1. Чем класс отличается от структуры?

Класс:

- Ссылочный тип.

- Поддерживает наследование.

- Может иметь конструкторы без параметров, деструкторы, виртуальные и абстрактные методы.

- Имеет возможность реализовывать интерфейсы.

Структура:

- Значимый тип.

- Не поддерживает наследование (может реализовывать интерфейсы).

- Не может иметь деструкторы.

- Инициализируется автоматически при объявлении.

2. Что может и чего не может быть в структуре?

Что может быть в структуре:

- Поля.

- Методы.

- Свойства.

- Конструкторы (но только с параметрами).

Что не может быть в структуре:

- Деструкторы.

- Наследование (не может наследовать другие структуры или классы).

- Поля, которые могут быть `null` (например, ссылки на классы).?

3. Что такое перечисление? Приведите пример определения и использования перечисления.

Перечисление — набор логически связанных констант.

4. Перечислите и поясните стандартные интерфейсы .Net?

- IEnumerable: Позволяет перебирать коллекцию объектов.

- IComparer: Определяет метод для сравнения двух объектов.

- ICloneable: Позволяет создавать копии объектов.

- IEnumerator: Перебор по необобщенной коллекции.

- ICompable: сравнение объектов для выяснения порядка

- IEquatable<T>: Сравнение объектов на равенство.

- IStructuralEquatable: Проверка на равенство по значению.

- IDisposable: механизм для освобождения управляемых ресурсов.

7. Что такое полиморфизм? Перечислите его формы. Приведите примеры.

Полиморфизм - способность объекта изменять свое поведение в зависимости от контекста его использования. Основные формы полиморфизма:

- Полиморфизм времени компиляции (перегрузка методов и операторов).

- Полиморфизм времени выполнения (наследование и виртуальные методы).

8. Зачем в классе определяют виртуальные методы?

Виртуальные методы позволяют производным классам переопределять их реализацию. Это дает возможность создавать более гибкие и расширяемые приложения.

9. Как сделать запрет переопределения методов?

Для запрета переопределения метода используется ключевое слово `sealed` при определении метода в производном классе.