

# Tugas 4 Praktikum Basis Data

Baskara (16/398499/PA/17460)

03-11-2017

## Single Entity

1. List products sold by order date.

```
SELECT o.orderDate, p.productName
FROM products p
JOIN orderdetails d USING(productCode)
JOIN orders o USING(orderNumber)
ORDER BY o.orderDate DESC;
```

RESULT:

| orderDate  | productName                         |
|------------|-------------------------------------|
| 2005-05-31 | 1962 LanciaA Delta 16V              |
| 2005-05-31 | 1957 Chevy Pickup                   |
| 2005-05-31 | 1998 Chrysler Plymouth Prowler      |
| 2005-05-31 | 1964 Mercedes Tour Bus              |
| 2005-05-31 | 1926 Ford Fire Engine               |
| .          | .                                   |
| .          | .                                   |
| 2003-01-06 | 1911 Ford Town Car                  |
| 2003-01-06 | 1932 Alfa Romeo 8C2300 Spider Sport |
| 2003-01-06 | 1936 Mercedes Benz 500k Roadster    |

2996 rows in set (0.36 sec)

2. List the order dates in descending order for orders for the 1940 Ford Pickup Truck.

```
SELECT o.orderDate
FROM orders o
JOIN orderdetails d USING(orderNumber)
JOIN products p USING(productCode)
WHERE p.productName = "1940 Ford Pickup Truck"
ORDER BY o.orderDate DESC;
```

RESULT:

| orderDate  |
|------------|
| 2005-05-31 |
| 2005-05-01 |
| 2005-03-09 |
| 2005-02-17 |

```

| 2005-01-20 |
.
.
.
| 2003-05-28 |
| 2003-03-26 |
| 2003-01-29 |
+-----+
28 rows in set (0.00 sec)

```

3. List the names of customers and their corresponding order number where a particular order from that customer has a value greater than \$25,000?

```

SELECT c.customerName, o.orderNumber, d.quantityOrdered*d.priceEach AS total_value
FROM customers c
JOIN orders o USING(customerNumber)
JOIN orderdetails d USING(orderNumber)
WHERE d.quantityOrdered*d.priceEach > 25000
ORDER BY d.quantityOrdered*d.priceEach DESC;

```

```

RESULT:
Empty set (0.16 sec)

```

4. Are there any products that appear on all orders?

```

SELECT p.productName, d.orderNumber
FROM orders o
LEFT JOIN orderdetails d USING(orderNumber)
LEFT JOIN products p USING(productCode)
HAVING COUNT(d.orderNumber) = (SELECT DISTINCT COUNT(o.orderNumber) FROM orders o);

```

```

RESULT:
Empty set (0.01 sec)

```

5. List the names of products sold at less than 80% of the MSRP.

```

SELECT p.productName, d.priceEach, p.MSRP
FROM products p
JOIN orderdetails d USING(productCode)
WHERE (d.priceEach/p.MSRP)<0.8;

```

```

RESULT:

```

| productName                 | priceEach | MSRP   |
|-----------------------------|-----------|--------|
| 1996 Moto Guzzi 1100i       | 95.15     | 118.94 |
| 1993 Mazda RX-7             | 113.23    | 141.54 |
| 1937 Lincoln Berline        | 82.19     | 102.74 |
| 1965 Aston Martin DB5       | 99.55     | 124.44 |
| 1965 Aston Martin DB5       | 99.55     | 124.44 |
| .                           |           |        |
| .                           |           |        |
| .                           |           |        |
| American Airlines: B767-300 | 73.07     | 91.34  |
| The USS Constitution Ship   | 57.82     | 72.28  |

|                           |       |       |
|---------------------------|-------|-------|
| American Airlines: MD-11S | 59.22 | 74.03 |
|---------------------------|-------|-------|

44 rows in set (0.01 sec)

6. Reports those products that have been sold with a markup of 100% or more (i.e., the priceEach is at least twice the buyPrice)

```
SELECT p.productName, d.priceEach, p.buyPrice
FROM products p
JOIN orderdetails d USING(productCode)
WHERE (d.priceEach/p.buyPrice)>=2;
```

RESULT:

| productName               | priceEach | buyPrice |
|---------------------------|-----------|----------|
| 1952 Alpine Renault 1300  | 214.30    | 98.58    |
| 1952 Alpine Renault 1300  | 197.16    | 98.58    |
| 1952 Alpine Renault 1300  | 205.73    | 98.58    |
| 1952 Alpine Renault 1300  | 212.16    | 98.58    |
| 1952 Alpine Renault 1300  | 207.87    | 98.58    |
| .                         |           |          |
| .                         |           |          |
| .                         |           |          |
| 1982 Camaro Z28           | 101.15    | 46.53    |
| American Airlines: MD-11S | 74.03     | 36.27    |
| American Airlines: MD-11S | 73.29     | 36.27    |

7. List the products ordered on a Monday.

```
SELECT p.productName, DAYNAME(o.orderDate) AS DAY
FROM products p
JOIN orderdetails d USING(productCode)
JOIN orders o USING(orderNumber)
WHERE DAYNAME(o.orderDate) = 'Monday';
```

RESULT:

| productName                           | DAY    |
|---------------------------------------|--------|
| 1969 Harley Davidson Ultimate Chopper | Monday |
| 1969 Harley Davidson Ultimate Chopper | Monday |
| 1969 Harley Davidson Ultimate Chopper | Monday |
| 1969 Harley Davidson Ultimate Chopper | Monday |
| 1969 Harley Davidson Ultimate Chopper | Monday |
| .                                     |        |
| .                                     |        |
| .                                     |        |
| Boeing X-32A JSF                      | Monday |
| Pont Yacht                            | Monday |
| Pont Yacht                            | Monday |

407 rows in set (0.00 sec)

8. What is the quantity on hand for products listed on 'On Hold' orders?

```

SELECT SUM(d.quantityOrdered) AS Quantity
FROM orders o
JOIN orderdetails d
WHERE o.status='On Hold';

```

RESULT:

```

+-----+
| Quantity |
+-----+
|   527580 |
+-----+
1 row in set (0.00 sec)

```

## Procedures

Create a procedure for purchasing an item. The procedure will be called PURCHASE and this procedure will have two parameters:

- Product Code
- Quantity

This procedure will do the following task:

1. Check if the following product exist
2. Check if the product have enough stock (based on the requested quantity)
3. Write the order and the details
4. Write the payments
5. Reduce the stock in product

```
DROP PROCEDURE IF EXISTS Purchase;
```

```
DELIMITER //
```

```
CREATE PROCEDURE Purchase (IN PCODE VARCHAR(10), IN QTY INT)
```

```
BEGIN
```

```
IF EXISTS (SELECT p.productCode FROM products p WHERE productCode = PCODE) THEN
```

```
IF (SELECT p.quantityInStock FROM products p WHERE productCode = PCODE) >= QTY THEN
```

```
INSERT INTO orders (orderDate, requiredDate, shippedDate, status, comments, customerNumber)
```

```
VALUES (CURDATE(), CURDATE(), CURDATE(), 'On Hold', 'Order Baru Gan', 103);
```

```
INSERT INTO orderdetails (orderNumber, productCode, quantityOrdered, priceEach, orderLineNumber)
```

```
VALUES (LAST_INSERT_ID(), PCODE, QTY, (SELECT buyPrice FROM products WHERE productCode = PCODE),
```

```
UPDATE products SET quantityInStock = quantityInStock - QTY WHERE productCode = PCODE;
```

```
ELSE
```

```
SELECT "STOCK TIDAK CUKUP" AS MESSAGE;
```

```
END IF;
```

```
ELSE
```

```
SELECT "PRODUCT NOT EXIST" AS MESSAGE;;
```

```
END IF;
```

```
END//
```

```
DELIMITER ;
```

```
CALL Purchase("S10_1949", 10);
```