

12

STORED FUNCTIONS

STORED FUNCTION

- Stored function adalah sekumpulan perintah SQL yang disusun dalam sebuah function (mirip dengan di pemrograman biasa) yang mempunyai nama dan kegunaan tertentu dan memiliki nilai RETURN.
- Keuntungan penggunaan stored function
 - Membuat function yang tidak disediakan oleh MySQL.
- Aturan membuat stored function mirip dengan stored procedure. Perbedaannya adalah adanya nilai RETURN.
- Sebuah Function biasanya digunakan dalam sebuah SQL SELECT.

STORED FUNCTION

- Struktur Pendeklarasian Stored Function adalah

```
CREATE FUNCTION
  nama_function ([[ IN | OUT | INOUT ] nama_param tipe [,...]])
  RETURNS typedata
  isi_FUNCTION;
```

PERBANDINGAN ANTARA STORED PROCEDURE DAN STORED FUNCTION (KASUS PENCARIAN INDEX)

VERSI STORED PROCEDURE

```
DELIMITER //
CREATE PROCEDURE CariIndeks (na double, OUT indeks char(1))
BEGIN
    IF na >= 80 THEN SET indeks = 'A';
    ELSEIF na >= 68 THEN SET indeks = 'B';
    ELSEIF na >= 56 THEN SET indeks = 'C';
    ELSEIF na >= 45 THEN SET indeks = 'D';
    ELSEIF na >= 0 THEN SET indeks = 'E';
    ELSE SET indeks = 'T';
    END IF;
END //
DELIMITER ;
```

```
CALL CariIndeks (78, @idx);
SELECT @idx; -- Menghasilkan B
```

VERSI STORED FUNCTION

```
DELIMITER //
CREATE FUNCTION Indeks(na double) RETURNS CHAR
BEGIN
    IF na>=80 THEN RETURN 'A';
    ELSEIF na>=68 THEN RETURN 'B';
    ELSEIF na>=56 THEN RETURN 'C';
    ELSEIF na>=45 THEN RETURN 'D';
    ELSEIF na>=0 THEN RETURN 'E';
    ELSE RETURN 'T';
    END IF;
END//
DELIMITER ;
```

```
SELECT      Indeks(50) , Indeks(90) , Indeks(75) ;
-- Menghasilkan D,          A,          B
```

PERBANDINGAN ANTARA STORED PROCEDURE DAN STORED FUNCTION (KASUS KETERANGAN INDEX)

VERSI STORED PROCEDURE

- Mencari keterangan berdasarkan Keterangan. Function CariKeterangan

```
DELIMITER //
CREATE PROCEDURE CariKeterangan(indeks char(1), OUT ket varchar(20))
BEGIN
    SET indeks=UPPER(indeks);
    CASE indeks
        WHEN 'A' THEN SET ket='Sangat Baik';
        WHEN 'B' THEN SET ket='Baik';
        WHEN 'C' THEN SET ket='Cukup';
        WHEN 'D' THEN SET ket='Kurang';
        WHEN 'E' THEN SET ket='Sangat Kurang';
        ELSE SET ket='Tidak Valid';
    END CASE;
END //
DELIMITER ;
```

```
CALL CariKeterangan('a',@ket);
select @ket -- Menghasilkan Sangat Baik
```


VERSI STORED FUNCTION

- Mencari keterangan berdasarkan Keterangan. Function Keterangan

```
DELIMITER //
CREATE FUNCTION Keterangan(indeks char(1)) RETURNS VARCHAR(20)
BEGIN
    DECLARE ket VARCHAR(20);
    SET indeks=UPPER(indeks);
    CASE indeks
        WHEN 'A' THEN SET ket='Sangat Baik';
        WHEN 'B' THEN SET ket='Baik';
        WHEN 'C' THEN SET ket='Cukup';
        WHEN 'D' THEN SET ket='Kurang';
        WHEN 'E' THEN SET ket='Sangat Kurang';
        ELSE SET ket='Tidak Valid';
    END CASE;
    RETURN ket;
END //
DELIMITER ;
```

```
SELECT Keterangan('B'),Keterangan(Indeks(63)); // Baik, Cukup
```