



Microsoft®
.NET



UNIT 10: ASP MVC FUNDAMENTALS



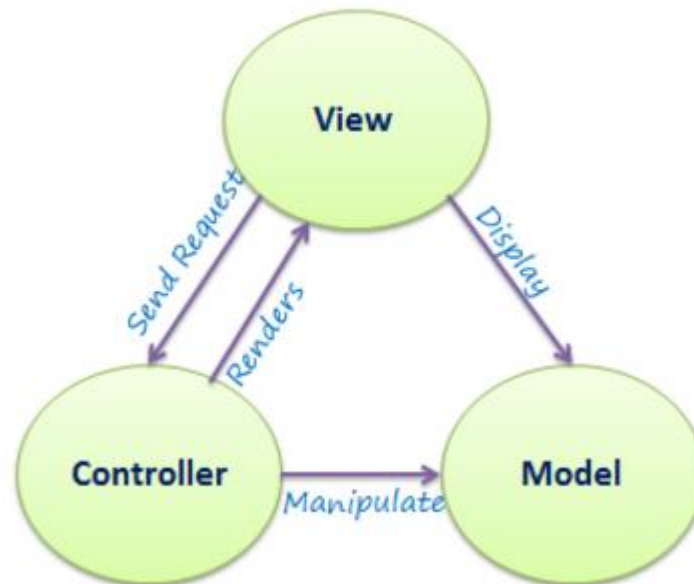
CONTENTS

- Introduction to ASP.MVC
- MVC Overview
- ASP.NET MVC vs Web Form
- MVC Application structure
- ASP.NET MVC Convention
- Routing in MVC



Introduction to MVC

- MVC stands for Model View Controller.
- Standard Architectural Pattern
- Separation of concerns: model, view, controller





Introduction to MVC

- **Models**
 - Business/domain logic
 - Model objects, retrieve and store model state in a persistent storage (database).
- **Views**
 - Display application's UI
 - UI created from the model data
- **Controllers**
 - Handle user input and interaction
 - Work with model
 - Select a view for rendering UI

Introduction to MVC

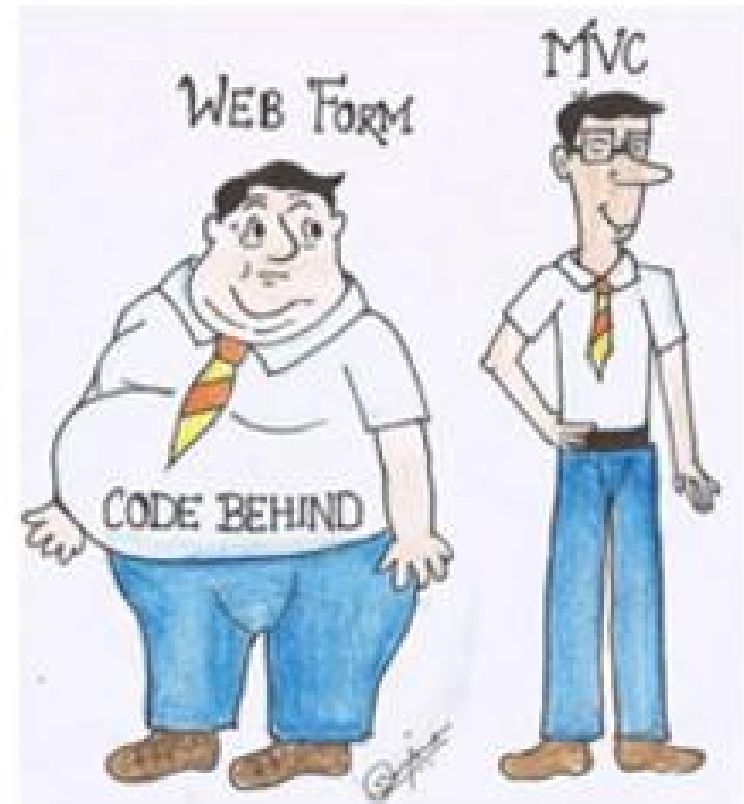


- ASP.NET MVC Version History:
 - Microsoft had introduced ASP.NET MVC in .Net 3.5
 - Versions: MVC 1.0,2.0,3.0,4.0,5.0,5.2
- MVC 5 is included with VS 2013

ASP.NET Web Forms vs MVC



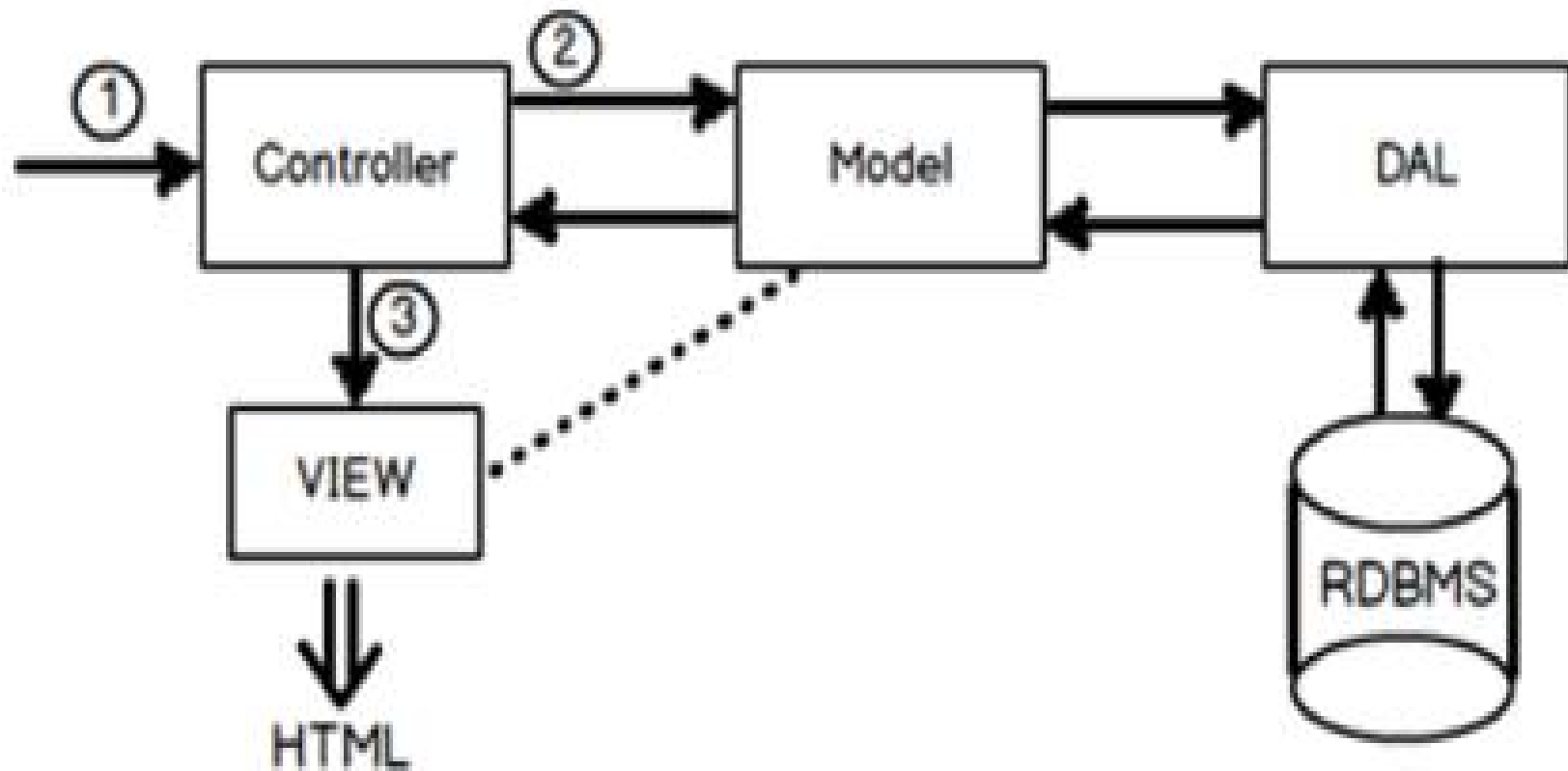
- Web Forms:
 - UI (.aspx)
 - Code Behind (aspx.cs)
- Problems:
 - Response time
 - Bandwidth consumption
- Solutions:
 - Get rid of
 - Code Behind
 - Server controls



ASP.NET Web Forms vs MVC



- Asp.Net MVC request flow



MVC Application structure



- Create a first MVC app

/Controllers	Where you put Controller classes that handle URL requests
/Models	Where you put classes that represent and manipulate data and business objects
/Views	Where you put UI template files that are responsible for rendering output, such as HTML
/Scripts	Where you put JavaScript library files and scripts (.js)
/fonts	The Bootstrap template system includes some custom web fonts, which are placed in this directory
/Content	Where you put CSS, images, and other site content, other than scripts
/App_Data	Where you store data files you want to read/write
/App_Start	Where you put configuration code for features like Routing, bundling, and Web API

MVC-BasicTutorials

- ▶ Properties
- ▶ References
 - App_Data
 - App_Start
 - Content
 - Controllers
 - fonts
 - Models
 - Scripts
 - Views
- ▶ favicon.ico
- ▶ Global.asax
- ▶ packages.config
- ▶ Project_Readme.html
- ▶ Web.config

ASP.NET MVC Convention

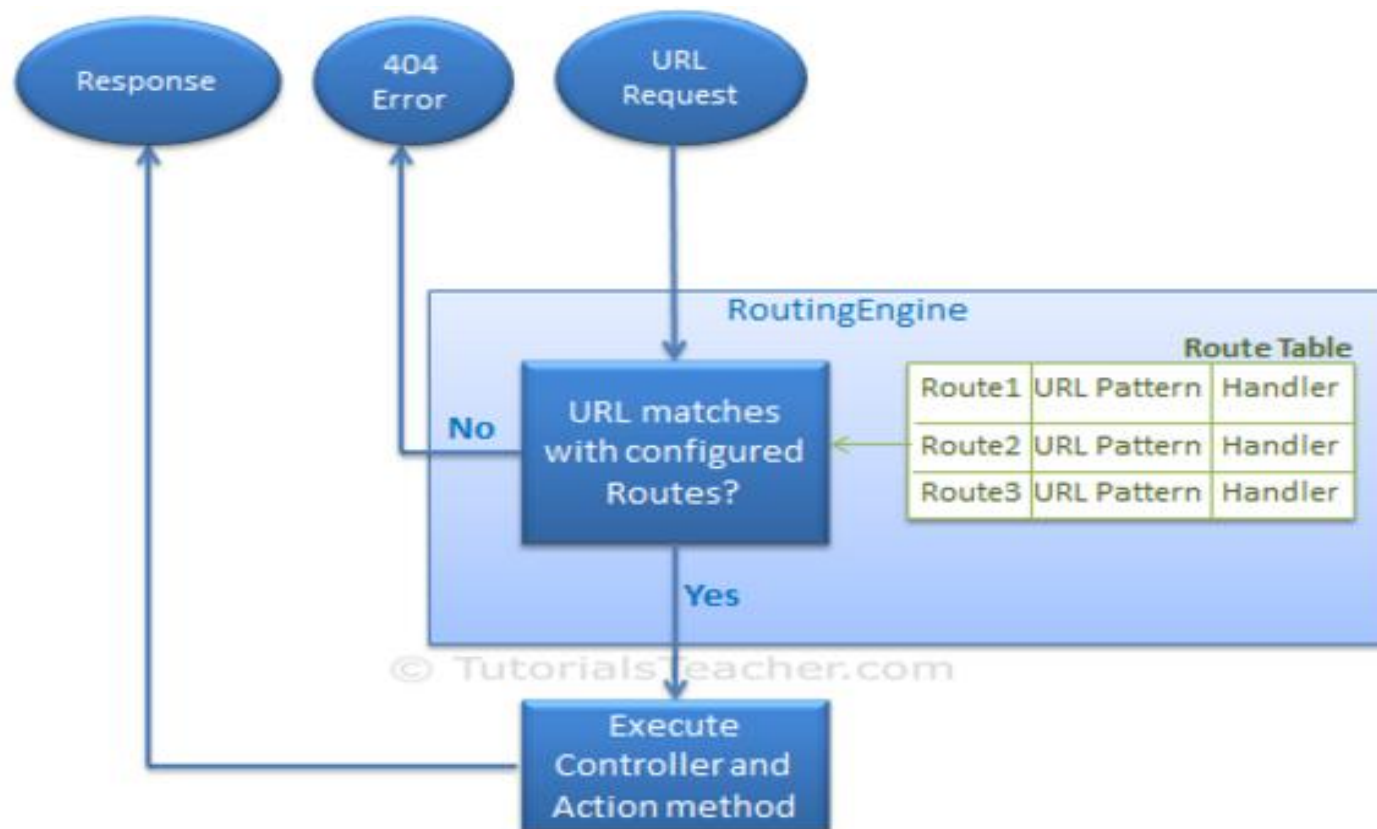


- Each controller's class name ends with Controller and lives in the Controllers directory
- There is a single Views directory for all the views of your application
- Views that controllers use live in a subdirectory of the Views main directory and are named according to the controller name

Routing in MVC



- Routing enable us to define URL pattern that maps to the request handler.



Routing in MVC



- Configure Route

```
public class RouteConfig
{
    public static void RegisterRoutes(RouteCollection routes)
    {
        routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

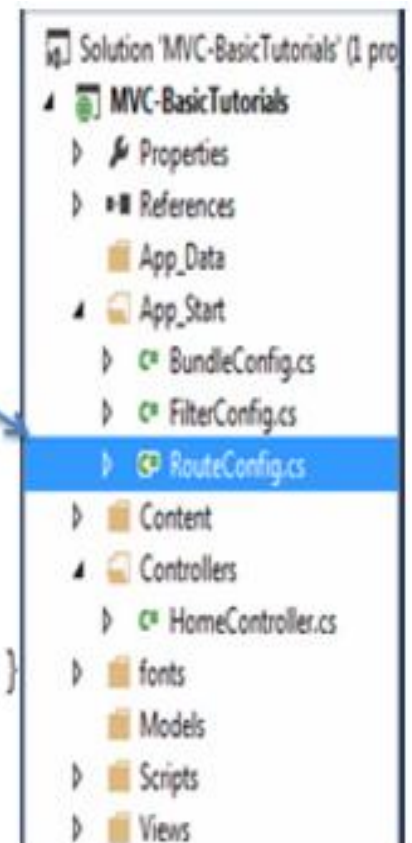
        routes.MapRoute(
            name: "Default",
            url: "{controller}/{action}/{id}",
            defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional }
        );
    }
}
```

Annotations in the code:

- Route to ignore* points to `routes.IgnoreRoute`
- Route name* points to `name: "Default"`
- URL Pattern* points to `url: "{controller}/{action}/{id}"`
- Defaults for Route* points to the `defaults` object

© TutorialsTeacher.com

RouteConfig.cs





Routing in MVC

- URL Partern

Controller *Action method*
`http://localhost:1234/home/index/100` *Id parameter value*

Controller *Action method*
`http://localhost:1234/home/index`

- Multiple Routes
- Route Constraints
- Register Routes