

CAB 4 LIN CHONG HUI A22ECO184

PART 1: CREATING NATURAL JOINS

① SELECT * from sales_representatives
Natural Joins sales_rep_address ;

② SELECT id, first_name, last_name, address_line_1, address_line_2, city, email, phone_number
FROM sales_representatives NATURAL JOIN sales_rep_addresses ;

PART 2: CREATING JOINS WITH THE USING CLAUSE

① SELECT id, first_name, last_name, address_line_1, address_line_2, city, email, phone_number
FROM sales_representative JOIN sales_rep_address USING (id);

② SELECT * from items JOIN price_history ON items.item_number = price_history.item_number;

PART 3 :

① SELECT c.ctr_number, c.first_name "Customer First Name", c.last_name "Customer last Name",
c.phone_number "Customer Phone number", c.email "Customer Email", s.id "Sales Representative ID",
s.first_name "Sales Representative First Name", s.last_name "Sales Representative Last Name",
s.email "Sales Representative Email" FROM customer C JOIN sales_representative s ON c.sre_id = s.id;

PART 4: CREATING THREE-WAY JOINS WITH THE ON CLAUSE

① SELECT c.ctr_number, c.first_name "Customer First Name", c.last_name "Customer Last Name",
c.phone_number "Customer Phone Number", c.email "Customer Email", s.id "Sales Representative ID",
s.first_name "Sales Representative First Name", s.last_name "Sales Representative Last Name",
s.email "Sales Representative Email", t.name "Team Name" FROM customer C JOIN sales_representative s
ON c.sreid = s.id JOIN team t ON c.team_id = t.id;

PART 5: APPLYING ADDITIONAL CONDITIONS TO A JOIN

① SELECT c.ctr_number, c.first_name "customer First Name", c.last_name "customer last Name",
c.phone_number "customer Phone Number", c.email "Customer Email", s.id "Sales Representative ID",
s.first_name "Sales Representative First Name", s.last_name "Sales Representative Last Name",
s.email "Sales Representative Email", t.name "TEAM Name" FROM Customer c JOIN sales-representatives ON
c.sre_id = s.id JOIN teams t ON c.team_id = t.id
WHERE c.ctr_number = 'C00001';

PART 6: RETRIEVING RECORD WITH NONEQUIJOINS

① SELECT 'The cost of the' || i.name || 'on this day was' || ph.price AS output from item;
LEFT JOIN price-history ph ON i.item_number = ph.item_number WHERE i.item_number
= 'IM01101045' AND TO_DATE ('12-Dec-2016', 'DD-Mon-YYYY') BETWEEN ph.start_date
AND NVL (ph.end_date, TO_DATE ('31-Dec-9999', 'DD-Mon-YYYY'));

PART 2:

PART 1: USE A SELF-JOIN TO JOIN A TABLE TO ITSELF

① SELECT s1.first_name || ' ' || s1.last_name AS Rep, s2.first_name || ' ' || s2.last_name AS
supervisor FROM sales-representative s1 JOIN sales-representative s2 ON s1.supervisor_id = s2.id;

PART 2: USE OUTER JOINS

① SELECT t*, c* FROM LEFT JOIN customers c ON t.id = c.team_id;

PART 3: GENERATING A CARTESIAN PRODUCT

① SELECT * FROM customer, sales-representative;