1. Для чего используют статические классы?

Статические классы объявляются с модификатором static и могут содержать только статические поля, свойства и методы По сути же, static классы - это классы содержащие глобальные переменные. static поля в публичных классах содержат неизменные данные для вообще всех экземпляров класса.

[Статический](https://learn.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/static) класс в основном такой же, как и нестатический класс, но имеется одно отличие: нельзя создавать экземпляры статического класса.

1. Что может содержать статический класс? 3

Статический класс может использоваться как обычный контейнер для наборов методов, статические поля и конструкторы статич????????????

. Что такое производный и базовый классы?

**Класс, члены которого наследуются, называется базовым классом.** **Класс, который наследует члены базового класса, называется производным классом**

1. Как используют ключевое слово base?

**Ключевое слово base используется** для доступа к членам базового из производного класса в следующих случаях: Вызов метода базового класса, который был **переопределен** другим методом. Определение конструктора базового класса, который должен вызываться при создании экземпляров производного класса

1. В чем заключена основная задача наследования?

Наследование (inheritance) является одним из ключевых моментов ООП. Благодаря наследованию один класс может унаследовать функциональность другого класса

1. Пусть базовый класс содержит метод basefunc(), а производный класс не имеет метода с таким именем. Может ли объект производного класса иметь доступ к методу basefunc()? Если да, то при каких условиях?

Да, всегда (у функц модификатор public)

7. Напишите объявление конструктора без аргументов для производного класса B, который будет вызывать конструктор без аргументов базового класса A.

8. Что такое полиморфизм? Приведите пример.



это свойство одних и тех же объектов и методов принимать разные формы.

9. Определите назначение виртуальных функций.

Те методы и свойства, которые мы хотим сделать доступными для переопределения, в базовом классе помечается модификатором **virtual**. Такие методы и свойства называют виртуальными. Итак, в основном, если в вашем классе-предке вы хотите определенного поведения для метода. Если ваш потомок использует тот же метод, но имеет другую реализацию, вы можете **переопределить** его, если у него есть ключевое слово **virtual** .

***Виртуальным*** называется такой метод, который объявляется как **virtual** в базовом классе. Виртуальный метод отличается тем, что он может быть переопределен в одном или нескольких производных классах. Следовательно, у каждого производного класса может быть свой вариант виртуального метода. Кроме того, виртуальные методы интересны тем, что именно происходит при их вызове по ссылке на базовый класс.

10. Кому доступны переменные с модификатором protected?

Любому классу-наследнику.

**Protected** — доступ открыт классам, производным от данного.

11. Наследуются ли переменные с модификатором private?

Да, но они не являются доступными.

12. As, is – что это, как применяется? В чем между ними отличие ?

Операторы приведения типов.  
If(o is Employee){  
Employee e = (Employee) o;  
}  
Проверка типа осущ., 2-ды. По этому в CLR  
Employee e = o as Employee;  
If(e != null){  
}

13. Поддерживает ли C# множественное наследование?

С# поддерживает множественное наследование в виде наследования от класса и нескольких интерфейсов, или просто от нескольких интерфейсов.  
Но не поддерживает наследование от нескольких классов

14. Можно ли запретить наследование от класса?

С помощью ключевого слова "sealed" **можно** **запретить** создавать наследников **от** **класса**.

15. Можно ли разрешить наследование класса, но запретить перекрытие метода?

Да. Указываем класс как public, а метод как sealed.

16. Что такое абстрактный класс?

Абстрактный класс в объектно-ориентированном программировании — **базовый класс, который не предполагает создания экземпляров**. Абстрактные классы реализуют на практике один из принципов ООП — полиморфизм. Абстрактный класс может содержать (и не содержать) абстрактные методы и свойства.

17. В каком случае вы обязаны объявить класс абстрактным?

1. В том случае, если класс является наследником абстрактного класса, но не все методы базового класса перекрыты и имеют реализацию.  
2. В том случае, если хотя бы один метод класса является абстрактным.

18. В чем разница между абстрактными и виртуальными классами? Между виртуальными и абстрактными методами?

Абстрактный класс это класс, содержащий хотя бы один метод (abstract)..  
Виртуальный метод имеет реализацию и м.б. переопределен в производном классе. Абстрактный метод не имеет реализацию, только описание метода, который д.б. реализован в производных классах.

19. Какие компоненты класса могут быть виртуальными?

Виртуальными могут быть:

* Методы
* Свойства
* Индексаторы
* События

20. Что такое интерфейс?

Интерфейсы, как и классы, определяют набор свойств, методов и событий. Но, в отличие от классов, они не содержат их реализации. Интерфейсы реализуются классами и определяются как самостоятельные сущности.

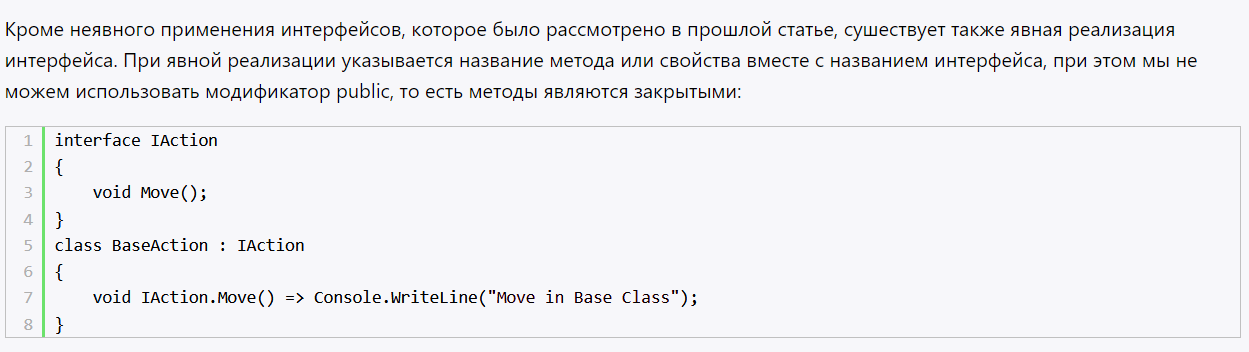
21. Что может содержать интерфейс?

* Методы
* Свойства
* Индексаторы
* События
* Статические поля и константы (начиная с версии C# 8.0)

22. Как работать с объектом через унаследованный интерфейс?



23. Приведите пример явной реализации интерфейса.



24. Почему нельзя указать модификатор видимости для методов интерфейса?

**Потому что все они должны иметь модификатор public, который и установлен по умолчанию**.

25. Можно ли наследовать от нескольких интерфейсов?

Один интерфейс может наследовать другой. Синтаксис наследования интерфейсов такой же, как и у классов. Когда в классе реализуется один интерфейс, наследующий другой, в нем должны быть реализованы все члены, определенные в цепочке наследования интерфейсов.

26. Назовите отличия между интерфейсом и абстрактным классом.

**1.Интерфейс описывает только поведение. У него нет состояния. А у абстрактного класса состояние есть: он описывает и то, и другое**

2. **Классы могут реализовывать сколько угодно интерфейсов, но наследоваться можно только от одного класса.**

3. **Абстрактный класс связывает между собой и объединяет классы, имеющие очень близкую связь. В то же время, один и тот же интерфейс могут реализовать классы, у которых вообще нет ничего общего.**

Абстрактный класс — это класс, у которого не реализован один или больше методов (некоторые языки требуют такие методы помечать специальными ключевыми словами).

Интерфейс — это абстрактный класс, у которого **ни один** метод не реализован, все они публичные и нет переменных класса.

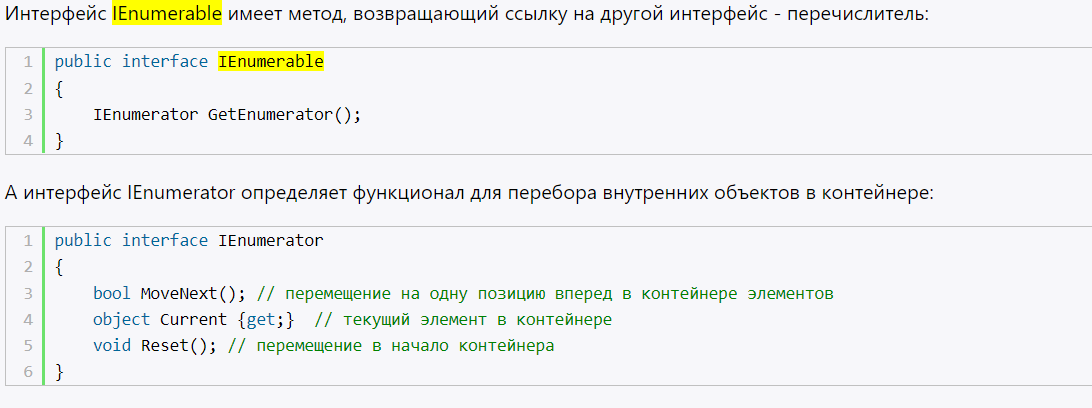
27. Для чего используются стандартные интерфейсы ICloneable, IComparable, IComparer, lEnumerable?

интерфейс IComparable задает метод сравнения объектов по принципу больше или меньше, что позволяет выполнять их сортировку

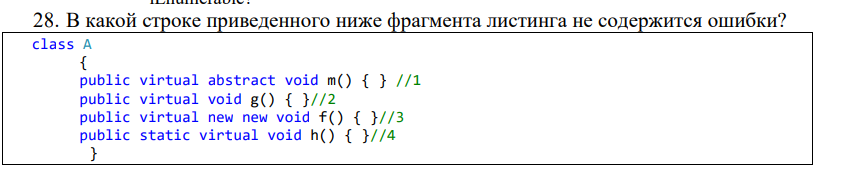
Интерфейс IComparable определен в пространстве имен System. Он содержит всего один метод CompareTo, возвращающий результат сравнения двух объектов — текущего и переданного ему в качестве параметра

ICloneable — клонировать объекты

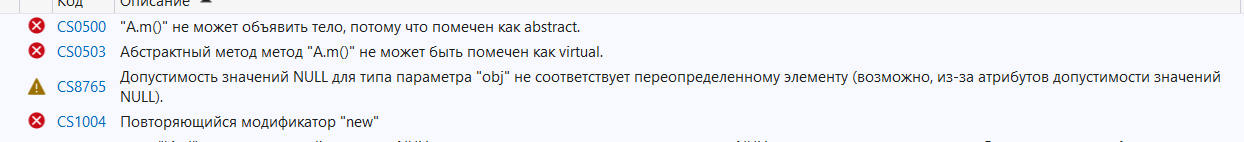
Интерфейс IComparer определен в пространстве имен System.Collections. Он содержит один метод CompareTo, возвращающий результат сравнения двух объектов, переданных ему в качестве параметров



28. второй







29. 3 4

30. A B