MVC Notes

Models

* prop + tab to make Model properties quickly
* Just like making a normal object class
* Example

Public class Movie {

public int Id { get; set; }

public string Name { get; set; }

}

Controllers

* Example

// GET: Movies/Random

public ActionResult Random() {

var movie = new Movie() { Name = “Shrek” };

return View(movie);

}

Views

* Views naming convention is the same name of its corresponding controller, without “Controller” at the end
* Example

@model AppName.Models.Movie

@{

}

<h2>@Model.Name</h2>

General

* Themes
  + Go to bootswatch.com/lumen/
  + Under Lumen tab, select bootstrap.css
  + Download and add it under the content folder
  + Change bootstrap to bootstrap-lumen under BundleConfig file
* Action Results
  + ViewResult – View()
  + PartialViewResult – PartialView()
  + ContentResult – Content()
  + RedirectResult – Redirect()
  + RedirectToRouteResult – RedirectToAction()
  + JsonResult – Json()
  + FileResult – File()
  + HttpNotFondResult – HttpNotFound()
  + EmptyResult – return new EmptyResult() (no helper method)
* Action parameters
  + Appear in the url as such
    - Controller/Action?param1=2&param2=52
* Custom Routes (OLD AND OUTDATED, USE ATTRIBUTED ROUTING INSTEAD)
  + Add above default route
  + Example

routes.MapRoute(

“MoviesByReleaseDate”,

“movies/released/{year}/{month}”,

new { controller = “Movies”, action = “ByReleaseDate” }

new { year = @”\d{4}”, month = @”\d{2}” }

);

* Attributed Routing
  + Activate it by calling routes.MapMvcAttributeRoutes();
  + In the controller, do this above the desired action
    - [Route(“movies/released/{year}/{month:regex(\\d{2}):range(1, 12)}”)]
* ViewModels
  + Needs a new folder, doesn’t go in Models folder
  + Name ends in “ViewModel”
* PartialViews
  + Prefix names with underscore (\_Navbar)
  + Similar to React components (reusable components)
  + @Html.Partial(“\_Navbar”) to render it in the markup