14.6.2 Statement Label Syntax

```
[begin_label:] BEGIN
    [statement_list]
END [end_label]

[begin_label:] LOOP
    statement_list
END LOOP [end_label]

[begin_label:] REPEAT
    statement_list
UNTIL search_condition
END REPEAT [end_label]

[begin_label:] WHILE search_condition DO
    statement_list
END WHILE [end_label]
```

Labels are permitted for BEGIN ... END blocks and for the LOOP, REPEAT, and WHILE statements. Label use for those statements follows these rules:

* *begin_label* must be followed by a colon.
* *begin_label* can be given without *end_label*. If *end_label* is present, it must be the same as *begin_label*.
* *end_label* cannot be given without *begin_label*.
* Labels at the same nesting level must be distinct.

* Labels can be up to 16 characters long.

To refer to a label within the labeled construct, use an ITERATE or LEAVE statement. The following example uses those statements to continue iterating or terminate the loop:

```
CREATE PROCEDURE doiterate(p1 INT)

BEGIN

  label1: LOOP

    SET p1 = p1 + 1;

    IF p1 < 10 THEN ITERATE label1; END IF;

    LEAVE label1;

  END LOOP label1;

END;
```

The scope of a block label does not include the code for handlers declared within the block