

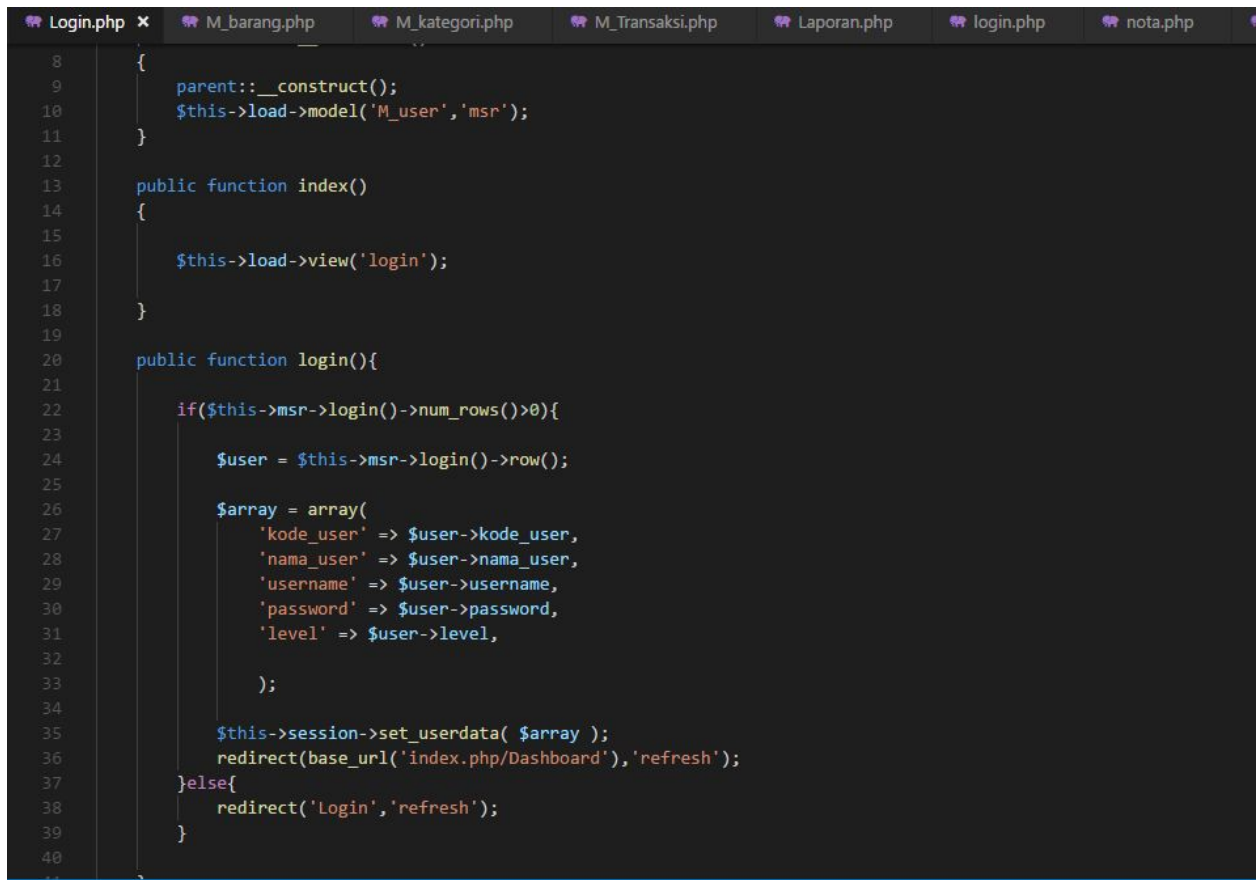
Nama : Firda Reinika

Kelas : XI R2

No : 16

## Laporan Tugas CI

### 1. Login



```
8 {
9     parent::__construct();
10    $this->load->model('M_user','msr');
11 }
12
13 public function index()
14 {
15
16     $this->load->view('login');
17
18 }
19
20 public function login(){
21
22     if($this->msr->login()->num_rows()>0){
23
24         $user = $this->msr->login()->row();
25
26         $array = array(
27             'kode_user' => $user->kode_user,
28             'nama_user' => $user->nama_user,
29             'username' => $user->username,
30             'password' => $user->password,
31             'level' => $user->level,
32
33         );
34
35         $this->session->set_userdata( $array );
36         redirect(base_url('index.php/Dashboard'),'refresh');
37     }else{
38         redirect('Login','refresh');
39     }
40 }
```

Merupakan script di bagian controllers yang berfungsi untuk memanggil login yang ada di views.

### 2. Barang

M\_barang

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class M_barang extends CI_Model {

    public function ambilbarang(){
        $ambilbarang = $this->db->join('kategori', 'kategori.kode_kategori = barang.kode_kategori')->get('barang')->result();

        return $ambilbarang;
    }

    public function ambilkategori(){

        $ambilkategori = $this->db->get('kategori')->result();

        return $ambilkategori;
    }

    public function tambah($nama_file){

        if($nama_file == ""){

            $tambah = array(
                'nama_barang' => $this->input->post('nama_barang'),
                'kode_kategori' => $this->input->post('kategori'),
                'harga' => $this->input->post('harga'),
                'stok' => $this->input->post('stok'), );
        }else{

            $tambah = array(
                'nama_barang' => $this->input->post('nama_barang'),

```

M\_barang merupakan kode dari Modals yang digunakan untuk mengatur tambah data, hapus data dan update data. Script ini berhubungan dengan Barang yang di controllers dan v\_barang yang ada di views.

### 3. Transaksi

```

5
6
7     public function __construct()
8     {
9         parent::__construct();
10        $this->load->model('M_barang','mbk');
11        $this->load->model('M_transaksi','mts');
12    }
13
14
15
16    public function index()
17    {
18
19        if($this->session->userdata('level')){
20
21
22            $data['konten']="v_transaksi";
23            $data['barang']=$this->mbk->ambilbarang();
24            $this->load->view('template', $data, FALSE);
25        }else{
26            redirect('Login','refresh');
27        }
28    }
29
30
31
32
33    public function transaksi(){
34        if($this->input->post('bayar')){
35
36            $this->form_validation->set_rules('pembeli', 'pembeli', 'trim|required');
37            $this->form_validation->set_rules('bayar', 'bayar', 'trim|required|greater_than_equal_to[$this->cart->total()];

```

```

43
44     $id_nota = $this->mts->simpan_db();
45
46     if($id_nota){
47
48         $this->session->set_flashdata('pesan', 'Transaksi '. $this->input->post('pembeli').' berhasil');
49         $data['dts']=$this->mts->ambil_dts($id_nota);
50         $data['ts']= $this->mts->ambil_ts($id_nota);
51
52         $bayar = $this->input->post('bayaru');
53         $total = $this->cart->total();
54         $kembali = $bayar - $total;
55
56         $this->session->set_flashdata('kembali', $kembali );
57         $nama=$this->input->post('pembeli');
58         $this->session->set_flashdata('pembeli', $nama);
59         $this->cart->destroy();
60         $this->load->view('nota', $data);
61
62     }
63     else{
64
65         $nama=$this->input->post('pembeli');
66         $this->session->set_flashdata('pembeli', $nama);
67         $this->session->set_flashdata('pesan', 'anda gagal bertransaksi');
68         redirect('transaksi','refresh');
69
70     }
71
72 }else{
73
74
75     $nama=$this->input->post('pembeli');

```

Transaksi yang ada di controllers sangat berpengaruh dalam kegiatan transaksi yang dilakukan oleh user. Karena berisi script yang digunakan untuk melakukan transaksi, mengambil, membayar dan menyimpan data transaksi yang kemudian data tersebut akan muncul di histori.

#### 4. Card

```

public function index()
{
}

public function tambah_cart($kode_barang){

    $bk = $this->mbk->ambilbarangcart($kode_barang);

    $stok = $bk->stok;

    if($stok == 0){

        $this->session->set_flashdata('pesan', 'Maaf stok Anda Habis Silahkan Hubungi Admin');
        redirect('Transaksi','refresh');

    }else{

        $data = array(
            'id'      => $bk->kode_barang,
            'qty'     => 1,
            'price'   => $bk->harga,
            'name'    => $bk->nama_barang,

        );

        $this->cart->insert($data);
        redirect('Transaksi','refresh');

    }
}

```

Card pada controllers berisi script-script yang berhubungan dengan m\_barang pada modals yang digunakan untuk melakukan tambah,update hapus barang

## 5. M\_transaksi

```

public function simpan_db(){

    for($i=0;$i<count($this->cart->contents());$i++){
        $stok = $this->db->where('kode_barang', $this->input->post('kode_barang')[$i])->get('barang')->row()->stok;
        $qty = $this->input->post('banyak')[$i];
        $sisas = $stok - $qty;
        $u_stok = array('stok' => $sisas, );
        $this->db->where('kode_barang', $this->input->post('kode_barang')[$i])->update('barang', $u_stok);
    }

    $s_transaksi = array('kode_user' => $this->session->userdata('kode_user'),
        'nama_pembeli' => $this->input->post('pembeli'),
        'total' => $this->cart->total(),
        'tanggal_beli' => date('Y-m-d'));

    $this->db->insert('transaksi', $s_transaksi);

    $ts=$this->db->order_by('kode_transaksi', 'desc')
        ->where('nama_pembeli',$this->input->post('pembeli'))
        ->limit(1)
        ->get('transaksi')
        ->row();

    for($i=0;$i<count($this->cart->contents());$i++){

        $s_dttransaksi[] = array('kode_transaksi' => $ts->kode_transaksi,
            'kode_barang'=>$this->input->post('kode_barang')[$i],
            'jumlah' => $this->input->post('banyak')[$i]);
    }

    $db = $this->db->insert_batch('detail_transaksi', $s_dttransaksi);
}

```

M\_transaksi berhubungan dengan v\_transaksi pada views dan Transaksi pada controllers yang berfungsi untuk proses transaksi dan menampilkan data transaksi.

## 6. Database

phpMyAdmin

Server: 127.0.0.1 » Database: reinika

Structure SQL Search Query Export Import Operations Privileges Routines Events More

Filters

Containing the word:

Table	Action	Rows	Type	Collation	Size	Overhead
barang	Browse Structure Search Insert Empty Drop	9	InnoDB	latin1_swedish_ci	32 KiB	-
detail_transaksi	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	48 KiB	-
kategori	Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	16 KiB	-
transaksi	Browse Structure Search Insert Empty Drop	8	InnoDB	latin1_swedish_ci	32 KiB	-
user	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 KiB	-
5 tables	Sum	28	InnoDB	latin1_swedish_ci	144 KiB	0 B

Check all With selected:

Print Data dictionary