

Part 2

- Welcome back...
- In this video I'll explain the syntax of a stored procedure and then show you how to implement it
- There was a configuration issue with my environment so I've left this in just in case you experience a similar issue

Creating a Stored Procedure

```
DELIMITER //
```

```
CREATE PROCEDURE GetAllProducts()
```

```
BEGIN
```

```
    SELECT * FROM products;
```

```
END //
```

```
DELIMITER ;
```

First we change the delimiter from a
; to //

Next we use the keywords CREATE
PROCEDURE and give the procedure
a name.

BEGIN and END signify the beginning
and end of the procedure.

The // signifies the end of the
procedures.

Change the delimiter back

Demo

```
DELIMITER ^
CREATE PROCEDURE custNo_Amount()
BEGIN
    SELECT
        customerNumber,
        SUM(amount)
    FROM payments
    GROUP BY customerNumber
    ORDER BY
        SUM(amount) DESC;
END ^
DELIMITER ;
```

To execute a Stored Procedure

```
CALL GetAllProducts();
```

```
ERROR 1436 (HY000): Thread stack overrun: 7636  
bytes used of a 131072 byte stack, and 128000  
bytes needed. Use 'mysqld --thread_stack=#' to  
specify a bigger stack.
```



- I'm using UniServer Server
- I needed to increase the thread_stack
- I found the configuration file here: \UniServerZ\core\mysql\my.ini
- I increased and saved the file
- I quit the SQL command prompt and tried again!
- Still didn't work!
- Then I restarted the SQL server. All good.
- You configurations might differ. Most text refers to my.cnf

Demo

Deleting a Stored Procedure

- How do you think you would delete a stored procedure?
 - a) DROP procedure procedureName()
 - b) DELETE procedure procedureName()
 - c) DROP procedure procedureName
 - d) DELETE procedure procedureName

Stored Procedure Parameters

- Many of the stored procedures that you develop will require parameters.
- Makes the stored procedure more flexible and useful.
- In MySQL, a parameter has one of three modes: IN,OUT, or INOUT.

IN Example

```
DELIMITER %%  
CREATE PROCEDURE GetOrderStatusByDate (  
    IN in_date DATE)  
BEGIN  
    SELECT orderNumber, status FROM orders  
    WHERE orderDate = in_date;  
END %%  
DELIMITER ;  
  
CALL GetOrderStatusByDate ('2005-05-30');
```

Demo

```
SELECT
customerNumber,
    SUM(amount)
FROM payments
GROUP BY customerNumber
HAVING SUM(amount) > tot_amount
ORDER BY
    SUM(amount) DESC;
```

OUT Example

```
DELIMITER ^^  
CREATE PROCEDURE GetNumEmployeeBySurname (  
    IN surname VARCHAR(255) ,  
    OUT total INT  
)  
BEGIN  
    SELECT COUNT(lastname)  
    INTO total  
    FROM employees  
    WHERE lastName = surname;  
END ^^  
DELIMITER ;  
  
CALL GetNumEmployeeBySurname ('Patterson', @theCount);
```

End of Part 2

- So I guess all of that so far is clear... if not post in the forum
- I've left you hanging a little... check out the next video for how things get a little trickier and how we can combine a couple of select statements and use multiple parameters with different operations (IN and OUT)