1. Концепции ООП (наследование, инкапсуляция, полиморфизм)
2. Модификаторы доступа (private, public, protected, internal)
3. Какой тип является базовым для всех остальных типов.
4. Различие между интерфейсом и абстрактным классом
5. Поддерживает ли C# множественное наследование?
6. Понятие "ссылочный тип" и «значимый тип»
7. Разница между структурой и классом
8. Делегаты (общие сведения)
9. Как работает сборщик мусора?
10. Finalize и Dispose
11. Для чего в .NET используется конструкция using(…){…}? Причем тут IDisposable?
12. Расскажите про интерфейс IEnumerable? Зачем он используется?
13. Когда мы можем пройтись по собственной коллекции foreach- ом? Что для этого надо сделать и почему?
14. Различие между IEnumerable and IQueryable?
15. Как работает Dictionary ?
16. Расскажите, как работает try, catch, finally? Когда вызывается каждый?
17. Чем отличаются друг от друга классы String и StringBuilder?
18. Что такое управляемый код и CLR?
19. Что такое Boxing и Unboxing?
20. Чем отличается event от delegate?
21. LINQ lazy loading ("ленивая загрузка"), eager loading ("жадная загрузка") в чем разница?
22. Как запретить наследоваться от своего класса?
23. Что такое MVC?
24. Каким образом можно присвоить значения полям, которые помечены ключевым словом readonly?
25. Чем отличаются константы и поля, доступные только для чтения?
26. Когда вызывается статический конструктор класса?
27. Что такое Entity Framework?
28. Какие подходы проектирования БД знаете?
29. Какие есть типы JOIN'ов? Кратко опишите каждый из типов.
30. Жизненный цикл приложения ASP.NET MVC (для себя)