|  |
| --- |
| -- CECS323 Practice SQL |
|  |  |
|  |  |
|  | -- Selects |
|  | -- 1. List all the data in the classic models database: |
|  |  |
|  | -- a) Product Lines (7) |
|  | SELECT \* FROM PRODUCTLINES; |
|  |  |
|  | -- b) Product (110); |
|  | SELECT \* FROM PRODUCTS; |
|  |  |
|  | -- c) Employees (23) |
|  | SELECT \* FROM EMPLOYEES; |
|  |  |
|  | -- d) Offices (7) |
|  | SELECT \* FROM OFFICES; |
|  |  |
|  | -- e) Customers (122) |
|  | SELECT \* FROM CUSTOMERS; |
|  |  |
|  | -- f) Orders (326) |
|  | SELECT \* FROM ORDERS; |
|  |  |
|  | -- g) Orderdetails (2996) |
|  | SELECT \* FROM ORDERDETAILS; |
|  |  |
|  | -- h) Payments (273) |
|  | SELECT \* FROM PAYMENTS; |
|  |  |
|  | -- 2. Select customer name from customer. Sort by customer name (122) |
|  | SELECT CUSTOMERNAME FROM CUSTOMERS |
|  | ORDER BY CUSTOMERNAME; |
|  |  |
|  | -- 3. List each of the different status that an order may be in (6) |
|  | SELECT DISTINCT STATUS from ORDERS; |
|  |  |
|  | -- 4. List firstname and lastname for each employee. Sort by lastname then firstname (23) |
|  | SELECT FIRSTNAME, LASTNAME from EMPLOYEES |
|  | ORDER BY LASTNAME, FIRSTNAME; |
|  |  |
|  | -- 5. List all the employee job titles (7) |
|  | SELECT DISTINCT JOBTITLE from EMPLOYEES; |
|  |  |
|  | -- 6. List all products along with their product scale (110) |
|  | SELECT PRODUCTNAME, PRODUCTSCALE from PRODUCTS; |
|  |  |
|  | -- 7. List all the territories where we have offices (4) |
|  | SELECT DISTINCT TERRITORY from OFFICES; |
|  |  |
|  | -- Where Clause |
|  | -- 8. select contact firstname, contact lastname and credit limit for all customers where credit |
|  | -- limit > 50000 (85) |
|  | SELECT CONTACTFIRSTNAME, CONTACTLASTNAME, CREDITLIMIT FROM CUSTOMERS |
|  | WHERE CREDITLIMIT > 50000; |
|  |  |
|  | -- 9. select customers who do not have a credit limit (0.00) (24) |
|  | SELECT CONTACTFIRSTNAME, CONTACTLASTNAME, CREDITLIMIT FROM CUSTOMERS |
|  | WHERE CREDITLIMIT = 0; |
|  |  |
|  | -- 10. List all offices not in the USA (4) |
|  | SELECT CITY, COUNTRY FROM OFFICES |
|  | WHERE COUNTRY != 'USA'; |
|  |  |
|  | -- 11. List orders made between June 16, 2014 and July 7, 2014 (8) |
|  | SELECT ORDERNUMBER, ORDERDATE from ORDERS |
|  | WHERE |
|  | month(ORDERDATE) = 6 |
|  | and day(ORDERDATE) > 16 |
|  | or |
|  | month(ORDERDATE) = 7 |
|  | and day(ORDERDATE) < 7 |
|  | and year(ORDERDATE) = 2014; |
|  |  |
|  | -- 12. List products that we need to reorder (quantityinstock < 1000) (12) |
|  | SELECT PRODUCTNAME |
|  | FROM PRODUCTS |
|  | WHERE QUANTITYINSTOCK < 1000; |
|  |  |
|  | -- 13. List all orders that shipped after the required date (1) |
|  | SELECT ORDERNUMBER, ORDERDATE, REQUIREDDATE, SHIPPEDDATE |
|  | FROM ORDERS |
|  | WHERE SHIPPEDDATE > REQUIREDDATE; |
|  |  |
|  | -- 14. List all customers who have the word ‘Mini’ in their name (10) |
|  | SELECT CUSTOMERNAME FROM CUSTOMERS |
|  | WHERE CUSTOMERNAME LIKE '%Mini%'; |
|  |  |
|  | -- 15. List all products supplied by ‘Highway 66 Mini Classics’ (9) |
|  | SELECT PRODUCTNAME FROM PRODUCTS |
|  | WHERE PRODUCTVENDOR = 'Highway 66 Mini Classics'; |
|  |  |
|  | -- 16. List all product not supplied by ‘Highway 66 Mini Classics’ (101) |
|  | SELECT PRODUCTNAME FROM PRODUCTS |
|  | WHERE PRODUCTVENDOR != 'Highway 66 Mini Classics'; |
|  |  |
|  | -- 17. List all employees that don't have a manager (1) |
|  | SELECT FIRSTNAME, LASTNAME FROM EMPLOYEES |
|  | WHERE REPORTSTO IS NULL; |
|  |  |
|  | -- Natural Join |
|  | -- 18. Display every order along with the details of that order for order numbers 10270, 10272, |
|  | -- 10279 (23) |
|  | SELECT \* FROM ORDERS |
|  | NATURAL JOIN ORDERDETAILS |
|  | WHERE ORDERNUMBER = 10270 or ORDERNUMBER = 10272 or ORDERNUMBER = 10279; |
|  |  |
|  | -- Hint: this can be done two ways. Try both of them. Which is easier if you have a large number of |
|  | -- selection criteria? |
|  | -- 19. List of productlines and vendors that supply the products in that productline. (65) |
|  | SELECT DISTINCT PRODUCTLINE, PRODUCTVENDOR FROM PRODUCTLINES |
|  | NATURAL JOIN PRODUCTS; |
|  |  |
|  | -- Inner Join |
|  | -- 20. select customers that live in the same state as one of our offices (26) |
|  | SELECT CUSTOMERNAME, STATE FROM CUSTOMERS |
|  | INNER JOIN OFFICES USING (STATE); |
|  |  |
|  | -- 21. select customers that live in the same state as their employee representative works (26) |
|  | SELECT CUSTOMERNAME FROM CUSTOMERS |
|  | INNER JOIN EMPLOYEES ON CUSTOMERS.SALESREPEMPLOYEENUMBER = EMPLOYEES.EMPLOYEENUMBER |
|  | INNER JOIN OFFICES ON EMPLOYEES.OFFICECODE = OFFICES.OFFICECODE |
|  | WHERE CUSTOMERS.STATE = OFFICES.STATE; |
|  |  |
|  | -- Multi-join |
|  | -- 22. select customerName, orderDate, quantityOrdered, productLine, productName for all |
|  | -- orders made and shipped in 2015 (444) |
|  | SELECT CUSTOMERNAME, ORDERDATE, QUANTITYORDERED, PRODUCTLINE, PRODUCTNAME FROM CUSTOMERS |
|  | INNER JOIN ORDERS USING (CUSTOMERNUMBER) |
|  | INNER JOIN ORDERDETAILS USING (ORDERNUMBER) |
|  | INNER JOIN PRODUCTS USING (PRODUCTCODE) |
|  | INNER JOIN PRODUCTLINES USING (PRODUCTLINE) |
|  | WHERE year(ORDERDATE) = 2015 and year(SHIPPEDDATE) = 2015; |
|  |  |
|  | -- Outer Join |
|  | -- 23. List products that didn't sell (1) |
|  | SELECT PRODUCTNAME FROM PRODUCTS |
|  | LEFT JOIN ORDERDETAILS ON PRODUCTS.PRODUCTCODE = ORDERDETAILS.PRODUCTCODE |
|  | WHERE ORDERDETAILS.PRODUCTCODE IS NULL; |
|  |  |
|  | -- 24. List all customers and their sales rep even if they don’t have a sales rep (122) |
|  | SELECT CUSTOMERNAME, FIRSTNAME, LASTNAME FROM CUSTOMERS |
|  | LEFT JOIN EMPLOYEES ON CUSTOMERS.SALESREPEMPLOYEENUMBER = EMPLOYEES.EMPLOYEENUMBER; |
|  |  |
|  | -- Aggregate Functions |
|  | -- 25. Find the total of all payments made by each customer (98) |
|  | SELECT CUSTOMERNUMBER, SUM(AMOUNT) AS TOTALAMOUNT |
|  | FROM CUSTOMERS |
|  | NATURAL JOIN PAYMENTS |
|  | GROUP BY CUSTOMERNUMBER; |
|  |  |
|  | -- 26. Find the largest payment made by a customer (1) |
|  | SELECT MAX(MAXAMOUNT) FROM (SELECT CUSTOMERNUMBER, MAX(AMOUNT) AS MAXAMOUNT |
|  | FROM PAYMENTS |
|  | NATURAL JOIN CUSTOMERS |
|  | GROUP BY CUSTOMERNUMBER) AS MAXIMUM; |
|  |  |
|  | SELECT MAX(AMOUNT) AS MAXIMUM FROM PAYMENTS; |
|  |  |
|  | -- 27. Find the average payment made by a customer (1) |
|  |  |
|  | SELECT AVG(AMOUNT) AS AVERAGE FROM PAYMENTS; |
|  |  |
|  | -- 28. What is the total number of products per product line (7) |
|  |  |
|  | SELECT PRODUCTLINE, COUNT(PRODUCTNAME) |
|  | FROM PRODUCTS |
|  | NATURAL JOIN PRODUCTLINES |
|  | GROUP BY PRODUCTLINE; |
|  |  |
|  |  |
|  | -- 29. What is the number of orders per status (6) |
|  | SELECT STATUS, COUNT(ORDERNUMBER) |
|  | FROM ORDERS |
|  | GROUP BY STATUS; |
|  |  |
|  | -- 30. List all offices and the number of employees working in each office (7) |
|  | SELECT OFFICECODE, COUNT(EMPLOYEENUMBER) |
|  | FROM OFFICES |
|  | NATURAL JOIN EMPLOYEES |
|  | GROUP BY OFFICECODE; |
|  |  |
|  | -- Having |
|  | -- 31. List the total number of products per product line where number of products > 3 (6) |
|  |  |
|  | SELECT PRODUCTLINE, COUNT(PRODUCTNAME) |
|  | FROM PRODUCTLINES |
|  | NATURAL JOIN PRODUCTS |
|  | GROUP BY PRODUCTLINE |
|  | HAVING COUNT(PRODUCTNAME) > 3; |
|  |  |
|  |  |
|  | -- 32. List the product lines and vendors for product lines which are supported by < 5 vendors |
|  | -- (3) |
|  |  |
|  | SELECT PRODUCTLINE, PRODUCTVENDOR FROM PRODUCTS |
|  | WHERE PRODUCTLINE IN |
|  | (SELECT PRODUCTLINE |
|  | FROM PRODUCTLINES |
|  | NATURAL JOIN PRODUCTS |
|  | GROUP BY PRODUCTLINE |
|  | HAVING COUNT(PRODUCTVENDOR) < 5); |
|  |  |
|  | -- Computations |
|  | -- 33. What product that makes us the most money (qty\*price) (1) |
|  |  |
|  | SELECT MAX(QUANTITYINSTOCK\*BUYPRICE) FROM PRODUCTS; |
|  |  |
|  | -- 34. What is the profit per product (MSRP-buyprice) (110) |
|  | SELECT PRODUCTNAME, MSRP-BUYPRICE AS PROFIT |
|  | FROM PRODUCTS |
|  | GROUP BY PRODUCTNAME, MSRP, BUYPRICE; |
|  |  |
|  | -- Self Join |
|  | -- 25. Find all of the customers who have the same sales representative as some other |
|  | -- customer, and either customer name has ‘Australian’ in it. List each of the customers |
|  | -- sharing a sales representative, and the name of the sales representative. Order by the |
|  | -- name of the first customer, then the second. Do not show any combination more than |
|  | -- once. (18) |
|  |  |
|  | SELECT one.CUSTOMERNAME, FIRSTNAME, LASTNAME, other.CUSTOMERNAME |
|  | FROM CUSTOMERS one |
|  | INNER JOIN CUSTOMERS other |
|  | ON one.SALESREPEMPLOYEENUMBER = other.SALESREPEMPLOYEENUMBER |
|  | INNER JOIN EMPLOYEES |
|  | ON one.SALESREPEMPLOYEENUMBER = EMPLOYEES.EMPLOYEENUMBER |
|  | WHERE one.CUSTOMERNUMBER < other.CUSTOMERNUMBER |
|  | AND (one.CUSTOMERNAME LIKE '%Australian%' OR other.CUSTOMERNAME LIKE '%Australian%'); |
|  |  |
|  | -- Set Operations |
|  | -- 36. List all customers who didn't order in 2015 (78) |
|  |  |
|  | SELECT CUSTOMERNAME FROM CUSTOMERS |
|  | WHERE NOT EXISTS |
|  | (SELECT ORDERDATE FROM ORDERS |
|  | WHERE YEAR(ORDERDATE) = 2015 AND ORDERS.CUSTOMERNUMBER = CUSTOMERS.CUSTOMERNUMBER); |
|  |  |
|  | -- 37. List all people that we deal with (employees and customer contacts). Display first name, |
|  | -- last name, company name (or employee) (145) |
|  |  |
|  | SELECT FIRSTNAME AS "First Name", LASTNAME AS "Last Name", 'Employee' AS "COMPANY" FROM EMPLOYEES |
|  | UNION |
|  | SELECT CONTACTFIRSTNAME, CONTACTLASTNAME, CUSTOMERNAME FROM CUSTOMERS; |
|  |  |
|  | -- 38. List the last name, first name, and employee number of all of the employees who do not |
|  | -- have any customers. Order by last name first, then the first name. (8). |
|  |  |
|  | SELECT LASTNAME, FIRSTNAME, EMPLOYEENUMBER FROM EMPLOYEES |
|  | EXCEPT |
|  | (SELECT LASTNAME, FIRSTNAME, EMPLOYEENUMBER |
|  | FROM EMPLOYEES INNER JOIN CUSTOMERS ON EMPLOYEES.EMPLOYEENUMBER = CUSTOMERS.SALESREPEMPLOYEENUMBER) |
|  | ORDER BY LASTNAME, FIRSTNAME; |
|  |  |
|  | -- 39. List the states and the country that the state is part of that have customers but not |
|  | -- offices, offices but not customers, or both one or more customers and one or more |
|  | -- offices all in one query. Designate which state is which with the string 'Customer', |
|  | -- 'Office', or 'Both'. If a state falls into the “Both” category, do not list it as a Customer or |
|  | -- an Office state. Order by the country, then the state. Give the category column (where |
|  | -- you list ‘Customer’, ‘Office’, or ‘Both’) a header of “Category” and exclude any entries in |
|  | -- which the state is null. (19) |
|  |  |
|  | SELECT STATE, COUNTRY, "CATEGORY" FROM |
|  | (SELECT STATE, COUNTRY, 'CUSTOMER' AS "CATEGORY" FROM CUSTOMERS |
|  | WHERE STATE IS NOT NULL |
|  | EXCEPT |
|  | SELECT STATE, COUNTRY, 'CUSTOMER' AS "CATEGORY" FROM OFFICES |
|  | WHERE STATE IS NOT NULL |
|  | UNION |
|  | SELECT STATE, COUNTRY, 'OFFICE' AS "CATEGORY" FROM OFFICES |
|  | WHERE STATE IS NOT NULL |
|  | EXCEPT |
|  | SELECT STATE, COUNTRY, 'OFFICE' AS "CATEGORY" FROM CUSTOMERS |
|  | WHERE STATE IS NOT NULL |
|  | UNION |
|  | SELECT STATE, COUNTRY, 'BOTH' AS "CATEGORY" FROM CUSTOMERS |
|  | WHERE STATE IS NOT NULL |
|  | INTERSECT |
|  | SELECT STATE, COUNTRY, 'BOTH' AS "CATEGORY" FROM OFFICES |
|  | WHERE STATE IS NOT NULL) AS STATES |
|  | ORDER BY COUNTRY, STATE; |
|  |  |
|  | -- 40. List the Product Code and Product name of every product that has never been in on |
|  | -- order in which the customer asked for more than 48 of them. Order by the Product |
|  | -- Name. (7) |
|  |  |
|  | SELECT PRODUCTS.PRODUCTCODE, PRODUCTNAME FROM PRODUCTS |
|  | EXCEPT |
|  | (SELECT ORDERDETAILS.PRODUCTCODE, PRODUCTNAME FROM PRODUCTS |
|  | INNER JOIN ORDERDETAILS ON PRODUCTS.PRODUCTCODE = ORDERDETAILS.PRODUCTCODE |
|  | WHERE QUANTITYORDERED > 48) |
|  | ORDER BY PRODUCTNAME; |
|  |  |
|  |  |
|  | -- Subqueries |
|  | -- 41. List the products in the product line with the most number of products (38) |
|  | -- 42. Find the first name and last name of all customer contacts whose customer is located in |
|  | -- the same state as the San Francisco office. (11) |
|  | -- 43. What is the customer and sales person of the highest priced order? (1) |
|  |  |
|  | -- Recursion |
|  | -- 44. What is the manager who manages the greatest number of employees (2) |
|  |  |
|  | SELECT LASTNAME, FIRSTNAME FROM EMPLOYEES WHERE EMPLOYEENUMBER IN (SELECT EMPLOYEENUMBER, COUNT(REPORTSTO) FROM EMPLOYEES WHERE EMPLOYEENUMBER = EMPLOYEES.REPORTSTO GROUP BY EMPLOYEENUMBER); |
|  |  |
|  | -- 45. Select all employees who work for the manager that manages the greatest number of |
|  | -- employee (12) |
|  |  |
|  | SELECT COUNT(REPORTSTO) FROM EMPLOYEES WHERE REPORTSTO IN (SELECT ); |
|  |  |
|  | -- 46. List all employees that have the same last name. Make sure each combination is listed |
|  | -- only once (5) |
|  |  |
|  | SELECT CUSTOMERNAME, CONTACTFIRSTNAME, CONTACTLASTNAME FROM CUSTOMERS WHERE CUSTOMERNAME= (SELECT MIN(CUSTOMERNAME) FROM CUSTOMERS); |
|  |  |
|  | SELECT one.CUSTOMERNAME AS "First Customer", two.CUSTOMERNAME as "Other Customer", PAYMENTDATE FROM ( |
|  |  |
|  | (SELECT CUSTOMERNUMBER, CUSTOMERNAME, PAYMENTDATE FROM CUSTOMERS INNER JOIN PAYMENTS USING (CUSTOMERNUMBER)) one |
|  |  |
|  | INNER JOIN |
|  |  |
|  | (SELECT CUSTOMERNUMBER, CUSTOMERNAME, PAYMENTDATE FROM CUSTOMERS INNER JOIN PAYMENTS USING (CUSTOMERNUMBER)) two |
|  |  |
|  | USING (PAYMENTDATE)) |
|  |  |
|  | WHERE one.CUSTOMERNUMBER < two.CUSTOMERNUMBer; |
|  |  |
|  |  |
|  | SELECT MAX("Value") |
|  | FROM (SELECT PRODUCTCODE, sum(PRICEEACH\*QUANTITYORDERED) "Value" |
|  | FROM ORDERDETAILS |
|  | GROUP BY PRODUCTCODE) max1 |
|  | WHERE "Value" = |
|  | (SELECT MAX("Value") |
|  | FROM (SELECT PRODUCTCODE, sum(PRICEEACH\*QUANTITYORDERED) "Value" |
|  | FROM ORDERDETAILS |
|  | GROUP BY ORDERDETAILS.PRODUCTCODE) maxv); |