**PADRÃO MVC**

O padrão MVC **(Model, View e Controller)** juntos eles formam um padrão arquitetural, diferente de estilo arquitetural. No **estilo arquitetural** definimos uma arquitetura por completa. No **padão arquitetural** ela aplica uma aplicação em cima de uma arquitetura.

Nesse padrão temos uma separação de responsabilidades, cada parte da estrutura é responsável por um processo.

Essa arquitetura está presente desde 1976 criado pela Xerox, sendo um padrão aberto aplicando-o em diversas plataformas.

**Controller:** é responsável de fazer as interações com as outras partes do sistema, banco de dados, models e views.

**Model:** é a classe que terá todas as propriedades da classe.

**View:** é a exibição das informações na tela para o usuário.

**ROTAS**

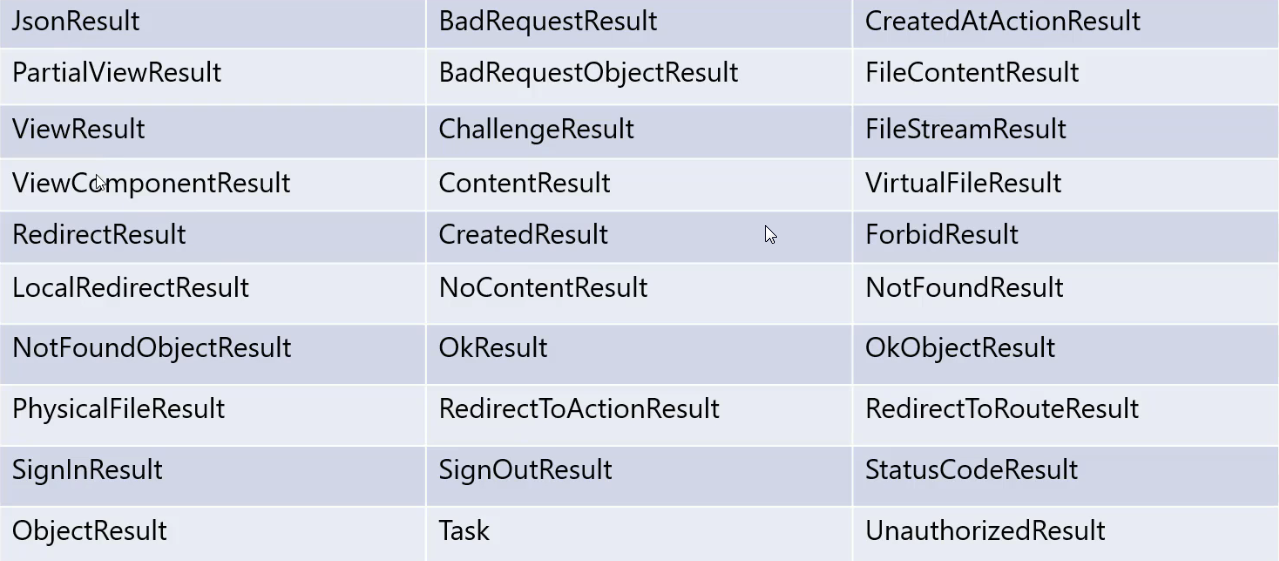
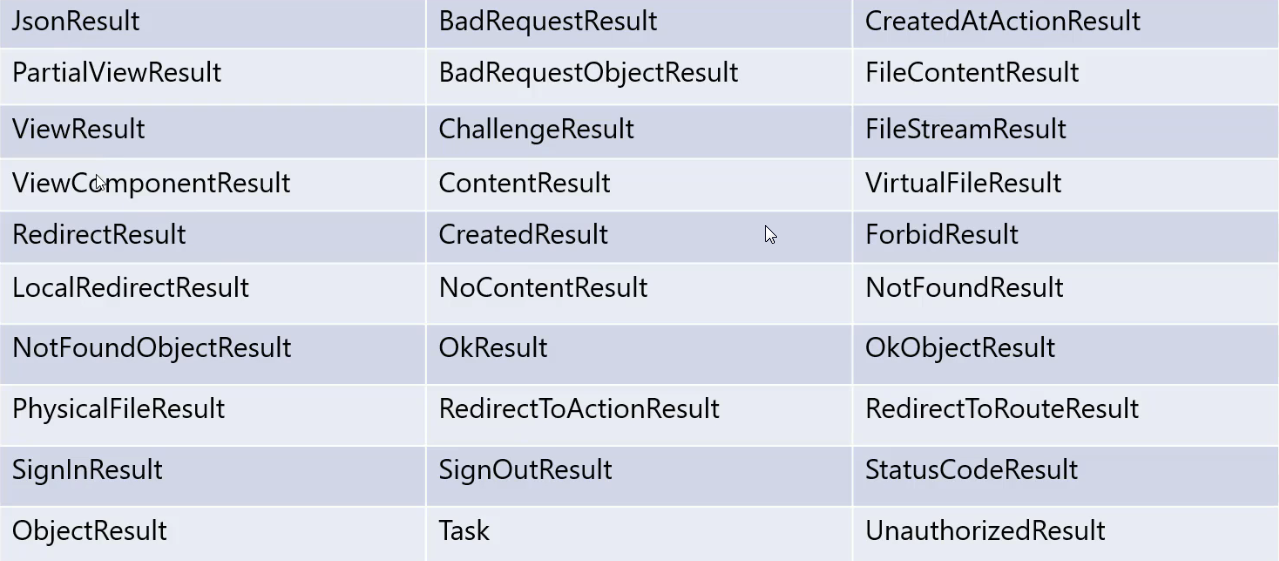
Rotas são basicamente **estruturas de navegação** personalizadas para que a URL da aplicação possua determinado padrão e atenda as necessidades de passagem de parâmetros.

As rotas possuem um **name**, possui um nome de rota, podendo ter outros. **Template**: mostra o caminho padrão que deve ser seguido na aplicação.

**Attribute Routes**, é uma maneira diferente de trabalhar com ortas, tornando-as mais flexíveis e fáceis de personalizar. Elas são personalizadas diretamente nas controllers. É uma boa recomendação, pois, com ela trabalhamos com personalização isoladas, não para o projeto todo.

**ACTION RESULTS**

No ASP.NET Core, uma **Action Result** é o tipo de retorno da action da Controller, é utilizada a interface IActionResult. que pode retornar alguns tipos de resultados.



**PROTOCOLOS HTTP – VERBOS BÁSICOS**

* **Get:** Um request (pedido) de uma informação ao server. É feito através da URL.
* **Post:** Envia informações ao servidor (formulários).
* **Put:** Similar ao Post, envia informações ao servidor. Sua diferença é por ser utilizado para atualização de informações existentes.
* **Delete:** Solicita a exclusão de uma informação no servidor através da URL indicada.
* **LEITURA DAS DEFINIÇÕES DOS METÓDOS HTTPS.**