

JS

1. Сколько типов данных в JS, перечислите их.
2. В чем разница между null и undefined?
3. В чем разница между операторами "==" и "==="?
4. Зададим две переменные

```
let a = {a: 1};  
let b = {a: 1};
```

Какое логическое значение будет иметь выражение a == b? Обоснуйте.
5. Что такое замыкание?
6. В чем разница между методами call, apply и bind?
7. Как создать функцию извлечения квадратного корня из числа без использования класса Math и оператора «**»?
8. В чем разница между ключевыми словами «var», «let» и «const»?
9. Что такое модули в JS?
10. Что делают атрибуты async и defer у тега script?
11. Что такое наследование, зачем оно нужно и как его использовать в ES6?
12. Какие объекты можно передавать в качестве ошибки команде throw?
13. В чём разница между приватными и защищенными полями в JS?
14. Что такое REPL?
15. Как мы можем наследовать свойства родительского конструктора с помощью синтаксиса функций?
16. Для чего предназначены методы setTimeout и setInterval?
17. Какой код нужно прописать в консоли, чтобы отключить все таймеры на странице?
18. Что такое анонимная функция?
19. Что такое стрелочные функции? Отличаются ли они чем-нибудь от обычных?
20. С помощью какого метода можно отсортировать массив? Как отсортировать массив в обратном порядке?
21. Что означает ключевое слово this? На какой объект оно указывает?
22. Что такое callback-и? Как их использовать?
23. Как получить функцию-конструктор, имея “на руках” только объект, который был ею создан?
24. Что такое геттеры и сеттеры? Зачем они нужны?
25. Что такое шаблонные литералы (Template Literals)?
26. Что такое прототипы? Какой смысл в JS они несут?

TS

1. Главная задача TypeScript?
2. Может ли TypeScript исполняться в браузере?
3. Объясните процесс понижения уровня кода.
4. Какие настройки предоставляет TypeScript для управления уровнем строгости проверок?
5. Перечислите базовые типы данных, используемые в TypeScript.

6. Каким образом в TypeScript аннотируются массивы?
7. Что в TypeScript называется кортежем?
8. Как аннотировать функции с помощью TypeScript? Всегда ли это необходимо?
9. Что в TypeScript называется объединением? Привести примеры использования.
10. Объяснить псевдонимы типов, интерфейсы и их различия.
11. Объяснить утверждения типов. Как производить некорректные утверждения типов?
12. Какие типы в TypeScript называются литеральными? Привести примеры использования.
13. Объяснить сужение типов. Способы сужения типов. Привести примеры.
14. Что называется сигнатурой вызова? Конструкторской сигнатурой?
15. Объяснить концепцию перегрузки функций в TypeScript. Привести пример.
16. Что в TypeScript называется объектным типом?
17. Как TypeScript подходит к сравнению типов?
18. Перечислите модификаторы свойств в TypeScript, назовите их особенности.
19. Объяснить сигнатуры индексов. Привести примеры использования.
20. Назовите особенности использования классов в TypeScript.
21. Объясните ключевые слова `extends` и `implements` в TypeScript. Назовите их различия.
22. Перечислите области видимости в TypeScript и назовите их особенности.
23. Объяснить концепцию обобщенных типов в TypeScript. Привести примеры.
24. Что в TypeScript называется ограничением обобщенного типа? Привести примеры.