# Теоретическая часть

* Привязка простых данных к методу действия контроллера
* Привязка сложных типов данных
* Привязка коллекция
* Задание источника для привязки данных к методу действия
* Понятие о компонентах представлений
* Создание компонентов и применение компонентов в разметке
* **POCO**-компонент (**P**lain **O**ld **C**LR **O**bject)
* Построение компонента как расширение класса **ViewComponent**
* Типы результата метода **Invoke()** компонентов представлений
* Размещение разметки компонента представления – в папке представлений контроллера или в папке **\_Shared**
* Передача параметров в компонент (в метод **Invoke()**) из представления
* Асинхронные компоненты
* Гибридные компоненты – контроллер с методом **Invoke()** и атрибутом **ViewComponent**

# Практическая часть

Создать приложение ASP.NET Core MVC. Стилизацию приложения выполняйте при помощи фреймворка Bootstrap. Примените страницу компоновки. Не забывайте создавать модель/модели для решения задачи. По GET-запросу **/Home/Index** выводите этот текст.

**Задача 1.** Требуется при помощи сервиса получать коллекцию из 42х персонажей, используя [**ресурс**](https://swapi.dev/) при помощи запроса [**https://swapi.dev/api/people/**](https://swapi.dev/api/people/). Обратите внимание, что выдается несколько записей, для загрузки требуемого количества придется использовать модификацию запроса. Какую именно – разберитесь по выдаче API. Сохранять в коллекции все поля, добавьте поле идентификатора. Коллекцию выводить в компонент представления в табличном виде, выводите не более 8 полей. Компонент и сервис рекомендуется сделать асинхронными. Обработка в коде компонента – вывод среднего веса персонажей, разности минимального и максимального весов.

* По кнопке, размещенной в табличном выводе компонента, на отдельной странице вывести подробную информацию о персонаже (все поля), вывод выполнить в еще один компонент представления
* По кнопке в интерфейсе страницы выбрать персонажей с заданным диапазоном веса, параметры выборки передавать в компонент представления вывода коллекции персонажей
* По кнопке в интерфейсе страницы выбрать персонажей с заданным названием «родной» планеты, параметры выборки передавать в компонент представления вывода коллекции персонажей
* По кнопке в интерфейсе страницы выводить в компонент представления коллекции персонажей, упорядоченной по возрастанию массы

**Задача 2.** Требуется при помощи сервиса получать коллекцию из 33х планет, используя [**ресурс**](https://swapi.dev/) при помощи запроса [**https://swapi.dev/api/planets/**](https://swapi.dev/api/planets/). Обратите внимание, что выдается несколько записей, для загрузки требуемого количества придется использовать модификацию запроса. Какую именно – разберитесь по выдаче API. Сохранять в коллекции все поля, добавьте поле идентификатора. Коллекцию выводить в компонент представления в табличном виде, выводите не более 8 полей. Компонент и сервис рекомендуется сделать асинхронными. Компонент должен быть гибридным. Обработка в коде компонента – вывод минимального, среднего и максимального диаметра планет; минимального, среднего и максимального орбитального периода планет.

* По кнопке, размещенной в табличном выводе компонента, на отдельной странице вывести подробную информацию о планете (все поля), вывод выполнить в еще один компонент представления
* По кнопке в интерфейсе страницы выбрать планеты с заданным типом поверхности, параметр выборки передавать в компонент представления вывода коллекции планет
* По кнопке в интерфейсе страницы выбрать планеты с заданным диапазоном значений диаметра планеты, параметры выборки передавать в компонент представления вывода коллекции планет
* По кнопке в интерфейсе страницы выводить в компонент представления вывода коллекции планет, сведения о планетах, упорядоченные по убыванию массы,

Разработайте сервис логирования, сохраняющий в файле **application.log** в папке **App\_Data** факты создания и редактирования сведений о клиентах: момент времени операции с точностью до секунды, собственно данные клиента (включая закодированный пароль).

Разработайте глобальный фильтр исключений – сервис с внедрением зависимостей. Параметры исключений записывайте в файл журнала при помощи сервиса логирования.

# Дополнительно

Запись занятия можно скачать по [**этой ссылке**](https://cloud.mail.ru/public/bhgA/bSXe9nWZS), материалы занятия в этом же архиве.