1. Поясните понятие **Шаблонизация**.

механизм генерации текстового файла на основе готового шаблона.

1. Поясните понятие **Шаблон (template).**

статический текст с вкраплениями специальных символов (комбинацией символов), предназначенных для динамической генерации новых фрагментов текста.

1. Поясните понятие **Контекст шаблона (template context).**

Контекст шаблона (template context) - это набор данных, доступных в момент рендеринга шаблона, которые могут использоваться для формирования вывода. Эти данные могут включать переменные, функции, объекты и другие типы данных.

1. Поясните понятие **Макет (layout).**

Макет (layout) - это шаблон, определяющий структуру и расположение элементов на веб-странице. Он определяет, где будут расположены заголовки, навигационные меню, контент и другие элементы веб-страницы. Основная идея использования макета состоит в том, чтобы разделить разметку и стилизацию веб-страницы от ее содержимого. Это позволяет легко менять внешний вид веб-страницы без изменения ее содержимого.

1. Поясните понятие **Частичное представление (partial-view).**

Элементы представлений для вставки в шаблон

1. Поясните понятие **Хелпер** (**helper)**.

это функция JavaScript, которая используется для создания дополнительной логики в шаблоне Handlebars. Она позволяет обрабатывать данные и возвращать разметку для вставки в шаблон. Хелперы позволяют создавать более гибкие и функциональные шаблоны Handlebars, и делают процесс создания шаблонов более эффективным и удобным для использования.

1. Перечислите встроенные хелперы Handelbars.
2. #if, #unless, #each, #with, #lookup, #log
3. Поясните понятие **Статический файл**.

файлы, расположенные на стороне сервера и предназначенные для считывания их без изменения с помощью HTTP GET-запроса по имени ресурса, включающего имя файла.

1. Поясните назначение http-заголовков:

**Cache-control** используется для задания инструкций кэширования как для запросов, так и для ответов.

**ETag** (или entity tag) — один из механизмов кэширования в HTTP. По сути это идентификатор, который присваивается файлу сервером для последующей проверки

**Last-Modified** в ответе HTTP содержит дату и время, в которую, по мнению удаленного сервера, запрашиваемый ресурс был изменен.

***If***-**Modified-Since** делает запрос условным: сервер отправит обратно запрошенный ресурс с статусом [200](https://developer.mozilla.org/ru/docs/Web/HTTP/Status/200), только если он был изменен после указанной даты. Если запрос не был изменен после указанной даты, ответ будет [304](https://developer.mozilla.org/ru/docs/Web/HTTP/Status/304) без какого-либо тела; заголовок Last-Modified при этом будет содержать дату последней модификации.

**If-None-Match** Для методов [GET](https://developer.mozilla.org/ru/docs/Web/HTTP/Methods/GET) и [HEAD](https://developer.mozilla.org/ru/docs/Web/HTTP/Methods/HEAD) сервер отправляет запрошенный ресурс только в том случае, если он соответствует одному из перечисленных **ETags**. Для [PUT](https://developer.mozilla.org/ru/docs/Web/HTTP/Methods/PUT) и других небезопасных методов он будет загружать только ресурс в этом случае.

**Expires** HTTP-заголовок Expires устанавливает дату истечения срока действия (expiration date) ресурса, который был запрошен клиентом, после которой клиент должен перезапросить этот ресурс с сервера.

Когда клиент запрашивает ресурс у сервера, сервер возвращает этот ресурс с заголовком Expires, указывающим дату истечения срока действия ресурса.

Если срок действия ресурса истек, то клиент должен запросить ресурс снова у сервера, чтобы получить обновленную версию. Если на сервере нет обновленной версии ресурса, то сервер может вернуть тот же самый ресурс, что и ранее, но с новым значением заголовка Expires.

1. Поясните свойство **\_\_dirname** объекта **global**.

Представляет собой строковое значение, содержащее полный путь к директории, в которой расположен текущий исполняемый скрипт.

Это свойство очень полезно при работе с файлами и папками в Node.js, поскольку оно позволяет получать абсолютный путь к текущей директории независимо от того, где находится скрипт.

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

Отправляет клиенту файл для скачивания

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

Передает вложенный файл

1. Поясните назначение http-заголовков: **Content-Disposition: attachment**

HTTP-заголовок Content-Disposition используется для управления тем, как браузер должен обрабатывать контент, который возвращается в ответ на запрос. Значение "attachment" для Content-Disposition указывает на то, что сервер должен представить ответ как файл, который должен быть загружен или сохранен на компьютере пользователя.

Если сервер отправляет HTTP-ответ с заголовком Content-Disposition со значением "attachment", то браузер обычно предлагает пользователю сохранить файл с заданным именем.

Content-Disposition может также использоваться с другими значениями, например, "inline", что указывает браузеру отобразить ответ в браузере вместо загрузки файла. Однако, когда используется значение "attachment", браузер обычно не пытается отображать контент и вместо этого сохраняет его как файл.