1. Создайте WEB-приложение на ASP.NET Core, на тему «Автопарк». Подключите к нему свою БД (минимум 5 таблиц, по 3 атрибута). Добавьте контроллер с несколькими действиями и представлениями для них.
2. Создайте WEB-приложение на ASP.NET Core. Подключите к нему свою БД (минимум 5 таблиц, по 3 атрибута). Реализуйте CRUD операции через MVC.
3. Создайте WEB-приложение на ASP.NET Core. Реализуйте авторизацию и регистрацию в приложении через MVC, с переходом на главную страницу сайта.
4. Создайте WEB-приложение на ASP.NET Core. Подключите к нему свою БД (минимум 5 таблиц, по 3 атрибута). Реализуйте выгрузку данных на сайт с помощью MVC.
5. Создайте WEB-приложение на ASP.NET Core. Подключите к нему свою БД (минимум 3 таблицs, по 3 атрибута). Добавьте модели с аннотациями валидации. Настройте отображение ошибок в представлениях.
6. Создайте WEB-приложение на ASP.NET Core, на тему «Супермаркет». Подключите к нему свою БД (минимум 5 таблиц, по 3 атрибута). Добавьте контроллер с несколькими действиями и представлениями для них.
7. Создайте WEB-приложение на ASP.NET Core, на тему «Магазин муз. инструментов». Подключите к нему свою БД (минимум 5 таблиц, по 3 атрибута). Добавьте контроллер с несколькими действиями и представлениями для них.
8. Реализуйте веб приложение калькулятор и конвертер валют на Java Spring. Используйте аннотации @GetMapping и @PostMapping, а также @RequestParam.
9. Реализуйте архитектуру Spring проекта, создайте 3 модели данных на любую тему (нельзя чтобы тема совпадала с той, которая была у вас на дисциплине). Реализуйте Crud действия для этих моделей.
10. Реализуйте архитектуру Spring проекта, подключите базу данных к проекту, создайте 3 модели данных на любую тему (нельзя чтобы тема совпадала с той, которая была у вас на дисциплине). Реализуйте Crud действия для этих моделей.
11. Реализуйте Spring проект с использованием Swagger API на тему “Аквапарк”. Минимум 3 таблиц необходимо реализовать.
12. Реализуйте Spring проект с использованием Swagger API на тему “Госпиталь”. Минимум 3 таблицы необходимо реализовать. Добавьте пояснения к вашей API.
13. Реализуйте Spring проект с использованием Swagger API на тему “Кинотеатр”. Минимум 3 таблицы необходимо реализовать. Добавьте авторизацию в вашу API через Quath2 или JWT.
14. Создайте Spring веб-приложение, реализуйте в нём авторизацию и регистрацию. Также добавьте страницу на которое после успешной авторизации будет переходить пользователь. Обязательно использование Spring Security.
15. Реализуйте архитектуру Spring проекта, подключите базу данных к проекту, создайте 3 модели данных на любую тему (нельзя чтобы тема совпадала с той, которая была у вас на дисциплине). Реализуйте Crud действия для этих моделей. Сделайте связи для моделей, минимум 2 типа.
16. Что такое ASP.NET CORE? Расскажите о его преимуществах.
17. Чем отличается контроллер от представления (View) в ASP.NET Core?
18. Какие способы аутентификации и авторизации доступны в ASP.NET Core?
19. Как можно подключить базу данных к проекту ASP.NET Core?
20. Как реализовать обработку ошибок и логирование в ASP.NET Core?
21. Что нужно сделать для того, чтобы ваши данные из БД перенеслись в ваш проект ASP.NET Core?
22. Что такое RESTful API?. Расскажите о примере его использования
23. Кэширование данных. Что это и для чего используется?
24. Тестирование проекта. Виды его тестирования и с помощью чего оно происходит?
25. Объясните архитектуру MVC в контексте ASP.NET Core. Какие основные компоненты входят в модель MVC?
26. Что такое валидация данных, как она используется в ASP.NET Core?
27. Как организовать работу с файлами, например, загрузку изображений, в проекте ASP.NET Core?
28. Объясните принцип работы View в ASP.NET Core. Как передать данные из контроллера во View?
29. Как выполнить операции CRUD в ASP.NET Core? Какие инструменты для этого используются?
30. Model, View, Controller. Основные понятия. Модели. Контроллеры. Представления.
31. Что такое MVC. Нарисуйте во время ответа на доске схему работы данного архитектурного паттерна.
32. Что такое MVC. Дайте определение Controller в данной архитектуре. Расскажите про аннотации @Controller и @RequestMapping
33. Что такое MVC. Расскажите про аннотации @GetMapping, @PostMapping, @RequestParam.
34. Расскажите анатомию Spring проекта. Расскажите про директорию Controller. Какой код в ней должен писаться и по какому принципу.
35. Расскажите анатомию Spring проекта. Расскажите про директорию Service. Какой код в ней должен писаться и по какому принципу. Почему мы используем два сервиса?
36. Расскажите анатомию Spring проекта. Расскажите про директорию Model. Какой код в ней должен писаться и по какому принципу.
37. Расскажите анатомию Spring проекта. Расскажите про директорию Repository. Какой код в ней должен писаться и по какому принципу.
38. Что такое JDBC. Расскажите как подключить базу данных к Spring проекту.
39. Что такое JDBC. Расскажите какие зависимости нужно подключить для работы с базой данных, опишите их конкретную пользу (минимум 3)
40. Что такое JDBC. Расскажите как изменяется Spring проект при работе с базой данных.
41. Что такое API? Дайте определению Swagger и расскажите про его применение в работе с API.
42. Что такое API? Какие зависимости необходимо подключить для работы со Swagger, расскажите их конкретную пользу и зачем нужна каждая (минимум 3)
43. Что такое API? Как реализовать пояснения к API и также как скрыть метод из Swagger
44. Дайте определение Spring Security. Как именно он реализует защиту сайта?
45. Что такое JDBC. Подробно расскажите как установить связи для данных в Spring проекте.
46. Дайте определение Spring Security. Зачем нужны файлы MvcConfig и WebSecurityConfig, что в них должно находиться?
47. Что такое Java Spring? Расскажите о плюсах и минусах его использования.