


План занятий










Модули:

- 1. Blazor и создание интерактивных веб-приложений
- 2. Использование внешнего API в веб-приложении
- 3. Создание Web Api и интеграция с Blazor
- 4. Основы MsSQL, Management Studio, Dapper (Database First)
- 5. Основы Entity Framework, генерация БД, взаимодействие с Asp Net (Code First)
- 6. Доп. материал: кэширование данных, авторизация и аутентификация, локализация и глобализация

Аннотации

- ×2 - 2 недели на данную тему
-  - практические занятия

Занятия:

- 1. Настройка рабочего окружения, создание первого проекта Blazor, теория по Blazor
- 2. Повторение основ HTML ×2
- 3. Повторяем/вспоминаем CSS, глобальные стили, внедрение стилей в станицы/компоненты
- 4. События html-элементов
- 5. Привязка данных (bind)
- 6. Директивы управления (@foreach , @if) ×2
- 7.  создание одностраничного веб-приложения ×2
- 8. Компоненты, компонентно-ориентированная архитектура
- 9. Передача данных между компонентами, параметры и события компонентов ×2
- 10. Страницы, навигация, навигаторы
- 11.  рефакторинг предыдущего проекта с использованием компонентов, многостраничный дизайн (с добавлением доп. функционала на остальные страницы) ×2
- 12. События жизненного цикла приложения, обновление состояния и перерисовка (Refresh)
- 13. Использование сторонних библиотек компонентов (на примере MudBlazor) ×2
- 14. Взаимодействие с внешним API, HTTP запросы на основе REST
- 15.  создание веб-приложения на основе готового API (криптовалюты, погода, расписание транспорта или кинопоиск) ×2
- 16. Рефакторинг приложения по принципам SOLID, сервисы и их жизненный цикл, инъекция сервисов в Blazor
- 17. Создание первого приложения Asp WebApi, разбор структуры файла Program, обзор контроллеров WebApi по умолчанию (шаблон WeatherForecast)
- 18. Углубляемся в контроллеры: типы запросов (Post , Get и т.д.), пути к котроллерам (/api/users), пути к конечным точкам (/api/users/all), запросы и ответы в формате json
- 19. Передача аргументов в пути к конечной точке (/api/users/10 или /api/users?id=10)
- 20.  написание своего WebApi
- 21. Взаимодействие проектов Blazor и WebApi
- 22. Повторяем/вспоминаем основы sql запросов
- 23. MsSQL Management Studio
- 24.  создаем и заполняем БД по выбранной теме (несколько таблиц, внешние ключи, нормализация)
- 25. Взаимодействие ORM Dapper с MsSQL
- 26. Вспоминаем про сервисы, регистрация и инъекция сервисов
- 27.  связываем созданную ранее БД с Dapper, продумываем WebApi для доступа к данным
- 28. Основы и краткий обзор Entity Framework, создаем контекст данных и интегрируем его в Asp Net (маленький пример с парой таблиц)
- 29. Внешние ключи в EF Core, функции для получения и изменения данных, трекинг объектов
- 30.  проектируем и реализуем бд для диплома (на выбор либо EF Core, либо Dapper и нативный SQL), пишем репозитории для взаимодействия с данными ×2
- 31.  реализация Web-Api на основе готовой БД
- 32.  создаем UI на Blazor, связываем его с WebApi ×2 (или даже ×3)