

```

--=====
-- Name: Meet Patel
-- Student Number: 169047180
-- Email: mspatel41@myseneca.ca
-- Section: NEE
-- LAB: 04
--=====

--1. Create a query that shows employee number, first name, last name, city,
phone number and postal code
--for all employees in France.
--a. Answer this question using an ANSI-89 Join
SELECT employeenumber, firstname, lastname, city, phone, postalcode
FROM offices, employees
WHERE employees.officecode = offices.officecode
AND LOWER (offices.country) LIKE 'france';
--b. Answer this question using an ANSI-92 Join
SELECT employeenumber,firstname, lastname, city, phone, postalcode
FROM offices INNER JOIN employees
ON employees.officecode = offices.officecode
WHERE LOWER (offices.country) LIKE 'france';

--2. Create a query that displays all payments made by customers from Canada.
--a. Sort the output by Customer Number.
--b. Only display the Customer Number, Customer Name, Payment Date and Amount.
--c. Make sure the date is displayed clearly to know what date it is. (i.e.
what date is 02-04-19??? Feb 4, 2019,
--April 2, 2019, April 19, 2002, .)
SELECT C.customernumber, C.customername, TO_CHAR(p.paymentdate, 'MON
DD,YYYY')paymentdate, P.amount
FROM payments P INNER JOIN customers C
ON P.customernumber = C.customernumber
AND LOWER(C.country) LIKE 'canada'
ORDER BY customernumber ASC;

--3. Create a query that shows all USA customers who have not made a payment.
--Display only the customer number and customer name sorted by customer number.
SELECT customernumber,customername
FROM customers FULL OUTER JOIN payments USING (customernumber)
WHERE payments.amount IS NULL AND LOWER (customers.country) LIKE 'usa'
ORDER BY customernumber DESC;

--4.
--a) Create a view (vwCustomerOrder) to list all orders with the following data
for all customers:
--Customer Number, Order number, order date, product name, quantity ordered, and
price for each product in every order.
CREATE OR REPLACE VIEW vwCustomerOrder AS
SELECT C.customername, C.customernumber, O.ordernumber, O.orderdate,
P.productname, OD.quantityordered, OD.priceeach
FROM customers C JOIN orders O
ON (C.customernumber = O.customernumber)
JOIN orderdetails OD ON (OD.ordernumber = O.ordernumber)
JOIN products P ON (P.productcode = OD.productcode);

--b) Write a statement to view the results of the view just created.
SELECT * FROM vwcustomerorder;

--5. Using the vwCustomerOrder view, display the order information for
customer number 124.
--Sort the output based on order number and then order line number.
--(Yes, I know orderLineNumber is not in the view)

--OPTION 1!

```

```

SELECT *
FROM vwcustomerorder V INNER JOIN orderdetails D
ON D.ordernumber = V.ordernumber
WHERE customernumber = 124
ORDER BY V.ordernumber, D.orderlinenumber;
--OPTION 2!
SELECT *
FROM vwcustomerorder
WHERE customernumber=124
ORDER BY ordernumber;
-----
SELECT
OD.orderlinenumber,vw.customername,vw.customernumber,vw.orderdate,vw.ordernumber
,
vw.priceeach,vw.quantityordered,vw.productname
FROM orderdetails OD JOIN vwcustomerorder vw ON (OD.ordernumber =
vw.ordernumber)
ORDER BY OD.ORDERLINENUMBER;
--sorry for answering in two option, was not sure what exactly the question
wanted as OUTPUT.

--6. Create a query that displays the customer number, first name, last name,
phone,
--and credit limits for all customers who do not have any orders.
SELECT C.customernumber, C.customername, C.phone, C.creditlimit
FROM customers C FULL OUTER JOIN orders O
ON (C.customernumber = O.customernumber)
WHERE O.ordernumber IS NULL;

--7. Create a view (vwEmployeeManager) to display all information of all
employees
--and the name and the last name of their managers if there is any manager that
the employee reports to.
--Include all employees, including those who do not report to anyone.
CREATE OR REPLACE VIEW vwEmployeeManager AS
SELECT * FROM employees
WHERE reportsto IS NOT NULL OR reportsto IS NULL;
-----
SELECT * from vwemployeemanager, CONCAT(vwemployeemanager.firstname,
vwemployeemanager.lastname) AS managername
WHERE ;
--I would not say that this question does not have an answer, but yes i googled
a lot but couldn't find nothing!!!!!!

--8. Modify the employee_manager view so the view returns only employee
information for employees
--who have a manager.Do not DROP and recreate the view 0 modify it. (Google is
your friend).
CREATE OR REPLACE VIEW vwemployeemanager AS
SELECT *
FROM employees
WHERE reportsto IS NOT NULL;
-----
SELECT * FROM vwemployeemanager;

--9. Drop both customer_order and employee_manager views.
DROP VIEW vwcustomerorder;
DROP VIEW vwEmployeeManager;

```