

## UPDATE VERSI 2022.05

### A. Hal Baru

Hal baru pada versi 2022.05 adalah:

1. Melakukan insert data tanpa menggunakan isi\_parameter (lihat di B.1)
2. Penambahan metode instant pada model atau metode yang tidak dibuat pada model tetapi dapat digunakan pada control (lihat di B.2)

### B. Cara Penggunaan

#### B.1. Tanpa menggunakan isi\_parameter saat menginsert data

B.1.1 Tanda tanya sebagai pengganti parameter pada syntax sql

##### Model

```
1 public function tambah_data_01($data = [])
2 {
3     $sql = "insert into tb_master (nomor, tanggal) values (?, ?)";
4     $this->db->isi_sql($sql);
5     $this->db->jalankan_sql([$data['n'], $data['t']]);
6 }
```

Nilai tanda tanya (?) pertama di baris-3 adalah \$data['n'] (variabel data index n) dan nilai tanda tanya kedua adalah \$data['t'] yang dimasukkan pada jalankan\_sql di baris ke-5

##### Control

```
1 public function proses_tambah()
2 {
3     $dataku = [
4         'n' => 'n1',
5         't' => '2022-05-27'
6     ];
7     $this->gunakan_model("nm_model")->tambah_data_01($dataku);
8 }
```

Variabel \$dataku akan menjadi nilai variabel \$data di model

### B.1.2 Menggunakan parameter pada syntax sql

Persyaratan, nama-nama parameter di syntax sql harus sesuai dengan index yang ada pada variable \$data

#### Model

```
1 public function tambah_data_02($data = [])
2 {
3     $sql = "insert into tb_master (nomor, tanggal) values (:n, :t)";
4     $this->db->isi_sql($sql);
5     $this->db->jalankan_sql($data);
6 }
```

Nilai parameter :n dan :t pada ke-3 merupakan index yang terdapat pada variabel \$data yang dimasukkan pada jalankan\_sql di baris ke-5.

#### Control

```
1 public function proses_tambah()
2 {
3     $dataku = [
4         'n' => 'n2',
5         't' => '2022-05-27'
6     ];
7     $this->gunakan_model("nm_model")->tambah_data_02($dataku);
8 }
```

Variabel \$dataku akan menjadi nilai variabel \$data di model

### B.1.3 Insert multi data

#### Model

```
1 public function tambah_data_03($data = [])
2 {
3     $sql = "insert into tb_master (nomor, tanggal) values (:n, :t)";
4     $this->db->isi_sql($sql);
5     $this->db->jalankan_sql($data, true);
6 }
```

Nilai parameter :n dan :t pada ke-3 merupakan index yang terdapat pada variabel \$dataku yang dimasukkan pada jalankan\_sql di baris ke-5. Parameter true pada baris ke-5 menandakan bahwa variabel data merupakan variabel 2 dimensi (banyak baris), hal tersebut berguna jika data yang diinsert lebih dari satu baris.

#### Control

```
1 public function proses_tambah()
2 {
3     $dataku = [
4         ['n' => 'n31', 't' => '2022-05-27'],
5         ['n' => 'n32', 't' => '2022-05-27'],
6         ['n' => 'n33', 't' => '2022-05-27'],
7         ['n' => 'n34', 't' => '2022-05-27']
8     ];
9     $this->gunakan_model("nm_model")->tambah_data_03($dataku);
10 }
```

Variabel \$dataku akan menjadi nilai variabel \$data di model

## B.2. Metode instant model

### B.2.1 Jenis-metode instant

Metode-metode berikut dapat dipanggil/digunakan pada control tanpa dibuat pada model:

1. `kj_ambil_semua([tidak wajib])`
2. `kj_ambil_by_id(id yang diinginkan)`
3. `kj_ambil_satu_baris_tertentu(kolom_kriteria, nila_diinginkan)`
4. `kj_ambil_satu_baris_tertentu(kolom_kriteria, nila_diinginkan, [tidak wajib])`
5. `kj_jumlah_baris()`
6. `kj_jumlah_baris_tertentu(kolom_kriteria, nila_diinginkan)`
7. `kj_tambah([data yang akan diinsert], true/false)`
8. `kj_update([data update], kolom_kriteria, nila_kriteria)`
9. `kj_jumlah_hapus(kolom_kriteria, nila_kriteria)`

### B.2.2 Syarat

Persyaratan untuk menggunakan metode instant model:

1. model dibuat dengan melakukan extends (mewarisi) model seperti pada baris ke-1
2. nama tabel harus dijelaskan saat mengimplementasikan database di baris ke-7
3. mengatur properti `atur_db` obyek yang diwarisi pada baris ke-8

```
1 class nm_model extends model
2 {
3     private $dataku;
4
5     public function __construct()
6     {
7         $this->dataku = new database("tb_detail");
8         parent::atur_db($this->dataku);
9     }
10
11     public function nama_metode()
12     {
13         . . . . .
14     }
15
16 }
```

## B.2.2 Cara penggunaan pada control

### 1. Mengambil semua data dari sebuah tabel

```
1 public function abc()  
2 {  
3     $x = $this->gunakan_model("nm_model")->kj_ambil_semua();  
4 }
```

### 2. Mengambil semua data dari sebuah tabel dan mengurutkan berdasarkan kolom unit

```
1 public function abc()  
2 {  
3     $x = $this->gunakan_model("nm_model ")->kj_ambil_semua(['unit']);  
4 }
```

### 3. Mengambil semua data dari sebuah tabel dan mengurutkan berdasarkan kolom unit kemudian harga pada unit yang sama

```
1 public function abc()  
2 {  
3     $x = $this->gunakan_model("nm_model")->kj_ambil_semua(['unit', 'harga']);  
4 }
```

### 4. Mengambil sebuah baris berdasarkan id. Tabel harus memiliki kolom id

```
1 public function abc()  
2 {  
3     $x = $this->gunakan_model("nm_model")->kj_ambil_by_id(10);  
4 }
```

### 5. Mengambil sebuah baris dengan kriteria kolom kode\_barang bernilai BRG\_14

```
1 public function abc()  
2 {  
3     $x = $this->gunakan_model("nm_model")->kj_ambil_satu_baris_tertentu('kode_barang', 'BRG_14');  
4 }
```

6. Mengambil banyak baris dengan kriteria kolom kode\_barang bernilai BRG\_14

```
1 public function abc()  
2 {  
3     $x = $this->gunakan_model("nm_model")->kj_ambil_banyak_baris_tertentu('kode_barang', 'BRG_14');  
4 }
```

7. Mengambil banyak baris dengan kriteria kolom kode\_barang bernilai BRG\_14 lalu diurutkan berdasarkan kolom unit

```
1 public function abc()  
2 {  
3     $x = $this->gunakan_model("nm_model")->kj_ambil_banyak_baris_tertentu('kode_barang', 'BRG_14', ['unit']);  
4 }
```

8. Mengambil banyak baris dengan kriteria kolom kode\_barang bernilai BRG\_14 lalu diurutkan berdasarkan kolom unit kemudian harga pada unit yang sama

```
1 public function abc()  
2 {  
3     $x = $this->gunakan_model("nm_model")->kj_ambil_banyak_baris_tertentu('kode_barang', 'BRG_14', ['unit', 'harga']);  
4 }
```

9. Menghitung jumlah baris data pada sebuah tabel

```
1 Public function abc()  
2 {  
3     $x = $this->gunakan_model("nm_model")->kj_jumlah_baris();  
4 }
```

10. Menghitung jumlah baris data pada sebuah tabel dengan kriteria unit bernilai 20

```
1 public function abc()  
2 {  
3     $x = $this->gunakan_model("nm_model")->kj_jumlah_baris_tertentu('unit', 20);  
4 }
```

11. Menambahkan sebuah baris ke dalam tabel, data yang ditambahkan disusun dengan array 1 dimensi, key pada array (\$dataku) harus sama dengan nama kolom tabel. Variable \$x di baris ke-9 akan bernilai 1 jika berhasil terinsert.

```
1 public function abc()
2 {
3     $dataku = [
4         'nomor' => '001',
5         'kode_barang' => 'BRG_01',
6         'unit' => 50,
7         'harga' => 3000
8     ];
9     $x = $this->gunakan_model("nm_model")->kj_tambah($dataku);
10 }
```

12. Sama dengan nomor 11 (menambahkan sebuah baris). Namun dapat ditambahkan parameter bernilai `true` (lihat baris 9). Jika ditambahkan parameter tersebut, maka variable \$x di baris ke-9 akan menjadi variabel array dengan key status dan id, key status di dalam variabel \$x bernilai 1 jika berhasil tersimpan dan key id bernilai id baris tersimpannya data

```
1 public function abc()
2 {
3     $dataku = [
4         'nomor' => '001',
5         'kode_barang' => 'BRG_01',
6         'unit' => 50,
7         'harga' => 3000
8     ];
9     $x = $this->gunakan_model("nm_model")->kj_tambah($dataku, true);
10 }
```

13. Mengupdate data berdasarkan kriteria tertentu. Variabel `$data` adalah data terbaru, kriteria data yang diupdate adalah kolom `id` bernilai `30`. Key pada array `$dataku` harus sama dengan nama kolom tabel. Nama kolom kriteria dan nilai kriteria dapat diatur sesuai kebutuhan. Variabel `$x` pada baris ke-10 bernilai jumlah baris yang terhapus

```
1 public function abc()  
2 {  
3     $dataku = [  
4         'nomor' => '001',  
5         'kode_barang' => 'BRG_01',  
6         'unit' => 50,  
7         'harga' => 3000,  
8     ];  
9  
10    $x = $this->gunakan_model("nm_model")->kj_update($dataku, 'id', 30);  
11 }
```

14. Menghapus data dengan kriteria. Contoh menghapus data dengan kriteria adalah kolom `id` bernilai `30`. Nama kolom kriteria dan nilai kriteria dapat diatur sesuai kebutuhan. Variabel `$x` pada baris ke-3 bernilai jumlah baris yang terupdate.

```
1 public function abc()  
2 {  
3     $x = $this->gunakan_model("nm_model")->kj_hapus('id', 30);  
4 }
```