

Tyler Milligan
Homework 3

1. List in alphabetical order the names of the cities in Australia where ClassicModels has customers. (5)

SELECT city FROM CUSTOMERS WHERE country='Australia' order by city ASC;

	city
▶	Chatswood
	Glen Waverly
	Melbourne
	North Sydney
	South Brisbane

2. List the EmployeeNumber, LastName, FirstName, Extension for all employees working out of the Sydney office. (4)

SELECT employeeNumber, lastName, firstName, extension FROM employees WHERE officeCode = '6';

	employeeNumber	lastName	firstName	extension
▶	1088	Patterson	William	x4871
	1611	Fixter	Andy	x101
	1612	Marsh	Peter	x102
	1619	King	Tom	x103
*	NULL	NULL	NULL	NULL

3. List the ProductCode, ProductName, ProductVendor, QuantityInStock and ProductLine for all products with a QuantityInStock greater than 4000 and less than 5000. (8)

SELECT productName, productVendor, productCode, quantityInStock, productLine FROM products WHERE quantityInStock BETWEEN 4000 AND 5000

	productName	productVendor	productCode	quantityInStock	productLine
▶	1998 Chrysler Plymouth Prowler	Gearbox Collectibles	S18_2238	4724	Classic Cars
	1936 Harley Davidson El Knucklehead	Welly Diecast Productions	S18_2625	4357	Motorcycles
	1913 Ford Model T Speedster	Carousel DieCast Legends	S18_2949	4189	Vintage Cars
	1999 Yamaha Speed Boat	Min Lin Diecast	S18_3029	4259	Ships
	1970 Dodge Coronet	Highway 66 Mini Classics	S24_1444	4074	Classic Cars
	1969 Chevrolet Camaro Z28	Exoto Designs	S24_3191	4695	Classic Cars
	1936 Chrysler Airflow	Second Gear Diecast	S24_4258	4710	Vintage Cars
	Boeing X-32A JSF	Motor City Art Classics	S72_1253	4857	Planes
*	NULL	NULL	NULL	NULL	NULL

SELECT productCode, productName, productVendor, quantityInStock, productLine FROM products WHERE quantityInStock > 4000 and quantityInStock < 5000;

	productCode	productName	productVendor	quantityInStock	productLine
▶	S18_2238	1998 Chrysler Plymouth Prowler	Gearbox Collectibles	4724	Classic Cars
	S18_2625	1936 Harley Davidson El Knucklehead	Welly Diecast Productions	4357	Motorcycles
	S18_2949	1913 Ford Model T Speedster	Carousel DieCast Legends	4189	Vintage Cars
	S18_3029	1999 Yamaha Speed Boat	Min Lin Diecast	4259	Ships
	S24_1444	1970 Dodge Coronet	Highway 66 Mini Classics	4074	Classic Cars
	S24_3191	1969 Chevrolet Camaro Z28	Exoto Designs	4695	Classic Cars
	S24_4258	1936 Chrysler Airflow	Second Gear Diecast	4710	Vintage Cars
	S72_1253	Boeing X-32A JSF	Motor City Art Classics	4857	Planes
*	NULL	NULL	NULL	NULL	NULL

4. (Use a SUBQUERY) List the ProductCode, ProductName, ProductVendor, BuyPrice and MSRP for the most expensive (highest MSRP) product sold by ClassicModels. ("MSRP" is the Manufacturer's Suggested Retail Price.) (1)

SELECT productCode, productName, productVendor, buyPrice FROM products WHERE MSRP = (SELECT MAX(MSRP) FROM products);

	productCode	productName	productVendor	buyPrice
▶	S10_1949	1952 Alpine Renault 1300	Classic Metal Creations	98.58
*	NULL	NULL	NULL	NULL

5. List the ProductName, MSRP, BuyPrice, and Margin of the product that has the lowest Margin (Margin = MSRP minus BuyPrice). (1)

SELECT productName, MSRP, buyPrice, (MSRP - buyPrice) FROM products WHERE (MSRP - buyPrice) = (SELECT MIN(MSRP - buyPrice) FROM products);

	productName	MSRP	buyPrice	(MSRP - buyPrice)
▶	1939 Chevrolet Deluxe Coupe	33.19	22.57	10.62

6. List the country and the number of customers from that country for all countries with more than five customers. List the countries in order from most to least customers. Title the column heading for the count of customers as "Customers". (4)

SELECT customerName, country, COUNT(*) AS Customers FROM customers GROUP BY country HAVING COUNT(*) > 4;

	customerName	country	Customers
▶	Atelier graphique	France	12
	Signal Gift Stores	USA	36
	Australian Collectors, Co.	Australia	5
	Blauer See Auto, Co.	Germany	13
	Euro + Shopping Channel	Spain	7
	AV Stores, Co.	UK	5

7. List the ProductCode, ProductName, and number of orders for the product with the most orders. Title the column heading for the count of orders as “OrderCount”. (1)

```
SELECT orderdetails.productCode, SUM(orderdetails.quantityOrdered) as QuantityOrdered from
orderdetails GROUP BY productCode ORDER BY QuantityOrdered DESC;
```

	productCode	QuantityOrdered
▶	S18_3232	1808
	S18_1342	1111
	S700_4002	1085
	S18_3856	1076
	S50_1341	1074
	S18_4600	1061
	S10_1678	1057
	S12_4473	1056
	S18_2319	1053
	S24_3856	1052
	S24_3949	1051
	S700_3167	1047
	S18_1662	1040
	S18_2949	1038

8. Create an “Employee Roster” report listing the Manager’s name and the name of the employees who report to that manager. Sort by Manager. For both managers and employees, concatenate Firstname + Lastname and refer to them as “ManagerName” and “EmployeeName” in the answer set. (22)

9. List the EmployeeNumber, LastName, FirstName of the president of the company (the one employee with no boss.) (1)

```
SELECT lastName, firstName, employeeNumber FROM employees WHERE jobtitle = 'President';
```

	lastName	firstName	employeeNumber
▶	Murphy	Diane	1002
✱	NULL	NULL	NULL

10. List the ProductName for all products in the “Classic Cars” product line from the 1960’s. (11)

```
SELECT productName FROM products WHERE productLine = "Classic Cars" AND productName LIKE
'196_';
```

	productName
▶	1962 LanciaA Delta 16V
	1968 Ford Mustang
	1969 Corvair Monza
	1968 Dodge Charger
	1969 Ford Falcon
	1969 Dodge Charger
	1965 Aston Martin DB5
	1969 Dodge Super Bee
	1966 Shelby Cobra 427 S/C
	1969 Chevrolet Camaro Z28
	1961 Chevrolet Impala

11. List the month name and the year and the order count for the TOP TWO months in which ClassicModels customers placed the highest number of orders. (2)

```
SELECT YEAR(orderDate) AS year, MONTH(orderDate) AS month , count(*) AS count FROM orders
GROUP BY month ORDER BY count DESC LIMIT 2;
```

	year	month	count
▶	2003	11	63
	2003	10	31

12. List the firstname, lastname of employees who are Sales Reps who have no assigned customers. (2)

```
SELECT employees.firstName, employees.lastName FROM employees LEFT OUTER JOIN customers ON
employees.employeeNumber = customers.salesRepEmployeeNumber WHERE
customers.customerNumber is NULL AND employees.jobTitle = "Sales Rep";
```

	firstName	lastName
▶	Tom	King
	Yoshimi	Kato

13. List the customername of customers from Switzerland with no orders. (2)

```
SELECT customers.customerName FROM customers LEFT OUTER JOIN orders ON
customers.customerNumber = orders.customerNumber WHERE customers.country = "Switzerland" AND
orders.orderNumber IS NULL;
```

	customerName
▶	BG&E Collectables
	Precious Collectables

14. List the customername and total quantity of products ordered for customers who have ordered less than 500 products across all their orders. (11)

```
SELECT customers.customerName, SUM(orderdetails.quantityOrdered) sumOrders FROM customers
LEFT OUTER JOIN orders ON customers.customerNumber = orders.customerNumber LEFT OUTER JOIN
orderdetails ON orderdetails.orderNumber = orders.orderNumber GROUP BY orders.customerNumber
HAVING sumOrders < 500 ORDER BY sumOrders DESC;
```

	customerName	sumOrders
▶	Clover Collections, Co.	490
	CAF Imports	468
	Bavarian Collectables Imports, Co.	401
	Microscale Inc.	381
	Cambridge Collectables Co.	357
	Double Decker Gift Stores, Ltd	357
	Auto-Moto Classics Inc.	287
	Royale Belge	278
	Frau da Collezione	272
	Atelier graphique	270
	Boards & Toys Co.	102

15. Create a NEW table named “LowCustomers” with three columns: CustomerNumber (integer), ContactDate (DATE) and OrderTotal (a decimal number with 9 digits in total having two decimal places). None of these columns can be NULL. Include a PRIMARY KEY constraint named “LowCustomer_PK” on CustomerNumber. (no answer set)

```
CREATE TABLE IF NOT EXISTS LowCustomers(
  CustomerNumber int NOT NULL,
  ContactDate date NOT NULL,
  OrderTotal Decimal(9,2) NOT NULL,
  CONSTRAINT LowCustomer_PK PRIMARY KEY (CustomerNumber)
);
```

16. Populate the new table “LowCustomers” with the CustomerNumber, today’s date, and the total value of all their orders (PriceEach * quantityOrdered) for those customers whose order total value is less than \$50,000. (inserted 16 rows, no answer set)

```
INSERT INTO LowCustomers
SELECT customers.customerNumber, Current_DATE(), SUM(orderdetails.priceEach *
orderdetails.quantityOrdered)
FROM customers, orderdetails, orders
```

```
WHERE customers.customerNumber AND orderdetails.orderNumber = orders.orderNumber  
GROUP BY customers.customerNumber  
HAVING SUM(orderdetails.priceEach * orderdetails.quantityOrdered) < 50000;
```

17. List the contents of the LowCustomers table in descending OrderTotal sequence. (16)

```
SELECT OrderTotal FROM LowCustomers ORDER BY OrderTotal DESC;
```

18. Add a new column to the LowCustomers table called OrderCount (integer). (No answer set)

```
ALTER TABLE LowCustomers ADD OrderCount INT;
```

19. Update the LowCustomers table, setting the OrderCount column to a random number (from 0 to 18). (Should update 16 rows) HINT: use the RAND() and FLOOR() functions.

```
UPDATE LowCustomers SET OrderCount = RAND() * 18;
```

20. List the contents of the LowCustomers table in descending OrderCount sequence. (16)

```
SELECT OrderCount FROM LowCustomers ORDER BY OrderCount Desc;
```

21. Drop the LowCustomers table. (no answer set)

```
DROP TABLE LowCustomers;
```