE_NUMBER	CURRENT_BALANCE	SRE_ID	TEM_ID	LOYALTY_CARD_NUMBER	ID	EMAIL
c00001	bob.thornberry@heatmail.com	Robert	Thornberry	01234567898	50	sr01	t001	-	sr01	chray@obl.com
c00103	MurciaA@globaltech.com	Andrew	Murcia	07715246890	50	-	-	lc2341	sr01	chray@obl.com
c02001	brianrog@hootech.com	Brian	Rogers	01654564898	50	-	-	lc4587	sr01	chray@obl.com
c00012	Jjones@freemail.com	Jennifer	Jones	01505214598	50	-	-	lc1015	sr01	chray@obl.com
c01986	margal87@delphiview.com	Maria	Galant	01442736589	50	sr03	t003	-	sr01	chray@obl.com
c00101	unknown@here.com	John	Doe	03216547808	50	sr01	t002	-	sr01	chray@obl.com
c00001	bob.thornberry@heatmail.com	Robert	Thornberry	01234567898	50	sr01	t001	-	sr03	bspeed@obl.com
c00103	MurciaA@globaltech.com	Andrew	Murcia	07715246890	50	-	-	lc2341	sr03	hsneed@obl.com

APEX

App Builder

SQL Workshop

Team Development

Gallery

Search

NH

Natasha Hazidi

sql2natasha

SQL Commands

Schema

WKSP_SQL2NATASHA

Language

SQL

Rows

10

Clear Command

Find Tables

Save

Run

A::

1

SELECT*

Results

Explain

Describe

Saved SQL

History

CTR_NUMBER	EMAIL	FIRST_NAME	LAST_NAME	PHONE_NUMBER	CURRENT_BALANCE	SRE_ID	TEM_ID	LOYALTY_CARD_NUMBER	ID	EMAIL
c00001	bob.thornberry@heatmail.com	Robert	Thornberry	01234567898	50	sr01	t001	-	sr01	chray@obl.com
c00103	MurciaA@globaltech.com	Andrew	Murcia	07715246890	50	-	-	lc2341	sr01	chray@obl.com
c02001	brianrog@hootech.com	Brian	Rogers	01654564898	50	-	-	lc4587	sr01	chray@obl.com
c00012	Jjones@freemail.com	Jennifer	Jones	01505214598	50	-	-	lc1015	sr01	chray@obl.com
c01986	margal87@delphiview.com	Maria	Galant	01442736589	50	sr03	t003	-	sr01	chray@obl.com
c00101	unknown@here.com	John	Doe	03216547808	50	sr01	t002	-	sr01	chray@obl.com
c00001	bob.thornberry@heatmail.com	Robert	Thornberry	01234567898	50	sr01	t001	-	sr03	bspeed@obl.com
c00103	MurciaA@globaltech.com	Andrew	Murcia	07715246890	50	-	-	lc2341	sr03	bspeed@obl.com
c02001	brianrog@hootech.com	Brian	Rogers	01654564898	50	-	-	lc4587	sr03	bspeed@obl.com
c00012	Jjones@freemail.com	Jennifer	Jones	01505214598	50	-	-	lc1015	sr03	bspeed@obl.com

More than 10 rows available. Increase rows selector to view more rows.