



Instituto Tecnológico Superior de Jerez

Jerez, Zacatecas

10 de diciembre de 2020

Felipe de Jesús Márquez Olague

flype360360@gmail.com

S18070182

Ingeniería en Sistemas Computacionales

Quinto Semestre

Taller de Bases de Datos

Capturas de procedimientos almacenados

ISC Salvador Acevedo Sandoval

```
mysql> DELIMITER $$
mysql>
mysql> CREATE PROCEDURE GetCustomers()
  -> BEGIN
  -> SELECT
  -> customerName,
  -> city,
  -> state,
  -> postalCode,
  -> country
  -> FROM
  -> customers
  -> ORDER BY customerName;
  -> END$$
Query OK, 0 rows affected (0.09 sec)
```

```
mysql> DELIMITER ;
mysql> CALL GetCustomers();
```

customerName	city	state	postalCode	country
Alpha Cognac	Toulouse	NULL	31000	France
American Souvenirs Inc	New Haven	CT	97823	USA
Amica Models & Co.	Torino	NULL	10100	Italy
ANG Resellers	Madrid	NULL	28001	Spain
Anna's Decorations, Ltd	North Sydney	NSW	2060	Australia
Anton Designs, Ltd.	Madrid	NULL	28023	Spain
Asian Shopping Network, Co	Singapore	NULL	038988	Singapore
Asian Treasures, Inc.	Cork	Co. Cork	NULL	Ireland
Atelier graphique	Nantes	NULL	44000	France
Australian Collectables, Ltd	Glen Waverly	Victoria	3150	Australia
Australian Collectors, Co.	Melbourne	Victoria	3004	Australia
Australian Gift Network, Co	South Brisbane	Queensland	4101	Australia
Auto Associés & Cie.	Versailles	NULL	78000	France
Auto Canal+ Petit	Paris	NULL	75016	France
Auto-Moto Classics Inc.	Brickhaven	MA	58339	USA
AV Stores, Co.	Manchester	NULL	EC2 5NT	UK
Baane Mini Imports	Stavern	NULL	4110	Norway
Bavarian Collectables Imports, Co.	Munich	NULL	80686	Germany
BG&E Collectables	Fribourg	NULL	1700	Switzerland
Blauer See Auto, Co.	Frankfurt	NULL	60528	Germany
Boards & Toys Co.	Glendale	CA	92561	USA
CAF Imports	Madrid	NULL	28023	Spain
Cambridge Collectables Co.	Cambridge	MA	51247	USA
Canadian Gift Exchange Network	Vancouver	BC	V3F 2K1	Canada
Classic Gift Ideas, Inc	Philadelphia	PA	71270	USA
Classic Legends Inc.	NYC	NY	10022	USA
Clover Collections, Co.	Dublin	NULL	2	Ireland
Collectable Mini Designs Co.	San Diego	CA	91217	USA
Collectables For Less Inc.	Brickhaven	MA	58339	USA
Corporate Gift Ideas Co.	San Francisco	CA	94217	USA
Corrida Auto Replicas, Ltd	Madrid	NULL	28023	Spain
Cramer Spezialitäten, Ltd	Brandenburg	NULL	14776	Germany
Cruz & Sons Co.	Makati City	NULL	1227 MM	Philippines
Daedalus Designs Imports	Lille	NULL	59000	France
Danish Wholesale Imports	Kobenhavn	NULL	1734	Denmark

MySQL CREATE PROCEDURE

```
mysql> DELIMITER //
mysql>
mysql> CREATE PROCEDURE GetAllProducts()
  -> BEGIN
  -> SELECT * FROM products;
  -> END //
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> DELIMITER ;
```

```
mysql> CALL GetAllProducts();
+-----+-----+-----+-----+-----+-----+-----+-----+
| productCode | productName | productLine | productScale | productVendor | productDescription | quantityInStock | buyPrice | MSRP |
+-----+-----+-----+-----+-----+-----+-----+-----+
| S10_1678 | 1969 Harley Davidson Ultimate Chopper | Motorcycles | 1:10 | Min Lin Diecast | This replica features working kickstand, front suspension, gear-shift lever, footbrake lever, drive chain, wheels and steering. All parts are particularly delicate due to their precise scale and require special care and attention. | 7933 | 48.81 | 95.70 |
| S10_1949 | 1952 Alpine Renault 1300 | Classic Cars | 1:10 | Classic Metal Creations | Turnable front wheels; steering function; detailed interior; detailed engine; opening hood; opening trunk; opening doors; and detailed chassis. | 7305 | 98.58 | 214.30 |
| S10_2016 | 1996 Moto Guzzi 1100i | Motorcycles | 1:10 | Highway 66 Mini Classics | Official Moto Guzzi logos and insignias, saddle bags located on side of motorcycle, detailed engine, working steering, working suspension, two leather seats, luggage rack, dual exhaust pipes, small saddle bag located on handle bars, two-tone paint with chrome accents, superior die-cast detail, rotating wheels, working kick stand, diecast metal with plastic parts and baked enamel finish. | 6625 | 68.99 | 118.94 |
| S10_4698 | 2003 Harley-Davidson Eagle Drag Bike | Motorcycles | 1:10 | Red Start Diecast | Model features, official Harley Davidson logos and insignias, detachable rear wheelie bar, heavy diecast metal with resin parts, authentic multi-color tampon-printed graphics, separate engine drive belts, free-turning front fork, rotating tires and rear racing slick, certificate of authenticity, detailed engine, display stand, precision diecast replica, baked enamel finish, 1:10 scale model, removable fender, seat and tank cover piece for displaying the superior detail of the v-twin engine | 3252 | 85.68 | 136.00 |
| S10_4757 | 1972 Alfa Romeo GTA | Classic Cars | 1:10 | Motor City Art Classics | Features include: Turnable front wheels; steering function; detailed interior; detailed engine; opening hood; opening trunk; opening doors; and detailed chassis. | 6791 | 103.42 | 147.74 |
| S10_4962 | 1962 Lancia Delta 16V | Classic Cars | 1:10 | Second Gear Diecast | Features include: Turnable front wheels; steering function; detailed interior; detailed engine; opening hood; opening trunk; opening doors; and detailed chassis. | 60 | 95.34 | 104.57 |
| S12_1099 | 1968 Ford Mustang | Classic Cars | 1:12 | Autoart Studio Design | Hood, doors and trunk all open to reveal highly detailed interior features. Steering wheel actually turns the front wheels. Color dark green. | 60 | 95.34 | 104.57 |
```

MySQL Delimiter

```
mysql> DELIMITER $$
mysql>
mysql> CREATE PROCEDURE sp_name()
  -> BEGIN
  -> -- statements
  -> END $$
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> DELIMITER ;
```

MySQL DROP PROCEDURE

```
mysql> DELIMITER ;
mysql> DELIMITER $$
mysql>
mysql> CREATE PROCEDURE GetEmployees()
  -> BEGIN
  ->     SELECT
  ->         firstName,
  ->         lastName,
  ->         city,
  ->         state,
  ->         country
  ->     FROM employees
  ->     INNER JOIN offices using (officeCode);
  ->
  -> END$$
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> DELIMITER ;
```

```
mysql>
mysql> DELIMITER ;
mysql> DELIMITER $$
mysql>
mysql> CREATE PROCEDURE GetEmployees()
  -> BEGIN
  ->     SELECT
  ->         firstName,
  ->         lastName,
  ->         city,
  ->         state,
  ->         country
  ->     FROM employees
  ->     INNER JOIN offices using (officeCode);
  ->
  -> END$$
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> DELIMITER ;
mysql> DROP PROCEDURE GetEmployees;
Query OK, 0 rows affected (0.04 sec)
```

MySQL Stored Procedure Parameters

```
mysql> DELIMITER ;
mysql> DROP PROCEDURE GetEmployees;
Query OK, 0 rows affected (0.04 sec)
```

```
mysql> DELIMITER //
mysql> CREATE PROCEDURE GetOfficeByCountry(
-> IN countryName VARCHAR(255)
-> )
-> BEGIN
-> SELECT *
-> FROM offices
-> WHERE country = countryName;
-> END //
Query OK, 0 rows affected (0.02 sec)
```

```
mysql>
mysql> DELIMITER ;
mysql> CALL GetOfficeByCountry('USA');
```

officeCode	city	phone	addressLine1	addressLine2	state	country	postalCode	territory
1	San Francisco	+1 650 219 4782	100 Market Street	Suite 300	CA	USA	94080	NA
2	Boston	+1 215 837 0825	1550 Court Place	Suite 102	MA	USA	02107	NA
3	NYC	+1 212 555 3000	523 East 53rd Street	apt. 5A	NY	USA	10022	NA

```
3 rows in set (0.52 sec)
```

```
Query OK, 0 rows affected (0.54 sec)
```

```
mysql> CREATE PROCEDURE GetOrderCountByStatus (
-> IN orderStatus VARCHAR(25),
-> OUT total INT
-> )
-> BEGIN
-> SELECT COUNT(orderNumber)
-> INTO total
-> FROM orders
-> WHERE status = orderStatus;
-> END$$
```

```
Query OK, 0 rows affected (0.59 sec)
```

```
mysql>
mysql> DELIMITER ;
mysql> CALL GetOrderCountByStatus('Shipped',@total);
Query OK, 1 row affected (0.02 sec)
```

```
mysql> SELECT @total;
```

```
+-----+
| @total |
+-----+
|      303 |
+-----+
```

```
1 row in set (0.00 sec)
```

```
mysql> CALL GetOrderCountByStatus('in process',@total);
Query OK, 1 row affected (0.00 sec)
```

```
mysql> SELECT @total AS total_in_process;
```

```
+-----+
| total_in_process |
+-----+
|                9 |
+-----+
```

```
1 row in set (0.00 sec)
```

```

mysql> SET @counter = 1;
Query OK, 0 rows affected (0.00 sec)

mysql> CALL SetCounter(@counter,1); -- 2
Query OK, 0 rows affected (0.00 sec)

mysql> CALL SetCounter(@counter,1); -- 3
Query OK, 0 rows affected (0.00 sec)

mysql> CALL SetCounter(@counter,5); -- 8
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT @counter; -- 8
+-----+
| @counter |
+-----+
|          8 |
+-----+
1 row in set (0.00 sec)

```

Alter Stored Procedures

```

mysql> DELIMITER $$
mysql>
mysql> CREATE PROCEDURE GetOrderAmount()
-> BEGIN
->     SELECT
->         SUM(quantityOrdered * priceEach)
->     FROM orderDetails;
-> END$$
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> DELIMITER ;

```

GetOrderAmount - Routine

Name: The name of the routine is parsed automatically from the DDL statement. The DDL is parsed automatically while you type.

DDL:

```
1 • CREATE DEFINER=`root`@`localhost` PROCEDURE `GetOrderAmount`(  
2   BEGIN  
3     SELECT  
4       SUM(quantityOrdered * priceEach)  
5     FROM orderDetails;  
6   END
```

Name: The name of the routine is parsed automatically from the DDL statement. The DDL is parsed automatically while you type.

DDL:

```
1 • CREATE DEFINER=`root`@`localhost` PROCEDURE `GetOrderAmount`(  
2   IN pOrderNumber INT  
3 )  
4 BEGIN  
5   SELECT  
6     SUM(quantityOrdered * priceEach)  
7   FROM orderDetails  
8   WHERE orderNumber = pOrderNumber;  
9   END
```

MySQL Stored Procedure Variables

```
mysql> DELIMITER $$
mysql>
mysql> CREATE PROCEDURE GetTotalOrder()
  -> BEGIN
  -> DECLARE totalOrder INT DEFAULT 0;
  ->
  ->     SELECT COUNT(*)
  ->     INTO totalOrder
  ->     FROM orders;
  ->
  ->     SELECT totalOrder;
  -> END$$
Query OK, 0 rows affected (0.01 sec)
```

```
mysql>
mysql> DELIMITER ;
mysql>
```

```
mysql> SELECT
  ->     routine_name
  -> FROM
  ->     information_schema.routines
  -> WHERE
  ->     routine_type = 'PROCEDURE'
  ->     AND routine_schema = 'classicmodels';
```

ROUTINE_NAME
GetAllProducts
GetCustomers
GetOfficeByCountry
GetOrderAmount
GetOrderCountByStatus
GetTotalOrder
SetCounter
sp_name

8 rows in set (0.01 sec)

```
mysql> create table Product
  -> (
  ->     ProductId int primary key,
  ->     ProductName varchar(20) unique,
  ->     ProductQty int,
  ->     ProductPrice float
  -> )
  -> ;
Query OK, 0 rows affected (0.07 sec)
```



```
mysql> insert product values(1,'Printer',10,4500);
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert product values(2,'Scanner',15,3500);
Query OK, 1 row affected (0.71 sec)
```

```
mysql> insert product values(3,'Mouse',45,500) ;
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from product
-> ;
```

ProductId	ProductName	ProductQty	ProductPrice
1	Printer	10	4500
2	Scanner	15	3500
3	Mouse	45	500

3 rows in set (0.00 sec)

```
mysql> Delimiter $$
```

```
mysql> CREATE PROCEDURE prcInsert(
-> in id int,
-> in name varchar(40),
-> in qty int,
-> in price float)
-> begin
-> DECLARE cnt int default 0;
-> DECLARE p float default 0;
-> select cnt=COUNT(ProductId) from Product where pname= name;
-> if cnt>0 THEN
-> update Product set ProductQty = ProductQty + qty where ProductName=name;
-> select p=ProductPrice from Product where ProductName=name;
-> if p<price THEN
-> update Product set ProductPrice=price where ProductName=name;
-> end IF;
-> else
-> insert Product values(id,name,qty,price);
-> end if;
-> end$$
```

```
Query OK, 0 rows affected (0.05 sec)
```