

```

CASE
WHEN Valor = 1 THEN SELECT 'Domingo';
WHEN Valor = 2 THEN SELECT 'Segunda-feira';
WHEN Valor = 3 THEN SELECT 'Terça-feira';
WHEN Valor = 4 THEN SELECT 'Quarta-feira';
WHEN Valor = 5 THEN SELECT 'Quinta-feira';
WHEN Valor = 6 THEN SELECT 'Sexta-feira';
WHEN Valor = 7 THEN SELECT 'Sabado';
END CASE;
END$
DELIMITER ;

CALL sp_dia_semana();

```

A Figura 9.12 exibe o resultado da sub-rotina `sp_dia_semana()`. Observe o uso das condições definidas após `WHEN`.

```

MySQL 5.5 Command Line Client
mysql> CALL sp_nome_mes2();
+-----+
| Setembro |
+-----+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

mysql> DELIMITER $
mysql> CREATE PROCEDURE sp_dia_semana()
-> BEGIN
-> DECLARE Valor INTEGER;
-> SET Valor = DAYOFWEEK(CURDATE());
-> CASE
-> WHEN Valor = 1 THEN SELECT 'Domingo';
-> WHEN Valor = 2 THEN SELECT 'Segunda-feira';
-> WHEN Valor = 3 THEN SELECT 'Terça-feira';
-> WHEN Valor = 4 THEN SELECT 'Quarta-feira';
-> WHEN Valor = 5 THEN SELECT 'Quinta-feira';
-> WHEN Valor = 6 THEN SELECT 'Sexta-feira';
-> WHEN Valor = 7 THEN SELECT 'Sabado';
-> END CASE;
-> END$
Query OK, 0 rows affected (0.00 sec)

mysql> DELIMITER ;
mysql> CALL sp_dia_semana();
+-----+
| Quarta-feira |
+-----+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.02 sec)

mysql>

```

Figura 9.12 - Criação e execução da stored procedure “`sp_dia_semana()`”.

Em relação ao uso do comando `CASE ... WHEN ... THEN` e `END CASE`, a primeira forma é mais simples e dá menos trabalho para codificar. No entanto, é pertinente lembrar que a segunda forma permite faixas de valores quando usar algum operador lógico.

Na sequência são testados os recursos de laços de repetição (*loopings*). Para os exemplos seguintes será utilizada a mesma situação codificada de formas diferentes. Considere que as sub-rotinas desenvolvidas necessitam apresentar o resultado da fatorial de um valor numérico inteiro qualquer.

A sub-rotina seguinte apresenta o valor da fatorial de um número inteiro qualquer utilizando o comando de laço `LOOP` e `END LOOP`. Considere o código seguinte:

```

DELIMITER $
CREATE PROCEDURE sp_fat1(IN Valor INTEGER)
BEGIN
  DECLARE Fator, I INTEGER;
  SET Fator = 1;
  SET I = 1;
  calculo: LOOP

```