**1) Откуда берётся контекст в функции?**  
У стрелочных функций нет своего контекста, берёт от родителя.  
Обычные функции имеют контекст того места, откуда она вызвана.  
  
**2) Чистая функция.**  
  
Функция без сайд эффектов, занимается своим внутренним состоянием.  
  
**3) Как мы можем изменить контекст у функции?**  
Bind, call, apply.  
  
Особенности bind.  
2 раза применить не получится, при первом разе жестко фиксируется контекст.  
  
**4) Что такое eval, почему его следует избегать?**  
Eval позволяет создать JS код на лету и использовать его.  
  
**5) Чем отличается область видимости от контекста?**  
Область видимости больше относится к переменным и к данным, а контекст к функциям, то, что оперирует этими данными.  
  
**6) Как работают модули в ES6?**  
**7) Можем ли мы поменять переменную вне своего модуля?**  
С объектами будет работать, а с примитивами нет.  
  
**8) Как можно эмулировать модули без них?**  
Что бы в одном файле не было проблем с одинаковыми названиями переменных.  
  
Как изолировать область видимости?  
- Блок использовать без function declaration.  
- Использовать замыкание.  
  
**9) Откуда берется свойство length у массива?**  
От прототипа есть свои свойства.  
  
Дескрипторы.  
  
**10) Event loop.**Бесконечно крутится, пока не появятся задачи.  
Как задача появляется, начинает работать callStack, задача синхронно отрабатывает, потом опять event loop крутится.  
  
Для того, чтобы не было блокировки потока ввода/вывода.  
  
**11) Как мы работаем с асинхронным кодом?**Можем через callback.  
  
  
**12) Shadow DOM.**  
Он придуман для изоляции стилей.  
  
Почему в инпуте не может обратиться к элементу?  
**13) Как мы будем структурировать html страницу?**  
  
**14) Можно ли сделать несколько H1 заголовков?**  
  
Можно, но плохо для SMM.