Session 5

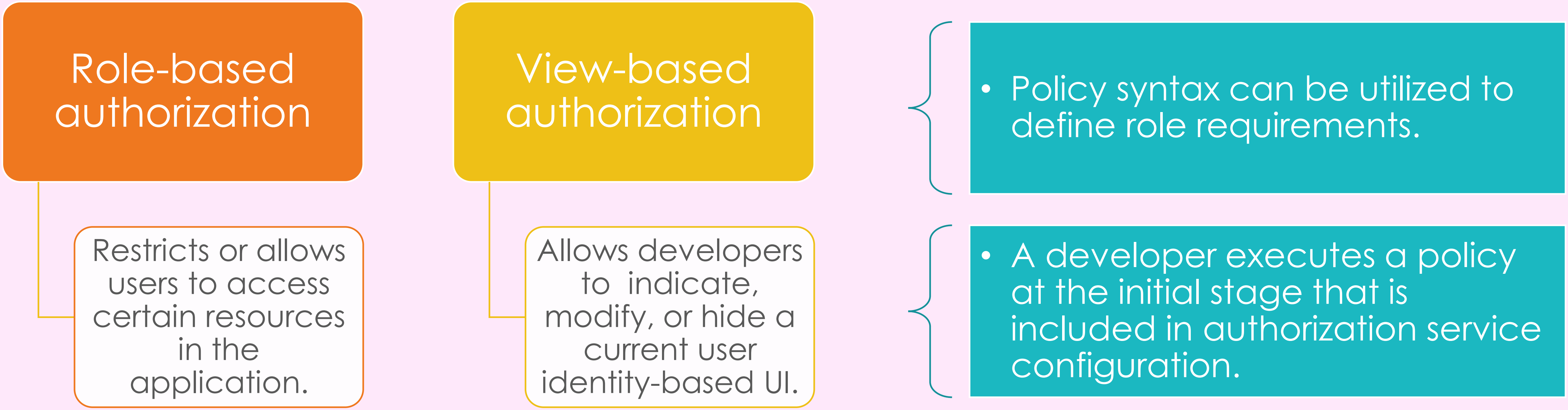
More on ASP.NET MVC and Core MVC

Session Overview

- Explain Role-based and View-based authorization
- Describe ASP.NET Selectors
- Explain ASP.NET Helpers
- List Action Filters
- Identify usage of Apply Action Filter and Custom Filter
- Explain ASP.NET MVC Security
- Describe Views and Partial Views in MVC

Role-based and View-based Authorization

Policy-based Role Checks



Role-based authorization

Restricts or allows users to access certain resources in the application.

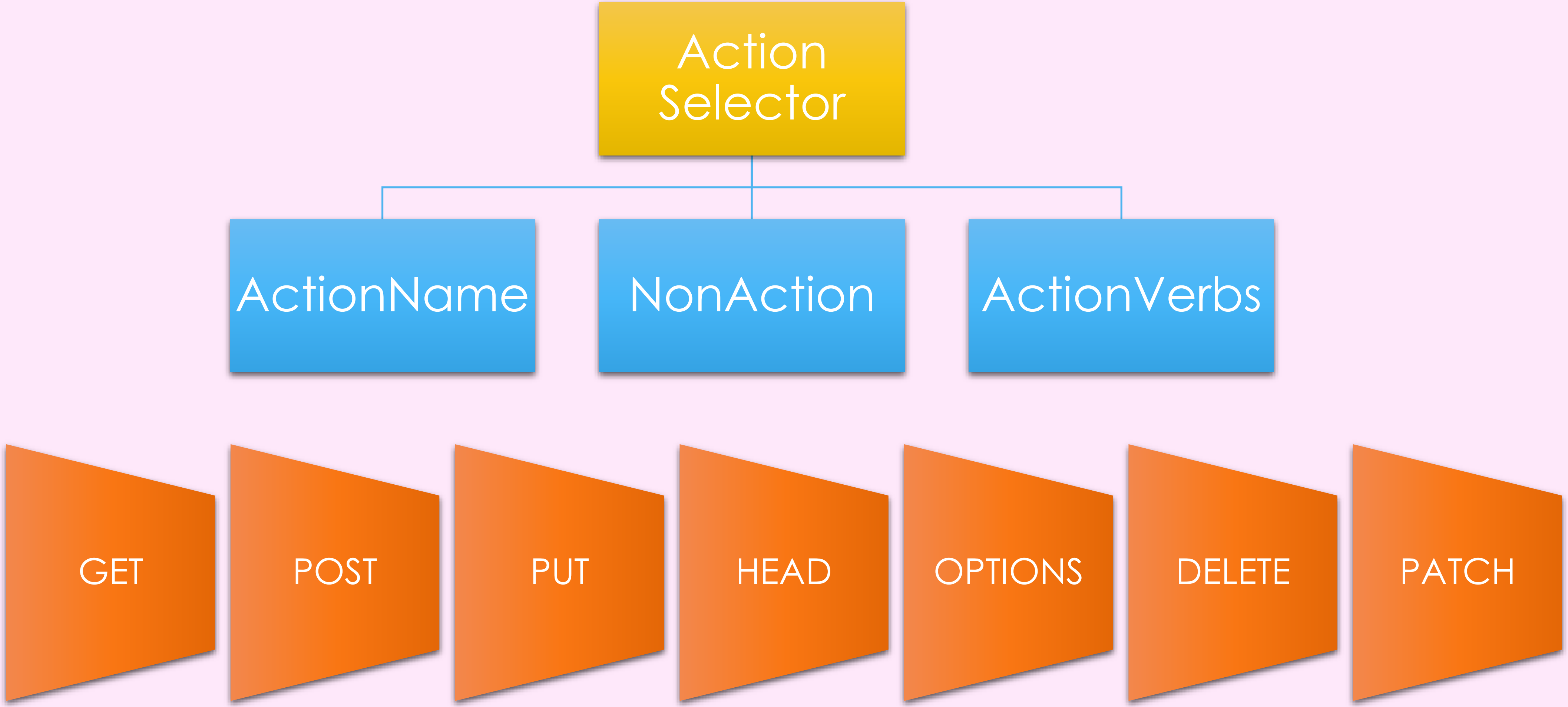
View-based authorization

Allows developers to indicate, modify, or hide a current user identity-based UI.

- Policy syntax can be utilized to define role requirements.

- A developer executes a policy at the initial stage that is included in authorization service configuration.

ASP.NET Selectors



Action Selector

ActionName

NonAction

ActionVerbs

GET

POST

PUT

HEAD

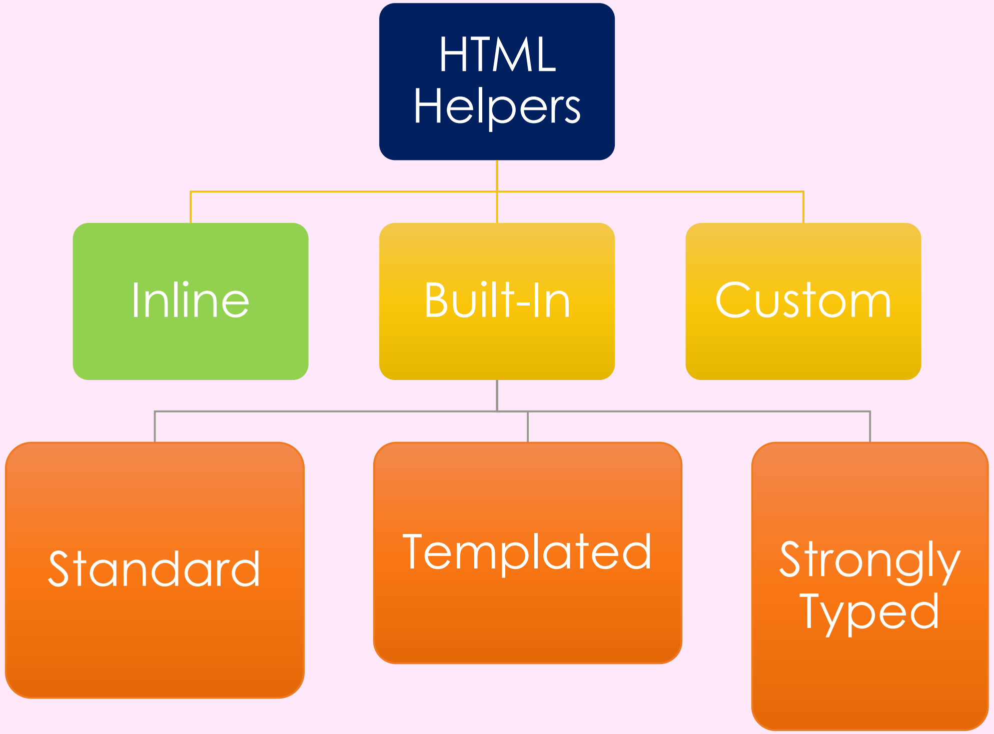
OPTIONS

DELETE

PATCH

HTTP methods

ASP.NET Helpers



HTML Helpers

Inline

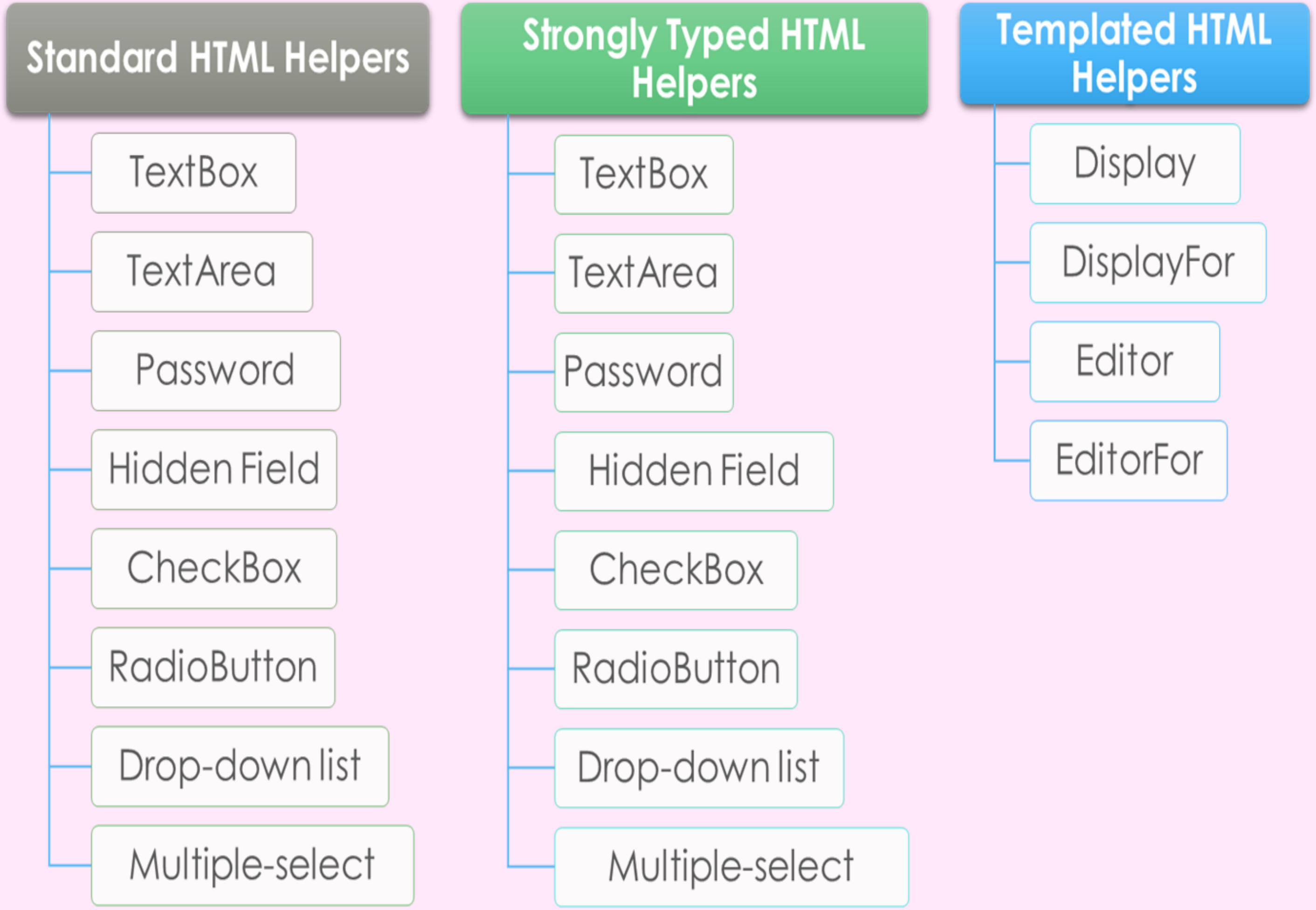
Built-In

Custom

Standard

Templated

Strongly Typed



Standard HTML Helpers

- TextBox
- TextArea
- Password
- Hidden Field
- CheckBox
- RadioButton
- Drop-down list
- Multiple-select

Strongly Typed HTML Helpers

- TextBox
- TextArea
- Password
- Hidden Field
- CheckBox
- RadioButton
- Drop-down list
- Multiple-select

Templated HTML Helpers

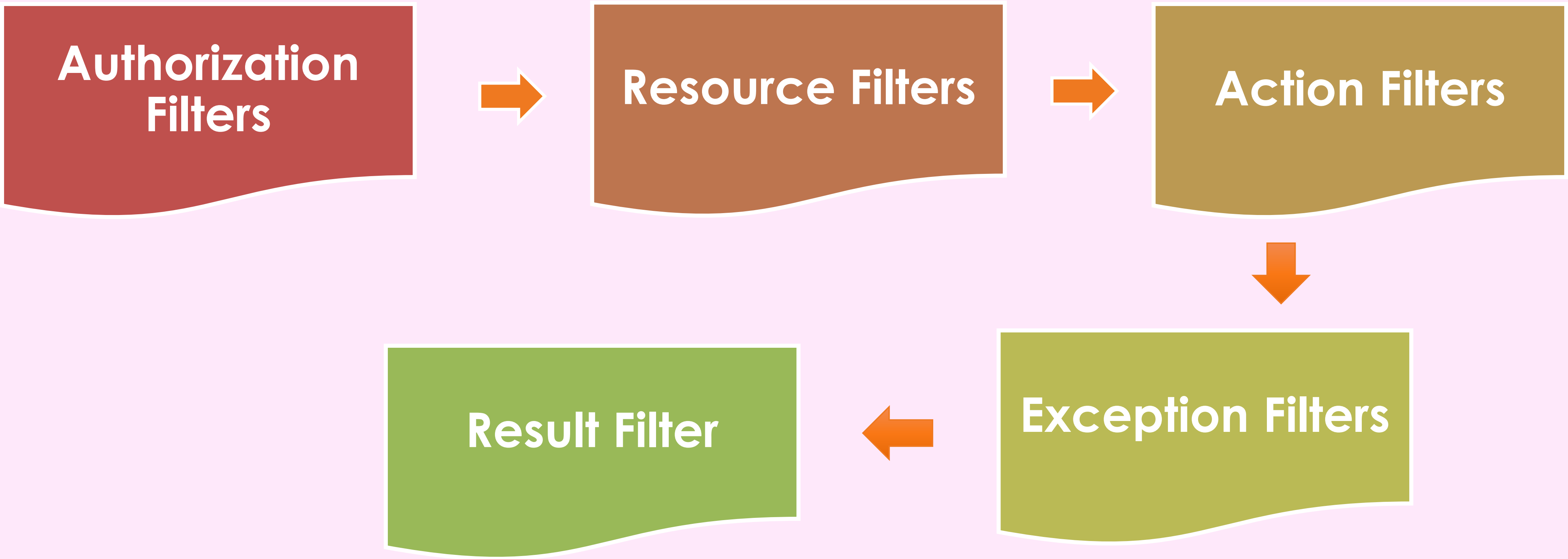
- Display
- DisplayFor
- Editor
- EditorFor

Introduction to Filters

Scenarios of When to Use Filters

- Authentication/Authorization
- Login Details of Users
- Logging
- Handling of Errors
- Data Caching
- Data In/Out

Different Types of Filters



Authorization Filters

Resource Filters

Action Filters

Exception Filters

Result Filter

Configuring Filters

Global Level

A developer can restrict access for every Web API controller by adding the AuthorizeAttribute filter to the global filter list.

Controller Level

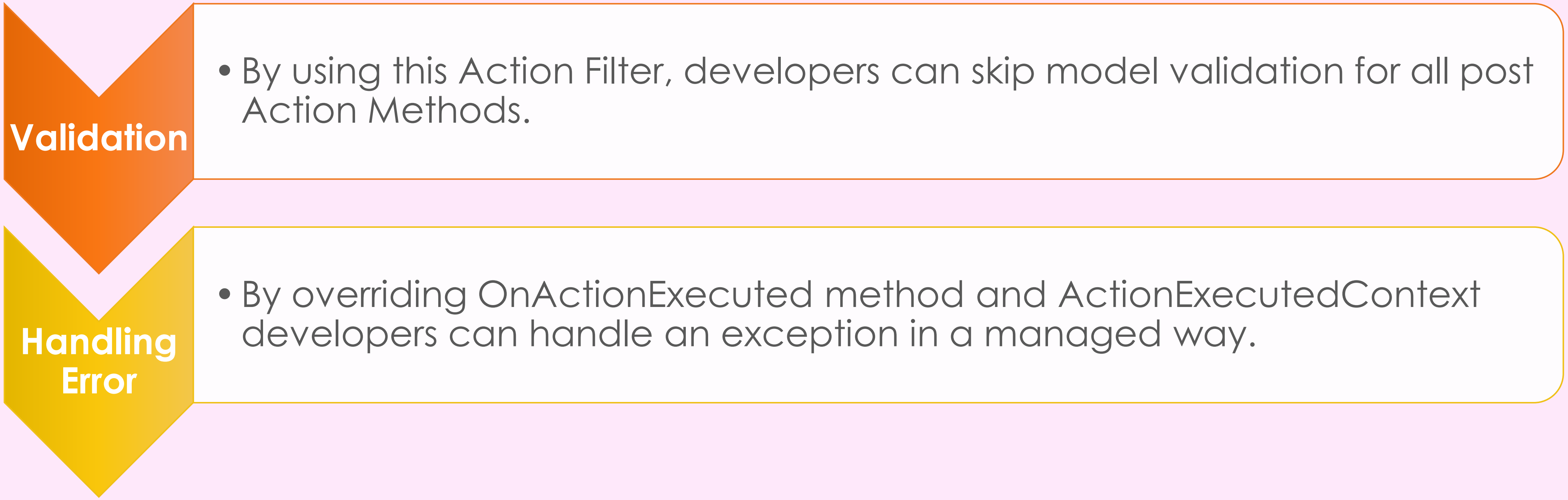
This can be achieved by locating the filter on the top of the controller name.

Action Level

This can be achieved by locating the filter on top of the action name.

More on Action Filters

Action filters for:



Validation

- By using this Action Filter, developers can skip model validation for all post Action Methods.

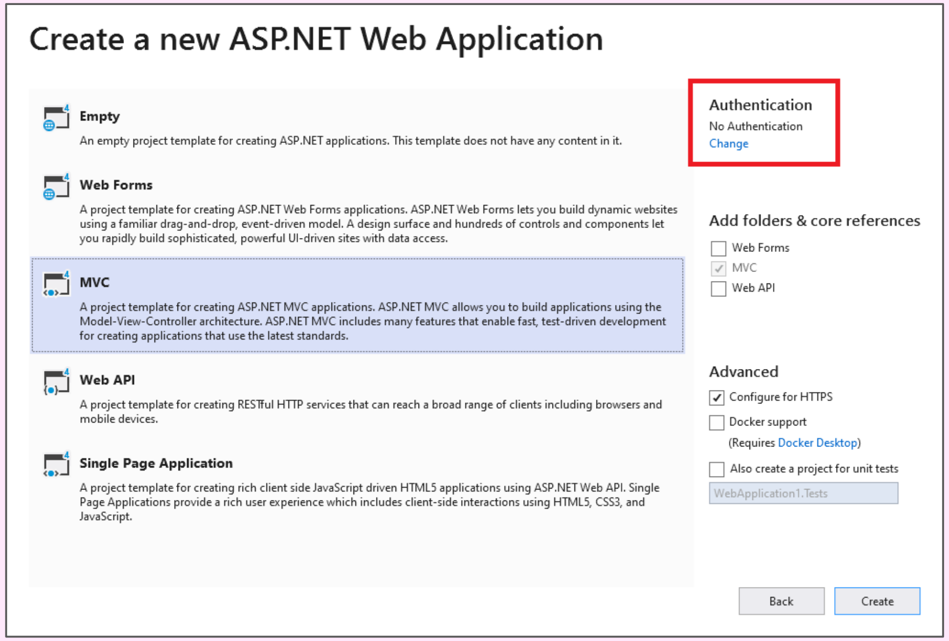
Handling Error

- By overriding OnActionExecuted method and ActionExecutedContext developers can handle an exception in a managed way.

Resource Filters

In case developers want to short-circuit any action method, they can implement IResourceFilter filter attribute.

ASP.NET MVC Security



Create a new ASP.NET Web Application

Empty
An empty project template for creating ASP.NET applications. This template does not have any content in it.

Web Forms
A project template for creating ASP.NET Web Forms applications. ASP.NET Web Forms lets you build dynamic websites using a familiar drag-and-drop, event-driven model. A design surface and hundreds of controls and components let you rapidly build sophisticated, powerful UI-driven sites with data access.

MVC
A project template for creating ASP.NET MVC applications. ASP.NET MVC allows you to build applications using the Model-View-Controller architecture. ASP.NET MVC includes many features that enable fast, test-driven development for creating applications that use the latest standards.

Web API
A project template for creating RESTful HTTP services that can reach a broad range of clients including browsers and mobile devices.

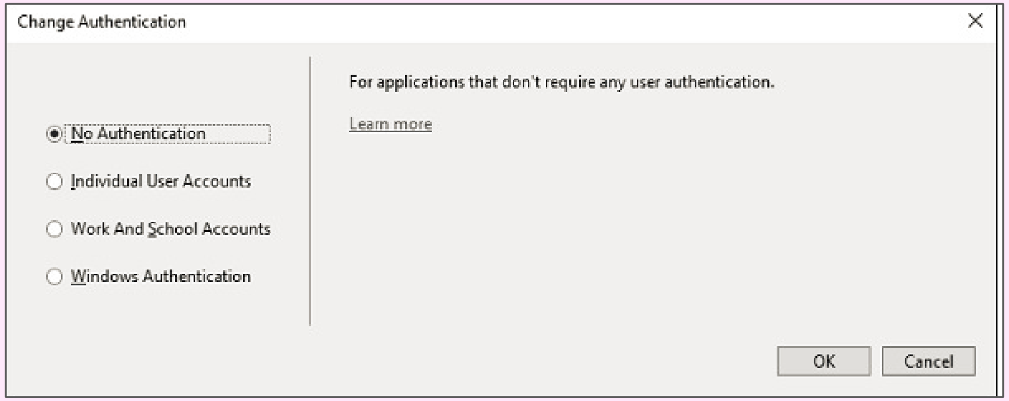
Single Page Application
A project template for creating rich client side JavaScript driven HTML5 applications using ASP.NET Web API. Single Page Applications provide a rich user experience which includes client-side interactions using HTML5, CSS3, and JavaScript.

Authentication
No Authentication
Change

Add folders & core references
□ Web Forms
☑ MVC
□ Web API

Advanced
☑ Configure for HTTPS
□ Docker support
(Requires Docker Desktop)
□ Also create a project for unit tests
WebApplication1.Tests

Figure 5.1: Change Authentication Button



Change Authentication

● No Authentication
○ Individual User Accounts
○ Work And School Accounts
○ Windows Authentication

For applications that don't require any user authentication.

Figure 5.2: Change Authentication Window

No Authentication

Individual User Accounts

Windows Authentication

Work and School Accounts

Types of Authentication

Partial Views in MVC

Partial view is an exclusive view capable of rendering a portion of the view content. Let's see how different it is from a View:

View	Partial View
It includes a layout page.	It does not include a layout page.
It renders viewstart page before rendering any view.	It does not look for viewstart.cshtml. Within the _viewstart.cshtml page, no common code for a partial view can be placed.
It includes markup tags, such as HTML, body, head, title, meta, and so on.	It does not contain any markup as it renders within the view.

Summary

- MVC is a framework that helps developers to create Web applications in which sections of code are organized by the functions they perform.
- The three basic components of MVC are Model, View, and Controller.
- The two life cycles of MVC are the application lifecycle and request lifecycle.
- The fundamental pattern component of an MVC application is the component called view. It is accountable for rendering the user interface, irrespective of it being an HTML or a UI widget on a desktop application.
- A model can be defined as a collection of classes that help developers to work with data and business logic.
- Validation forms a vital aspect in ASP.NET MVC applications. It helps developers to assess if the user input is valid.
- Controllers form the central unit of the ASP.NET MVC application. It is the first unit that any incoming HTTP Request communicates with. It also has the responsibility of deciding which model to choose.
- The MVC design pattern is definitely a better approach to create software applications. Projects created using MVC model consume less expenditure and time too.