1. Что такое MVC?

Паттерн MVC включает ряд компонентов:

* Модели определяют структуру и логику используемых данных
* Представления (views) определяет визуальную часть, как данные будут отображаться
* Контроллеры обрабатывают входящие http-запросы, используя для обработки модели и представления, и отправляет в ответ клиенту некоторый результат обработки, нередко в виде html-кода.
* Система маршрутизация как дополнительный компонент сопоставляет запросы с маршрутами и выбирает для обработки запросов определенный контроллер.

1. Поясните принципы разработки приложения в соответствии с паттерном MVC.
2. Поясните назначение маршрутизатора.

сопоставляет запросы с маршрутами и выбирает для обработки запросов определенный контроллер

1. Что такое endpoint?

комбинация URI i HTTP метода (GET http://localhost:3000/weather, POST http://localhost:3000/posts и т.д.)

# 1. Поясните понятие **Шаблонизация**.

Позволяет использовать html-шаблоны для генерации конечных html-страниц

# 2. Поясните понятие **Шаблон (template).**

**Шаблоны** обычные файлы HTML в формате handlebars, в которых с помощью специального синтаксиса выводятся передаваемые данные

# 3. Поясните понятие **Контекст шаблона (template context).**

**Контекст шаблона** содержит все переменные шаблона

# 4. Поясните понятие **Макет (layout).**

**Макет** оформление, которое пишем в отдельный файл и указываем, куда вставлять

# 5. Поясните понятие **Частичное представление (partial-view).**

**Частичные представления**, - разделяемые общие элементы, которые можно добавлять на обычные представления.

# 6. Поясните понятие **Хелпер** (**helper)**.

**Хелперы** функции, которые возвращают некоторую строку. После получения эту строку можно добавить в любое место представления

# 7. Перечислите встроенные хелперы Handlebars.

**If** Вывод блока по условию. Если аргумент этого хелпера вернет значение false, то блок внутри хелпера не будет выполнен

**Unless** Хелпер обратный хелперу if. Блок будет выведен, если аргумент вернет значение false.

**Each** Хелпер для перебора списков

**With** Позволяет перепривязать контекст

# 8. Поясните понятие **Статический файл**.

**Статика** - это файлы, которые лежат на сервере и отдаются пользователю в том виде, в каком есть (HTML, CSS, JS, изображения, другие файлы).

# 9. Поясните назначение http-заголовков: **Cache-control, ETag, Last-Modified**, **If**-**Modified-Since, If-None-Match, Expires.**

**Cache-Control** задания инструкций кэширования для запросов и ответов

**ETag** идентификатором специфической версии ресурса позволяет более эффективно использовать кэш, позволяя серверу отправлять не весь ответ, если содержимое не изменилось.

**Last-Modified** в ответе HTTP содержит время, когда запрашиваемый ресурс был изменен

**If-Modified-Since** указывает время, за которое браузер впервые загрузил ресурс с сервера

**If-None-Match** используется для обновления тегов сущностей на сервере

**Expires** содержит дату/время, по истечении которой ответ сервера считается устаревшим

# 10. Поясните свойство **\_\_dirname** объекта **global**.

**\_\_dirname** возвращает путь к папке, в которой находится текущий файл JavaScript.

# 11. Поясните назначение и принцип работы метода **response.download().**

Функция **res.download()** браузеры предлагают пользователю загрузить переданный файл

# 12. Поясните назначение и принцип работы метода **response.attachment().**

**«Content-Type»** будет автоматически установлен на основе расширения файла

заголовок **«Content-Disposition»** будет установлен на «filename= filename».

# 13. Поясните назначение http-заголовка: **Content-Disposition: attachment.**

указывает, что файл должен быть загружен; браузеры предоставляют диалоговое окно «Сохранить как», предварительно заполненное со значением параметров filename если они есть