1. **Вопрос:** Что представляет собой веб-разработка? **Ответ:** Веб-разработка — это процесс создания веб-сайтов или веб-приложений, включая их дизайн, программирование и поддержку.
2. **Вопрос:** Классификация ЭВМ: по принципу действия, по поколениям, назначению, по размерам и функциональным возможностям. **Ответ:** ЭВМ классифицируются по принципу действия (аналоговые, цифровые), по поколениям (первое, второе, третье и т.д.), по назначению (универсальные, специализированные), по размерам (суперкомпьютеры, мейнфреймы, мини-компьютеры, микрокомпьютеры) и по функциональным возможностям.
3. **Вопрос:** Какие языки программирования чаще всего используются в веб-разработке? **Ответ:** В веб-разработке чаще всего используются языки программирования, такие как HTML, CSS, JavaScript, а также серверные языки, например, PHP, Python, Ruby и другие.
4. **Вопрос:** Что такое HTML? **Ответ:** HTML (HyperText Markup Language) — это язык разметки, используемый для создания структуры веб-страниц с помощью различных элементов и тегов.
5. **Вопрос:** Что такое CSS? **Ответ:** CSS (Cascading Style Sheets) — это язык стилей, который определяет внешний вид и форматирование элементов HTML на веб-странице.
6. **Вопрос:** Что такое JavaScript? **Ответ:** JavaScript — это язык программирования, который используется для создания интерактивных элементов на веб-страницах, обеспечивая клиентскую сторону веб-приложений.
7. **Вопрос:** Что такое RESTful API? **Ответ:** RESTful API (Representational State Transfer) — это структурированный подход к созданию веб-сервисов, использующий стандартные протоколы HTTP, URI и форматы данных, такие как JSON или XML.
8. **Вопрос:** Что такое JSON и для чего он используется в веб-разработке? **Ответ:** JSON (JavaScript Object Notation) — это формат обмена данными, основанный на синтаксисе объектов JavaScript. Он часто используется в веб-разработке для передачи данных между клиентом и сервером.
9. **Вопрос:** Что представляет собой серверное приложение? **Ответ:** Серверное приложение — это программное обеспечение, выполняемое на сервере, которое обрабатывает запросы от клиентов и управляет ресурсами сервера.
10. **Вопрос:** Каким образом серверное приложение обрабатывает запросы от клиентов? **Ответ:** Серверное приложение обрабатывает запросы от клиентов, выполняя необходимые операции, взаимодействуя с базой данных (если необходимо) и отправляя обратно клиенту результаты выполнения запроса. Обычно это осуществляется через протоколы связи, такие как HTTP.
11. **Вопрос:** Какие языки программирования чаще всего используются для разработки серверных приложений? **Ответ:** Для разработки серверных приложений часто используются языки программирования, такие как Java, Python, Ruby, PHP, Node.js (JavaScript), C#, и другие.
12. **Вопрос:** Что такое API (интерфейс программирования приложений) на сервере? **Ответ:** API на сервере представляет собой набор методов и функций, которые позволяют взаимодействовать с серверным приложением, отправлять запросы и получать ответы.
13. **Вопрос:** Какие протоколы используются для обмена данными между серверным и клиентским приложениями? **Ответ:** Для обмена данными между серверным и клиентским приложениями часто используются протоколы HTTP и HTTPS, а также веб-сокеты для реализации двусторонней связи.
14. **Вопрос:** Что такое база данных на сервере? **Ответ:** База данных на сервере — это система хранения и управления данными, которая располагается на сервере и обеспечивает доступ к данным для серверных приложений.
15. **Вопрос:** Каким образом серверные приложения обрабатывают асинхронные запросы? **Ответ:** Серверные приложения могут использовать асинхронные методы обработки запросов, такие как многозадачность, обратные вызовы или промисы, чтобы эффективно обрабатывать большое количество запросов одновременно без блокировки.
16. **Вопрос:** Что такое серверная масштабируемость? **Ответ:** Серверная масштабируемость означает способность серверной системы эффективно обрабатывать рост количества запросов, обеспечивая стабильную производительность при увеличении нагрузки.
17. **Вопрос:** Какие принципы RESTful архитектуры? **Ответ:** Принципы RESTful архитектуры включают в себя использование стандартных методов HTTP (GET, POST, PUT, DELETE), ориентацию на ресурсы (URI представляют ресурсы), предоставление недостаточного, но достаточного для понимания клиента, состояния ресурсов.
18. **Вопрос:** Каким образом клиент и сервер обмениваются данными в веб-приложении? **Ответ:** Обмен данными между клиентом и сервером в веб-приложении происходит через отправку HTTP-запросов от клиента к серверу и получение HTTP-ответов с необходимыми данными.
19. **Вопрос:** Что такое CORS (Cross-Origin Resource Sharing)? **Ответ:** CORS — это механизм, позволяющий определенному серверу разрешать или запрещать запросы ресурсов с другого домена в браузере. Это средство обеспечивает безопасность и защиту от запросов с других источников.
20. **Вопрос:** Что такое база данных? **Ответ:** База данных — это структурированное хранилище данных, организованное для эффективного добавления, удаления и обновления информации. Она может быть использована для хранения данных различных типов и форматов. **Вопрос:** Какие типы баз данных существуют? **Ответ:** Существует несколько типов баз данных, включая реляционные (SQL), NoSQL (например, MongoDB, Cassandra), объектно-ориентированные, временные, иерархические и др.
21. **Вопрос:** Что такое SQL? **Ответ:** SQL (Structured Query Language) — это язык программирования и запросов, используемый для управления и манипулирования данными в реляционных базах данных.
22. **Вопрос:** Какие основные операции поддерживает SQL? **Ответ:** Основные операции в SQL включают SELECT (выбор данных), INSERT (вставка данных), UPDATE (обновление данных), DELETE (удаление данных) и другие операции для управления структурой базы данных.
23. **Вопрос:** Что представляет собой таблица в реляционной базе данных? **Ответ:** Таблица в реляционной базе данных представляет собой структурированный набор данных, организованных в виде строк и столбцов, где каждая строка представляет собой запись, а каждый столбец — атрибут.
24. **Вопрос:** Какие ключи используются для связывания таблиц в реляционной базе данных? **Ответ:** Для связывания таблиц в реляционной базе данных используются внешние ключи, которые связываются с первичными ключами в других таблицах.
25. **Вопрос:** Что такое запрос SELECT в SQL? **Ответ:** Запрос SELECT в SQL используется для выбора данных из одной или нескольких таблиц базы данных. Он может включать условия, сортировку и другие параметры для получения конкретных результатов.
26. **Вопрос:** Какие типы данных могут использоваться в столбцах таблицы? **Ответ:** Типы данных включают целые числа, числа с плавающей точкой, строки (текст), даты, времена, булевы значения и другие, в зависимости от спецификации конкретной реляционной базы данных.
27. **Вопрос:** Каким образом предотвратить дублирование данных в реляционной базе данных? **Ответ:** Дублирование данных предотвращается путем использования уникальных ключей и ограничений целостности данных, таких как PRIMARY KEY и UNIQUE constraints.
28. **Вопрос:** Что такое транзакция в контексте баз данных? **Ответ:** Транзакция в базе данных представляет собой логическую операцию, состоящую из одного или нескольких SQL-запросов, которые выполняются как единое целое. Транзакции обеспечивают атомарность, согласованность, изолированность и долговечность данных (свойства ACID).
29. **Вопрос:** Каким образом выполняется объединение таблиц в SQL? **Ответ:** Объединение таблиц в SQL выполняется с использованием операторов JOIN. Например, INNER JOIN, LEFT JOIN, RIGHT JOIN и FULL JOIN используются для объединения данных из разных таблиц на основе определенных условий. **Вопрос:** Какие типы соединений (joins) существуют в SQL, и как они работают? **Ответ:** Существует несколько типов соединений в SQL, включая INNER JOIN (возвращает строки, имеющие совпадения в обеих таблицах), LEFT JOIN (возвращает все строки из левой таблицы и соответствующие строки из правой), RIGHT JOIN (аналогично LEFT JOIN, но для правой таблицы), и FULL JOIN (возвращает все строки, имеющие совпадения в любой из таблиц).
30. **Вопрос:** Каким образом можно добавить новую таблицу в реляционную базу данных? **Ответ:** Новую таблицу можно добавить в реляционную базу данных с помощью SQL-запроса CREATE TABLE, указав необходимые столбцы и их типы данных.
31. **Вопрос:** Какое ключевое слово используется для объявления переменной в JavaScript? **Ответ:** Ключевое слово var, let или const используется для объявления переменной в JavaScript.
32. **Вопрос:** Какой тип данных представляет собой массив в JavaScript? **Ответ:** Массив в JavaScript представляет собой объект, и его тип данных можно определить с использованием оператора typeof.
33. **Вопрос:** Каким образом можно вывести текст в консоль в JavaScript? **Ответ:** Для вывода текста в консоль в JavaScript используется метод console.log().
34. **Вопрос:** Что такое оператор "===" в JavaScript? **Ответ:** Оператор "===" в JavaScript используется для строгого сравнения значений и типов данных. Он возвращает true, если значения и типы данных совпадают.
35. **Вопрос:** Какой метод используется для добавления элемента в конец массива в JavaScript? **Ответ:** Метод push() используется для добавления элемента в конец массива в JavaScript.
36. **Вопрос:** Каким образом можно создать функцию в JavaScript? **Ответ:** Функцию можно создать с использованием ключевого слова function, например:

function myFunction() {

// код функции

}

**Вопрос:** Каким образом можно добавить обработчик события в JavaScript? **Ответ:** Обработчик события можно добавить с использованием метода addEventListener() или присваиванием функции к свойству события, например:

element.addEventListener('click', myFunction);

1. **Вопрос:** Каким образом можно выполнить запрос к серверу в JavaScript? **Ответ:** Для выполнения запроса к серверу в JavaScript используется объект XMLHttpRequest или современный API fetch().
2. **Вопрос:** Каким ключевым словом объявляется класс в JavaScript? **Ответ:** Класс объявляется с использованием ключевого слова class, например:

class MyClass {

// тело класса

}

**Вопрос:** Что такое стрелочные функции (arrow functions) в JavaScript? **Ответ:** Стрелочные функции — это сокращенный синтаксис для объявления функций в JavaScript, например:

const add = (a, b) => a + b;

1. **Вопрос:** Каким образом можно прервать выполнение цикла в JavaScript? **Ответ:** Выполнение цикла можно прервать с использованием оператора break.
2. **Вопрос:** Что такое localStorage в JavaScript? **Ответ:** localStorage — это механизм веб-хранилища в браузере, позволяющий хранить данные в виде пар ключ-значение на стороне клиента.
3. **Вопрос:** Каким образом можно преобразовать строку в число в JavaScript? **Ответ:** Для преобразования строки в число можно использовать функции parseInt() или parseFloat(), либо оператор Number().
4. **Вопрос:** Что такое fetch() в JavaScript? **Ответ:** Метод fetch() в JavaScript используется для отправки запросов к серверу и получения ответов. Он возвращает объект Promise, что облегчает работу с асинхронными запросами.
5. **Вопрос:** Что такое cookie в JavaScript? **Ответ:** Cookies — это небольшие текстовые файлы, хранящиеся на компьютере пользователя, которые используются для отслеживания информации о посещенных сайтах.

**Вопрос:** Каким образом можно скрыть элемент на веб-странице с использованием JavaScript? **Ответ:** С использованием свойства style.display и присвоения ему значения "none" можно скрыть элемент в JavaScript, например:

document.getElementById('myElement').style.display = 'none';