1. Инкапсуляция . Назначение и примеры.
2. Обобщения, примеры обобщений в стандартной поставке .Net
3. Делегаты. Назначение, примеры, синтаксис
4. Шаблоны проектирования Классы шаблонов, примеры (не менее трех).
5. Назовите основные принципы ООП. Объясните их.
6. Модульное тестирование. Назначение и примеры
7. Исключения. Синтаксис, влияние на стек вызовов, особенности производительности.
8. Сборщик мусора (Garbage Collector) – назначение, методы работы, ограничения.
9. Исключения. Назначение, примеры, синтаксические конструкции.
10. REST. Аббревиатура, назначение, примеры.
11. Ссылочные и значимые типы. Классы и структуры - расположение в оперативной памяти, занимаемый объем.
12. Применение string builder и string. Различия и назначение.
13. LINQ. Аббревиатура, область применения.
14. Назначение и применение пространств имен (namespace)
15. Различие между абстрактным классом и интерфейсом. Наследование и композиция. Назначение интерфейсов.
16. Применение ключевых слов async / await Task Parallel Library. Fire &Forget.
17. наследование в c#. Назначение, примеры, возможность множественного наследования. Разница между наследованием класса и определением интерфейса.
18. Виды и назначение модификаторов доступа
19. Аббревиатура SOLID. Принципы, их назначение. Примеры.
20. Назначение и реализация паттерна Singleton. Его отношения с принципом единственной ответственности.
21. Управляемая память, куча и стек. Различия, принцип работы
22. Управление ресурсами в c#. Управляемые ресурсы, ключевое слово using.
23. Рефлексия
24. асинхронные методы. Отличие асинхронности от параллелизма.
25. Библиотека параллельных задач. Примеры использования. Средства обеспечения многопоточности.
26. полиморфизм, виды полиморфизма. Примеры.
27. Методы HTTP
28. Коллекции, IEnumerable, foreach.
29. Операторы, методы и атрибуты для ограничения параллелизма.
30. Определение ORM. Работа с Entity Framework. Code first, model first
31. Назначение dependency injection. Примеры применения
32. Шаблон проектирования «итератор» и его использование в foreach.
33. SOLID. Принцип I.
34. Назначение enum flags.
35. Способ реализации плагинной архитектуры приложения на c#
36. Расскажите о domain-driven-development
37. Структуры данных. Отличие между массивом (T [ ]) и списком (List )
38. Статические классы. Участие в наследовании и композиции, отличия от абстрактных классов и интерфейсов.
39. Понятие класса в объектно-ориентированном программировании. Синтаксис, пример определения. Члены класса.
40. Переопределение виртуальных методов. Ключевые слова, пример применения.
41. Абстрактные классы. Назначение, примеры, отличия от статических классов и интерфейсов.
42. MVC. Основные термины.
43. Абстрактные классы и интерфейсы. Назначение, участие в наследовании. Могут ли абстрактные классы иметь реализацию методов. Возможность частичной реализации.
44. MVVM. Основные термины.
45. Порождающие паттерны. Назначение и примеры.
46. Назначение и пример шаблона проектирования Стратегия.
47. Поведенческие паттерны проектирования. Назначение, примеры
48. Назначение и пример шаблона проектирования Строитель.
49. Архитектура систем. Монолит, сервисы, микросервисы
50. Назначение и пример шаблона проектирования фабрика.
51. . MVC. Понятие контроллера
52. MVVM. Понятие представления
53. MVC. Понятие модели
54. MVVM. Понятие модели представления.
55. MVC. Понятие представления, синтаксис Razor
56. MVVM. Понятие модели.
57. Boxing, unboxing. Система типов в C#
58. Атрибуты в C#. Определение, применение.