|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| CI-4 Minggu ke -18 | 95 |

**Saya Sudah Belajar dan Mengerti dan Saya BISA**

1. INSERT DATA

* Didalam model ketikkan protected $allowedFields = [‘nama kolom’,'nama kolom']; untuk kolom apa aja yang ditambahkan
* Buatlah form insert di views dan arahkan ke function insert di controller
* Buat function insert di controller
* Didalam function insert buat modelnya
* Didalam function insert tambahkan method insert dan tangkap isi inputan dengan $\_POST untuk memasukkan inputan di form kedatabase
* Didalam function insert tambahkan method redirect ke kategori untuk mengarahkan setelah mengirim form kembali ke kategori

2. DELETE DATA

* Tentukan primary key nya yang ada gambar kuncinya
* Ketikkan protected $primaryKey = 'nama kolom primary key'; di model
* Di dalam views select tambahkan tombol hapus dan arahkan ke function delete di controller dan tampilkan id-nya supaya bisa ikut dikirim
* Buat function delete di controller dan tangkap id-nya
* Didalam function delete buat modelnya
* Didalam function delete tambahkan method delete dan tangkap id-nya untuk menghapus data di database
* Didalam function delete tambahkan method redirect ke kategori untuk mengarahkan setelah mengirim form kembali ke kategori

3. UPDATE DATA

* Di dalam views select tambahkan tombol ubah dan arahkan ke function find di controller dan tampilkan id-nya supaya bisa ikut dikirim
* Buat function find di controller
* Didalam function find buat modelnya
* Didalam function find tambahkan method find dan tangkap id-nya untuk mencari data yang sesuai dengan id-nya
* Di dalam views select tambahkan tombol hapus dan arahkan ke function delete di controller dan tampilkan id-nya supaya bisa ikut dikirim
* Didalam function find buat array assosiatif yang isinya judul sama hasil dari method find
* Setelah ditangkap id-nya tampilkan views update menggunakan return view()
* Di file views update buat form dan arahkan ke function update di controller
* Di dalam function update buat modelnya
* Di dalam function update tambahkan method save($\_POST) untuk menyimpan inputan di form kedalam database
* Didalam function update tambahkan method redirect ke kategori untuk mengarahkan setelah mengirim form kembali ke kategori

4. UPLOAD IMAGE

* Buat form di views untuk menginput file gambarnya dan arahkan ke function insert di controller
* Didalam function insert buat variabel file isi dengan $this->request->getFile('nama inputan untuk gambar'); untuk mengambil inputan gambarnya
* Tambahkan method getName() untuk mengetahui namanya
* Tambahkan method move(‘lokasi folder yang diinginkan’) untuk memindahkan lokasi inputan gambar ke dalam folder yang diinginkan

5. SESSION

* Buat function construct untuk inisialisasinya yaitu $this->session = \Config\Services::session(); supaya langsung dijalankan
* Buat function create
* Didalam function create buat array assosiatif isi dengan data yang ingin dimasukkan kedalam session
* Didalam function create ketik $this->session->set($nama array); untuk membuat session
* Buat function read
* Didalam function read ketik $this->session->get(nama key di array assosiatif yang ingin dipanggil); Untuk menampilkan session
* Buat function delete
* Di dalam function delete ketik $this->session->remove(nama key di array assosiatif yang ingin dihapus); untuk menghapus session yang dipilih
* Buat function destroy
* Di dalam function destroy ketik $this->session->destroy(); untuk menghapus semua session

6. Validation

* Di dalam model ketik protected $validationRules =[

'nama kolom tabel'=> 'peraturannya'

]; untuk menambahkan peraturan saat input

* Di dalam model ketik protected $validationMessages =[

'nama kolom tabel' =>[

'Peraturannya'=>'Pesan jika dilanggar'

]

]; Untuk menambahkan pesan jika inputan tidak sesuai peraturan

* Peraturan alpha\_numeric\_space adalah inputan tidak boleh symbol
* Peraturan min\_length[berapa huruf] adalah inputan minimal berapa huruf
* Peraturan is unique adalah data tidak boleh sama
* Di function insert di controller gunakan if jika insert bernilai salah maka ketik $error = $model->errors(); dan tampilkan echo $error[‘nama kolom’]

7. Menambahkan bootstrap

8. Paging

* Nyalakan Service-nya yaitu $pager = \Config\Services::pager();
* Buat modelnya
* Buat array assosiatif
* Di dalam array assosiatif ketik 'kategori'=>$model->paginate(berapa data setiap page,'group1' (nama harus sama dengan di file select)), 'pager'=>$model->pager
* Di file select ketik <?= $pager->links('group1' (nama harus sama dengan di file controller),'bootstrap (nama harus sama dengan di file pager)') ?>
* Di file paging copy di CI-4 yang menampilkan halaman paging
* Di file pager ketik 'bootstrap (nama harus sama dengan di file select)' => 'Lokasi file paging',

9. View cell digunakan untuk memotong motong views

* Buat file di view cell untuk bagian yang akan dipotong seperti bagian navigasi
* Buat function di controller yang digunakan untuk memanggil file view cell
* Cara memanggilnya <?= view\_cell(‘alamat function di controller’)?>

10. Session Flash Data

* Di dalam if di function insert di controller redirect ke function create agar jika false tetap kedalam form input
* Copy session Flash data di CI-4
* Ubah variabel menjadi function dengan cara mengahapus tanda $ dan menambahkan ()
* Di dalam if di function insert di controller ketik session()->setFlashdata('info',$error['kategori']);
* Didalam file insert di views ketik echo session()->getFlashdata('info');Untuk menampilkan pesan yang disimpan di flash data

**Saya Belum Mengerti**

1. Kenapa Method save() setelah di bootstrap tidak berfungsi