# Bài thực hành 4 XÂY DỰNG MENU CRUD CHO SLIDER

# create

## SLIDER controller

/////////////////////////////////////////////////////////////////////////////////////

// GET: Admin/Slider/Create

public ActionResult Create()

{

ViewBag.OrderList = new SelectList(slidersDAO.getList("Index"), "Order", "Name");

return View();

}

// POST: Admin/Slider/Create

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create(Sliders sliders)

{

if (ModelState.IsValid)

{

//Xu ly cho muc Order

if (sliders.Order == null)

{

sliders.Order = 1;

}

else

{

sliders.Order = sliders.Order + 1;

}

//Xu ly cho muc Slug

string slug = XString.Str\_Slug(sliders.Name);[[1]](#footnote-1)

//xu ly cho phan upload hình ảnh

var img = Request.Files["img"];//lay thong tin file

if (img.ContentLength != 0)

{

string[] FileExtentions = new string[] { ".jpg", ".jpeg", ".png", ".gif" };

//kiem tra tap tin co hay khong

if (FileExtentions.Contains(img.FileName.Substring(img.FileName.LastIndexOf("."))))

//lay phan mo rong cua tap tin

{

//ten file = Slug + phan mo rong cua tap tin

string imgName = slug + img.FileName.Substring(img.FileName.LastIndexOf("."));

sliders.Image = imgName;

//upload hinh

string PathDir = "~/Public/img/slider/";

string PathFile = Path.Combine(Server.MapPath(PathDir), imgName);

img.SaveAs(PathFile);

}

}//ket thuc phan upload hinh anh

//Xu ly cho muc CreateAt

sliders.CreateAt = DateTime.Now;

//Xu ly cho muc CreateBy

sliders.CreateBy = Convert.ToInt32(Session["UserId"]);

slidersDAO.Insert(sliders);

//Thong bao thanh cong

TempData["message"] = new XMessage("success", "Thêm danh mục thành công");

return RedirectToAction("Index");

}

ViewBag.OrderList = new SelectList(slidersDAO.getList("Index"), "Order", "Name");

return View(sliders);

}

## create.cshtml

@model MyClass.Model.Sliders

@{

ViewBag.Title = "Thêm mới Slider";

Layout = "~/Areas/Admin/Views/Shared/\_LayoutAdmin.cshtml";

}

@using(Html.BeginForm("Create","Slider",FormMethod.Post,new{@enctype="multipart/form-data"}))

{

@Html.AntiForgeryToken()

@Html.ValidationSummary(true, "", new { @class = "text-danger" })

<!-- Content Wrapper. Contains page content -->

<div class="content-wrapper">

<!-- Main content -->

<section class="content my-2">

<!-- Default box -->

<div class="card">

<div class="card-header">

<div class="row">

<div class="col-md-6">

<strong class="text-danger">THÊM MỚI SLIDER</strong>

</div>

<div class="col-md-6 text-right">

<a href="@Url.Action("Index", "Slider")" class="btn-sm btn-info"> <i class="fa fa-long-arrow-alt-left"></i> Quay lại</a>

</div>

</div>

</div>

<div class="card-body">

@\*Phan NOI DUNG\*@

<div class="row">

@\* Phần nhập tay \*@

<div class="col-md-9">

<div class="form-group">

<label>Tên Slider <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.EditorFor(model=>model.Name,new{htmlAttributes=new{@class="form-control"}})

@Html.ValidationMessageFor(model => model.Name,"",new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Mô tả chi tiết <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.TextAreaFor(model => model.MetaDesc, new { @class = "form-control", @rows = "2" })

@Html.ValidationMessageFor(model => model.MetaDesc, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Từ khóa <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.TextAreaFor(model => model.MetaKey, new { @class = "form-control", @rows = "2" })

@Html.ValidationMessageFor(model => model.MetaKey, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.URL, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.URL, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.URL, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Ảnh <span class="text-danger"></span></label>

<div class="col-md-10">

<input name="img" type="file" class="form-control" />

</div>

</div>

</div>

@\* Phần lựa chọn \*@

<div class="col-md-3">

<div class="form-group">

<label>Vị trí <span class="text-danger"></span></label>

<div class="col-md-10">

@Html.DropDownListFor(model => model.Position, new[] { new SelectListItem { Text = "Slider", Value = "Slider" }, new SelectListItem { Text = "Footer", Value = "Footer" } }, new { @class = "form-control" })

@Html.ValidationMessageFor(model => model.Position, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Sắp xếp <span class="text-danger"></span></label>

<div class="col-md-10">

@Html.DropDownListFor(model => model.Order, (SelectList)@ViewBag.OrderList, "Sắp xếp", new { @class = "form-control" })

@Html.ValidationMessageFor(model => model.Order, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Trạng thái <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.DropDownListFor(model => model.Status, new[] { new SelectListItem { Text = "Không xuất bản", Value = "2" }, new SelectListItem { Text = "Xuất bản", Value = "1" } }, new { @class = "form-control" })

@Html.ValidationMessageFor(model => model.Status, "", new { @class = "text-danger" })

</div>

</div>

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<button class="btn-sm btn-success" type="submit" name="create"> Lưu [Thêm] <i class="fa fa-save"></i> </button>

</div>

</div>

</div><!-- /.card-body -->

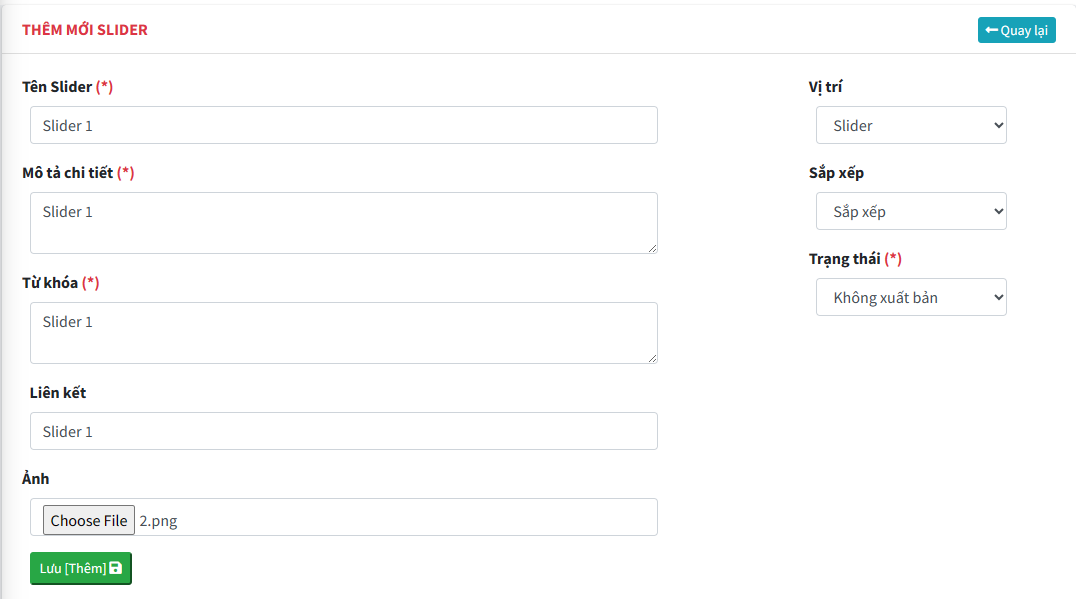
</div> <!-- /.card -->

</section><!-- /.content -->

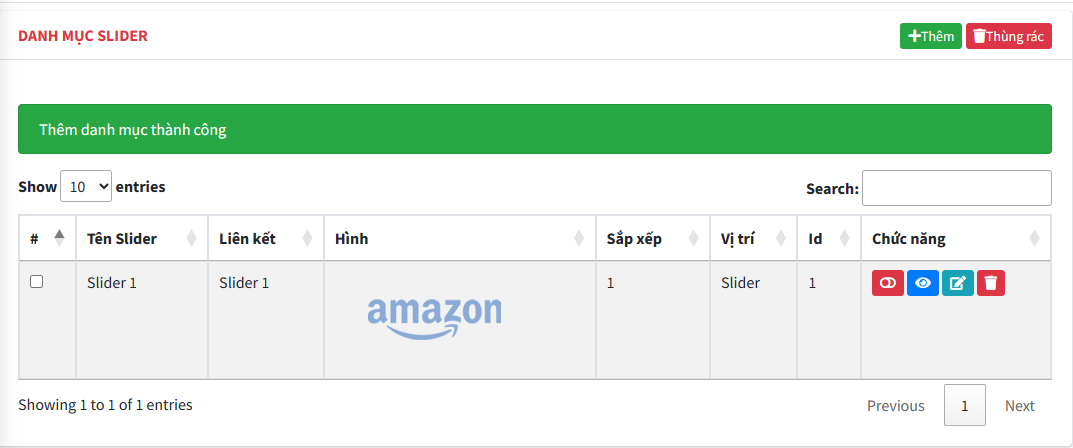
</div><!-- /.content-wrapper -->

}

# Kết quả



## 



1. Do Slider không quản lý slug [↑](#footnote-ref-1)