# Performance Review. Сентябрь 2020

## Список вопросов JS:

|  |  |  |  |
| --- | --- | --- | --- |
| Номер | Компетенция | Навык | Готово |
|  | ООП (знание парадигмы программирования) | Что такое Объект, класс, абстрактный класс и интерфейс? | ✓ |
|  | ООП (знание парадигмы программирования) | Что такое реализация и наследование | ✓ |
|  | ООП (знание парадигмы программирования) | Трейты - что такое? | ✓ |
|  | ООП (знание парадигмы программирования) | Абстракция | ✓ |
|  | ООП (знание парадигмы программирования) | Инкапсуляция | ✓ |
|  | ООП (знание парадигмы программирования) | Полиморфизм | ✓ |
|  | ООП (знание парадигмы программирования) | Где сильнее связанность при наследовании или инкапсуляции? | ✓ |
|  | ООП (знание парадигмы программирования) | Что такое прототип? | ✓ |
|  | ООП (знание парадигмы программирования) | function getSum(){return this.sum}. От куда возьмётся this? Как его можно поменять? В каких случаях this не определён? | ✓ |
|  | Объектно-ориентированный дизайн (SOLID) | Показать как минимум 2 примера использования принципа единой ответсвенности в своём коде, который был написан в период с 01.07.2020 по настоящее время |  |
|  | Объектно-ориентированный дизайн (SOLID) | Показать как минимум 2 примера использования принципа открытости/закрытости в своём коде, который был написан в период с 01.07.2020 по настоящее время, доказать, что принцип соблюдается т.е. он пригодился хотя бы 1 раз |  |
|  | Объектно-ориентированный дизайн (SOLID) | Показать как минимум 2 примера использования принципа подстановки Барбары Лисков, который был написан в период с 01.07.2020 по настоящее время, обсоновать, что он уже сейчас работает. |  |
|  | Объектно-ориентированный дизайн (SOLID) | Показать как минимум 2 примера использования принципа разделения интефейсов, которые были написаны в период с 01.07.2020 по настоящее время. |  |
|  | Объектно-ориентированный дизайн (SOLID) | Показать как минимум 2 примера использования принципа инверсии зависимостей, которые были написаны в период с 01.07.2020 по настоящее время. Кто является заказчиком данных частей, почему мы считаем это принципом единой отвесвенности. |  |
|  | Объектно-ориентированный дизайн (SOLID) | Когда необходимо применять принцип обратной инверсии зависмости? |  |
|  | Знание HTML | Необходимо сверстать страницу для не видящих людей. Как это сделать? |  |
|  | Знание HTML | Необходимо сверстать страницу как для видящих, так для не видящих. Как это сделать? |  |
|  | Знание HTML | Сверстать флаг российской федерации во всё окно браузера с равными пропорциями. | ✓ |
|  | Знание CSS | Сверстать флаг российской федерации во всё окно браузера с равными пропорциями. | ✓ |
|  | Знание CSS | Есть ли возможность изменить цвет svg картинки? | ✓ |
|  | Знание CSS | Как в css можно анимировать переход от одного состояния в другое? |  |
|  | Знание CSS | Как распределить элементы равной ширины, чтобы между ними было равное растояния, зависящее от ширины родительского блока? | ✓ |
|  | Знание JS | Сделайте POST запрос к серверу из браузера не используя библиотеки? | ✓ |
|  | Знание JS | Какие проблемы могут возникнуть при запросе другой домен? | ✓ |
|  | Знание JS | Только неоптимальные запросы могут быть проблемой производительности БД? | ✓ |
|  | Знание JS | Видны ли переменные var в замыкании? |  |
|  | Знание алгоритмов и структур данных | Есть 2-х мерный массив 3 на 3 каждый элемент - пустая строка, символ '0' или символ 'X', определить является ли данная позиция выйгрышем в крестики нолики? Кто выйграл, если позиция игра выйграна? Ошибка, если выйгрыш 2 раза и более. | ✓ |
|  | Декомпозиция задач | Как можно разбить большую задачу, чтобы заказчик мог проверять степень её выполнения? |  |
|  | Декомпозиция задач | Нужен визуальный редактор диаграммы ганта. Какие вопросы надо задать и какие подзадачи можно выделить? |  |
|  | Умеет самостоятельно проводить код-review | Тестовое задание | ✓ |
|  | Знание Git | Какими коммандами отправить изменения, которые вы только что внесли, в репозиторий? | ✓ |
|  | Знание Git | Какими командами можно получить изменения в текущей ветки из репозитория | ✓ |
|  | Знание Git | Как переключиться на ветку 'test', которая есть в репозитории origin, но нет локально? | ✓ |
|  | Знание Git | Вы сделали коммит не в ту ветку, в которую хотели сделать коммит. Как переместить коммит в другую ветку? | ✓ |
|  | Знание Git | Как отменить коммит? | ✓ |
|  | Знание unit - тестов | Примеры написанных тестов. |  |
|  | Знание unit - тестов | Примеры тестов с подменой зависимостей. |  |
|  | Знание технического английского | [https://github.com/strongloop/strong-soap Как можно получить soap-заголовки?](https://github.com/timiki/rpc-server-bundle) |  |
|  | Знание React | Написать компонент для галочки, галочка должна выглядеть одинаково в разных местах. Компонент должен метод, который будте вызван при изменении состояния. | ✓ |
|  | Декомпозиция больших задач, фич, их оценка и распределение по разработчикам, разработка архитектуры | Компания по производству мороженного в Екатеринбурге решила открыть свой интернет магазин для розницы. Какие вопросы необходимо решить для начала разработки? |  |
|  | Знание сторонних библиотек | Какие библиотеки/фреймворки знаете? |  |
|  | Анализ сбоев | Какие случаи были за последнии полгода. Приложите задачу. |  |
|  | Улучшение действующей архитектуры кода | Примеры. Приложите ссылку на коммит. |  |
|  | Изучение новых библиотек | Какую бибилотеку вы ппробовали изучить за последнии 3 месяца? |  |
|  | Модули | Что такое принцип эквивалентности повторгного использования и выпусков? |  |
|  | Модули | Что такое принцип согласованного изменения? |  |
|  | Модули | Что такое принцип совместного повторного использования? |  |
|  | Модули | Противоречать ли принциы "эквивалентности повторгного использования и выпусков", "принцип согласованного изменения" и "совместного повторного использования" друг другу? |  |
|  | Качество кода | и 5 коммитов которыми гордитесь, если нет таких, то за которые не стыдно. |  |

## Что такое Объект, класс, абстрактный класс и интерфейс?

Объект – это тип данных, который используется для хранения коллекций различных значений и сложных сущностей.

Класс – это сущность, описывающая какой-либо абстрактный тип данных.

В JavaScript класс – это разновидность функции.

Абстрактный класс – класс, определяющий абстрактные методы, от которого будут наследоваться классы, а не создаваться экземпляры.

Интерфейс – контракт класса реализовывать определенные методы.

Интерфейс – внутренний и внешний, методы и свойства класса.

## Что такое реализация и наследование.

Реализация интерфейса – полностью определенный класс.

Наследование – концепция при которой абстрактный класс может наследовать методы и свойства другого класса.

## Трейты - что такое?

Трейт – механизм позволяющий группировать методы и наследовать их.

## Абстракция.

Абстракция — это объявление только тех методов и свойств которые необходимы в исследуемой области.

## Инкапсуляция.

Инкапсуляция — это размещение в одном месте свойств и методов управления ими, а также защита их от использования извне.

## Полиморфизм.

Полиморфизм – способность метода работать с разными типами данных

## Где сильнее связанность при наследовании или инкапсуляции?

Связанность сильнее при наследовании исходя из определения инкапсуляции. При наследовании компоненты (классы) связаны и зависят друг от друга в этом случае сильнее.

## Что такое прототип?

Экземпляр объекта, который скопировали, для наследования методов и свойств.

## function getSum(){return this.sum}. От куда возьмётся this? Как его можно поменять? В каких случаях this не определён?

this это ссылка на объект который вызывает getSum().

Если функция была вызвана без объекта, то this== undefined

## Показать, как минимум 2 примера использования принципа единой ответственности в своём коде, который был написан в период с 01.07.2020 по настоящее время.

## Показать, как минимум 2 примера использования принципа открытости/закрытости в своём коде, который был написан в период с 01.07.2020 по настоящее время, доказать, что принцип соблюдается т.е. он пригодился хотя бы 1 раз.

## Показать, как минимум 2 примера использования принципа подстановки Барбары Лисков, который был написан в период с 01.07.2020 по настоящее время, обосновать, что он уже сейчас работает.

## Показать, как минимум 2 примера использования принципа разделения интерфейсов, которые были написаны в период с 01.07.2020 по настоящее время.

## Показать, как минимум 2 примера использования принципа инверсии зависимостей, которые были написаны в период с 01.07.2020 по настоящее время. Кто является заказчиком данных частей, почему мы считаем это принципом единой ответственности.

## Когда необходимо применять принцип обратной инверсии зависимости?

## Необходимо сверстать страницу для не видящих людей. Как это сделать?

## Необходимо сверстать страницу как для видящих, так для не видящих. Как это сделать?

## Сверстать флаг российской федерации во всё окно браузера с равными пропорциями.

git

## Сверстать флаг российской федерации во всё окно браузера с равными пропорциями.

git

## Есть ли возможность изменить цвет svg картинки?

git

## Как в css можно анимировать переход от одного состояния в другое?

## Как распределить элементы равной ширины, чтобы между ними было равное растояния, зависящее от ширины родительского блока?

У родительского блока задать свойство

Displyay: flex;

Justify-content:space-between

## Сделайте POST запрос к серверу из браузера не используя библиотеки?

В js сделать http запросы можно двумя способами.

1 способ

XMLHttpRequest – это встроенный в браузер объект, который даёт возможность делать HTTP-запросы к серверу без перезагрузки страницы.

let xhr = new XMLHttpRequest();

xhr.open(method, URL, [async, user, password])

xhr.send([body])

2 способ

C Помощью Функции Fetch

let promise = fetch(url, [options])

let user = {

name: 'John',

surname: 'Smith'

};

let response = await fetch('/article/fetch/post/user', {

method: 'POST',

headers: {

'Content-Type': 'application/json;charset=utf-8'

},

body: JSON.stringify(user)

});

let result = await response.json();

alert(result.message);

## Какие проблемы могут возникнуть при запросе другой домен?

При запросе на другой домен может сработать политика CORS при которой запрос не будет обработан

Cross-Origin Resource Sharing (CORS) — механизм, использующий дополнительные HTTP-заголовки, чтобы дать возможность агенту пользователя получать разрешения на доступ к выбранным ресурсам с сервера на источнике (домене), отличном от того, что сайт использует в данный момент.

Если в ресурсе на который посылается cross-origin HTTP request, политикой безопасности определено что не все могут посылать кросс доменные запросы, то ответ будет отправлен обратно.

Для этого в заголовке ответа устанавливаются определенные парамтеры.

## Только неоптимальные запросы могут быть проблемой производительности БД?

Нет, так же БД должна удовлетворять 3 нормам построения.

Атомарность, отсутствие зависимости не ключевых полей от части составного ключа. отсутствие зависимостии не ключевых полей от других не ключевых полей

## Видны ли переменные var в замыкании?

## Есть 2-х мерный массив 3 на 3 каждый элемент - пустая строка, символ '0' или символ 'X', определить является ли данная позиция выйгрышем в крестики нолики? Кто выйграл, если позиция игра выйграна? Ошибка, если выйгрыш 2 раза и более.

Git

## Как можно разбить большую задачу, чтобы заказчик мог проверять степень её выполнения?

## Нужен визуальный редактор диаграммы ганта. Какие вопросы надо задать и какие подзадачи можно выделить?

## Тестовое задание

Умеет ли сам проводить code review. Умею

## Какими коммандами отправить изменения, которые вы только что внесли, в репозиторий?

Git add .

Git commit –m”сообщение коммита”

Git push «удаленный репозиторий» «ветка»

## Какими командами можно получить изменения в текущей ветки из репозитория

Git pull

Git fetch +git merge

## Как переключиться на ветку 'test', которая есть в репозитории origin, но нет локально?

Git pull

Git checkout test

## Вы сделали коммит не в ту ветку, в которую хотели сделать коммит. Как переместить коммит в другую ветку?

Git checkout «желаемая ветка»

Git cherry-pick « ветка куда влили»

## Как отменить коммит?

1 способ

Изменить файлы и сделать коммит, который будет отменять предыдущие изменения

Git revert head

2 способ

git reset --hard v1

## Примеры написанных тестов.

## Примеры тестов с подменой зависимостей.

## https://github.com/strongloop/strong-soap Как можно получить soap-заголовки?

## Написать компонент для галочки, галочка должна выглядеть одинаково в разных местах. Компонент должен метод, который будте вызван при изменении состояния.

## Компания по производству мороженного в Екатеринбурге решила открыть свой интернет магазин для розницы. Какие вопросы необходимо решить для начала разработки?

## Какие библиотеки/фреймворки знаете?

## Какие случаи были за последнии полгода. Приложите задачу.

## Примеры. Приложите ссылку на коммит.

## Какую бибилотеку вы пробовали изучить за последнии 3 месяца?

## Что такое принцип эквивалентности повторгного использования и выпусков?

## Что такое принцип согласованного изменения?

## Что такое принцип совместного повторного использования?

## Противоречать ли принциы "эквивалентности повторгного использования и выпусков", "принцип согласованного изменения" и "совместного повторного использования" друг другу?

## и 5 коммитов которыми гордитесь, если нет таких, то за которые не стыдно.

## Сделайте POST запрос к серверу из браузера не используя библиотеки?

<https://learn.javascript.ru/xmlhttprequest>

<https://learn.javascript.ru/fetch>

## Видны ли переменные var в замыкании?

<https://learn.javascript.ru/closures>

<https://developer.mozilla.org/ru/docs/Web/JavaScript/Closures>