

Stored Procedure

1. Find all the details of suppliers

DELIMITER//

CREATE PROCEDURE select_supplier()

BEGIN

SELECT*FROM supplier;

END//

DELIMITER ;

CALL select_supplier

1 > >> | ☐ Show all | Number of rows: 25 Filter rows: Search this table

+ Options

Id	CompanyName	ContactName	ContactTitle	City	Country	Phone	Fax
1	Exotic Liquids	Charlotte Cooper	NULL	London	UK	(171) 555-2222	NULL
2	New Orleans Cajun Delights	Shelley Burke	NULL	New Orleans	USA	(100) 555-4822	NULL
3	Grandma Kelly's Homestead	Regina Murphy	NULL	Ann Arbor	USA	(313) 555-5735	(313) 555-3349
4	Tokyo Traders	Yoshi Nagase	NULL	Tokyo	Japan	(03) 3555-5011	NULL
5	Cooperativa de Quesos 'Las Cabras'	Antonio del Valle Saavedra	NULL	Oviedo	Spain	(98) 598 76 54	NULL
6	Mayumi's	Mayumi Ohno	NULL	Osaka	Japan	(06) 431-7877	NULL
7	Pavlova, Ltd.	Ian Devling	NULL	Melbourne	Australia	(03) 444-2343	(03) 444-6588
8	Specialty Biscuits, Ltd.	Peter Wilson	NULL	Manchester	UK	(161) 555-4448	NULL
9	PB Knäckebröd AB	Lars Peterson	NULL	Göteborg	Sweden	031-987 65 43	031-987 65 91
10	Refrescos Americanas LTDA	Carlos Diaz	NULL	Sao Paulo	Brazil	(11) 555 4640	NULL
11	Heli Süßwaren GmbH & Co. KG	Petra Winkler	NULL	Berlin	Germany	(010) 9984510	NULL
12	Plutzer Lebensmittelgroßmärkte AG	Martin Bein	NULL	Frankfurt	Germany	(069) 992755	NULL
13	Nord-Ost-Fisch Handelsgesellschaft mbH	Sven Petersen	NULL	Cuxhaven	Germany	(04721) 8713	(04721) 8714
14	Formaggi Fortini s.r.l.	Elio Rossi	NULL	Ravenna	Italy	(0544) 60323	(0544) 60603
15	Norske Meierier	Beate Vileid	NULL	Sandvika	Norway	(0)2-953010	NULL
16	Bigfoot Breweries	Cheryl Saylor	NULL	Bend	USA	(503) 555-9931	NULL
17	Svensk Sjöföda AB	Michael Björn	NULL	Stockholm	Sweden	08-123 45 67	NULL
18	Aux joyeux ecclésiastiques	Guyène Nodier	NULL	Paris	France	(1) 03.83.00.68	(1) 03.83.00.62
19	New England Seafood Cannery	Robb Merchant	NULL	Boston	USA	(617) 555-3267	(617) 555-3389
	Console	Chandra Leka	NULL	Singapore	Singapore	555-8787	NULL

2. Find all the details of suppliers who are from "UK"?

```

DELIMITER //
CREATE PROCEDURE supplier_by_country (IN preferred_country VARCHAR (40))
BEGIN
SELECT*FROM supplier where Country=preferred_country;
END//
DELIMITER ;

```

✓ Showing rows 0 - 1 (2 total, Query took 0.1095 seconds.)

CALL supplier_by_country('UK')

☐ Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

Id	CompanyName	ContactName	ContactTitle	City	Country	Phone	Fax
1	Exotic Liquids	Charlotte Cooper	NULL	London	UK	(171) 555-2222	NULL
8	Specialty Biscuits, Ltd.	Peter Wilson	NULL	Manchester	UK	(161) 555-4448	NULL

3. Find the phone number of the company name “Tokyo Traders”

```

DELIMITER //
CREATE PROCEDURE supplier_phoneNo
(IN search_supplier VARCHAR (40),
OUT output_phoneNo VARCHAR (30))
BEGIN
SELECT phone INTO output_phoneNo FROM supplier where CompanyName=search_supplier;
END//
DELIMITER ;

```

Executing the Procedure

```

CALL supplier_phoneNo('Tokyo Traders',@output_phoneNo);
SELECT @output_phoneNo as Phone;

```

✓ MySQL returned an empty result set (i.e. zero rows).

CALL supplier_phoneNo('Tokyo Traders',@output_phoneNo)

✓ Showing rows 0 - 0 (1 total, Query took 0.0005 seconds.)

SELECT @output_phoneNo as Phone

☐ Show all | Number of rows: 25 | Filter

+ Options

Phone
(03) 3555-5011

4. Create a Stored Procedure 'UnitPriceCategory' where the unit price of product is more than 100 as 'High', more than 50 as 'Medium' and less than 50 as 'Low'.

DELIMITER //

CREATE PROCEDURE UnitPriceCategory
(IN search_product VARCHAR (40),
OUT proRate VARCHAR (30))

BEGIN

DECLARE product_price INT;

SELECT UnitPrice INTO product_price FROM product WHERE ProductName=search_product;

IF product_price > 100 THEN

SET proRate = 'High';

ELSEIF (product_price <= 100 AND product_price >= 50) THEN

SET proRate = 'Medium';

ELSEIF (product_price < 50) THEN

SET proRate = 'Low';

END IF;

END//

DELIMITER ;

Procedure Calling:

```
CALL UnitPriceCategory ('Mishi Kobe Niku',@proRate);  
SELECT @proRate
```

MySQL returned an empty result set (i.e. zero rows). (Query OK, 0 rows affected)

```
CALL UnitPriceCategory ('Mishi Kobe Niku',@proRate)
```

Showing rows 0 - 0 (1 total, Query took 0.0005 seconds.)

```
SELECT @proRate
```

☐ Show all | Number of rows: 1

+ Options

@proRate

Medium

Showing rows 0 - 0 (1 total, Query took 0.0283 seconds.)

```
SELECT UnitPrice FROM product where ProductName='Mishi Kobe Niku'
```

☐ Show all | Number of rows: 25 | Filter rows:

+ Options

UnitPrice
97.00

☐ Edit ☐ Copy ☐ Delete

5) Product supplying companies in various countries are supplying different products. Assume that you need to identify which countries are leading the product supply chain. Based on the count of different products they are supplying; you need to apply below rating criteria.

```
CREATE PROCEDURE SupplyCountry
(IN sup_country VARCHAR (40),
OUT no_product VARCHAR (40),
OUT lendstatus VARCHAR (30))

BEGIN
SELECT
COUNT(product.SupplierId) into no_product
FROM
product INNER JOIN supplier ON
product.SupplierId = supplier.Id
WHERE
supplier.Country=sup_country LIMIT 10;

IF no_product > 9 THEN
SET lendstatus = 'Top Leading supplier country';
ELSEIF (no_product <= 9 AND no_product >= 5) THEN
SET lendstatus = 'Middle Leading supplier country';
ELSEIF (no_product < 5) THEN
SET lendstatus = 'Not a Leading supplier country';
END IF;
END//
DELIMITER ;
```

Procedure Call

The screenshot displays a MySQL query execution window. At the top, a green status bar indicates "MySQL returned an empty result set (i.e. zero rows). (Query to)". Below this, the executed query is shown: `CALL SupplyCountry ('UK',@no_product,@lendstatus)`. A yellow warning box follows, stating "Warning: #1265 Data truncated for column 'lendstatus' at row 3". Another green status bar shows "Showing rows 0 - 0 (1 total, Query took 0.0008 seconds.)". The query `SELECT @no_product AS no_of_products,@lendstatus as lendingStatus` is displayed below. At the bottom, there are controls for "Show all", "Number of rows" (set to 25), and "Filter rows:". A table with two columns, "no_of_products" and "lendingStatus", is shown with one row: "7" and "Middle Leading supplier countr".

no_of_products	lendingStatus
7	Middle Leading supplier countr

6) Create the Stored Procedure '*OrderItemCount*' to get the count of items in a given customer's purchase order.

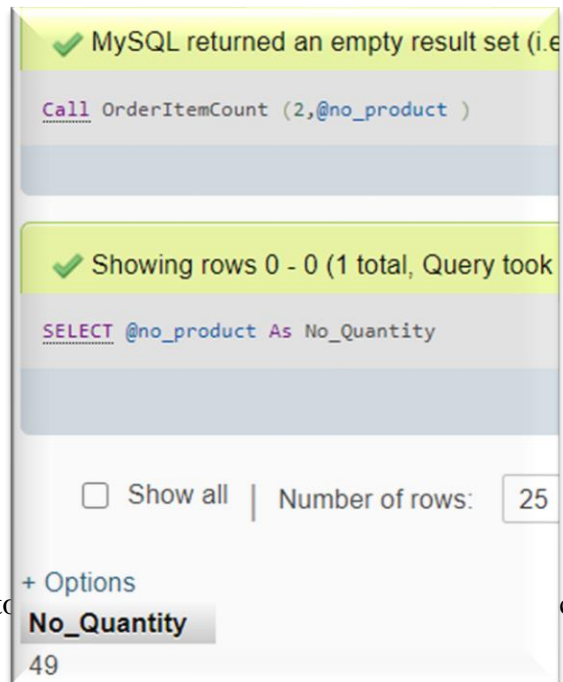
```
DELIMITER //
CREATE PROCEDURE OrderItemCount
(IN cus_order VARCHAR (40),
OUT no_product VARCHAR (40))

BEGIN
SELECT
SUM(orderitem.Quantity) into no_product
FROM
orderitem INNER JOIN purchase_order ON
purchase_order.Id = orderitem.OrderId
WHERE
purchase_order.Id=cus_order;

END//
DELIMITER ;
```

When Procedure Called

Call OrderItemCount (2,@no_product); (2 Is Customer OrderNumber)
SELECT @no_product As No_Quantity;



7) What are the advantages of having stored procedures?

1. **Better Performance –**

The procedure calls are quick and efficient as stored procedures are compiled once and stored in executable form. Hence the response is quick. The executable code is automatically cached, hence lowers the memory requirements.

2. **Ease of Use –**

To create a stored procedure, one can use any Java Integrated Development Environment (IDE). Then, they can be deployed on any tier of network architecture.

3. **Scalability –**

Stored procedures increase scalability by isolating application processing on the server.

4. **Maintainability –**

Maintaining a procedure on a server is much easier than maintaining copies on various client machines, this is because scripts are in one location.