1. Чем класс отличается от структуры?

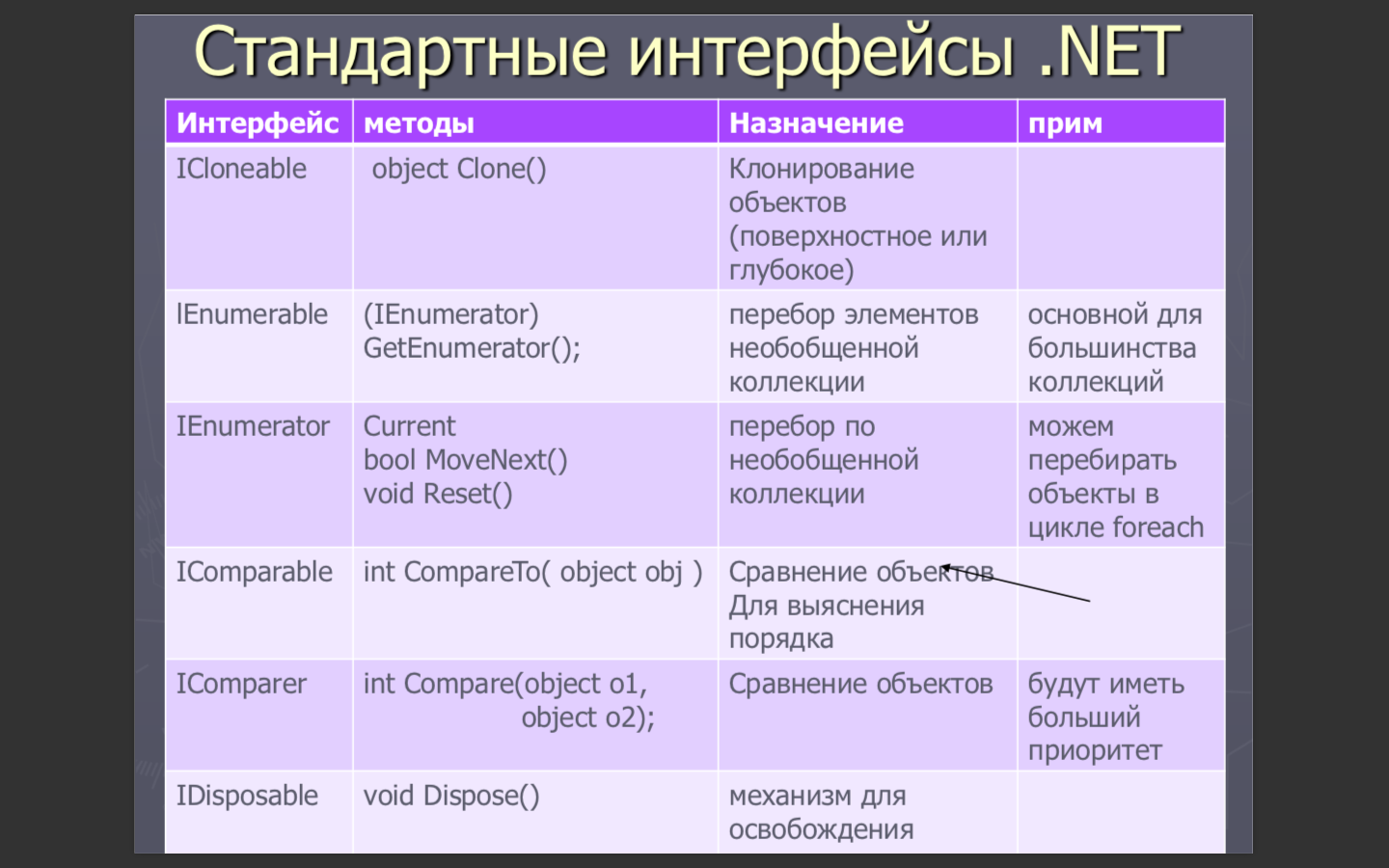
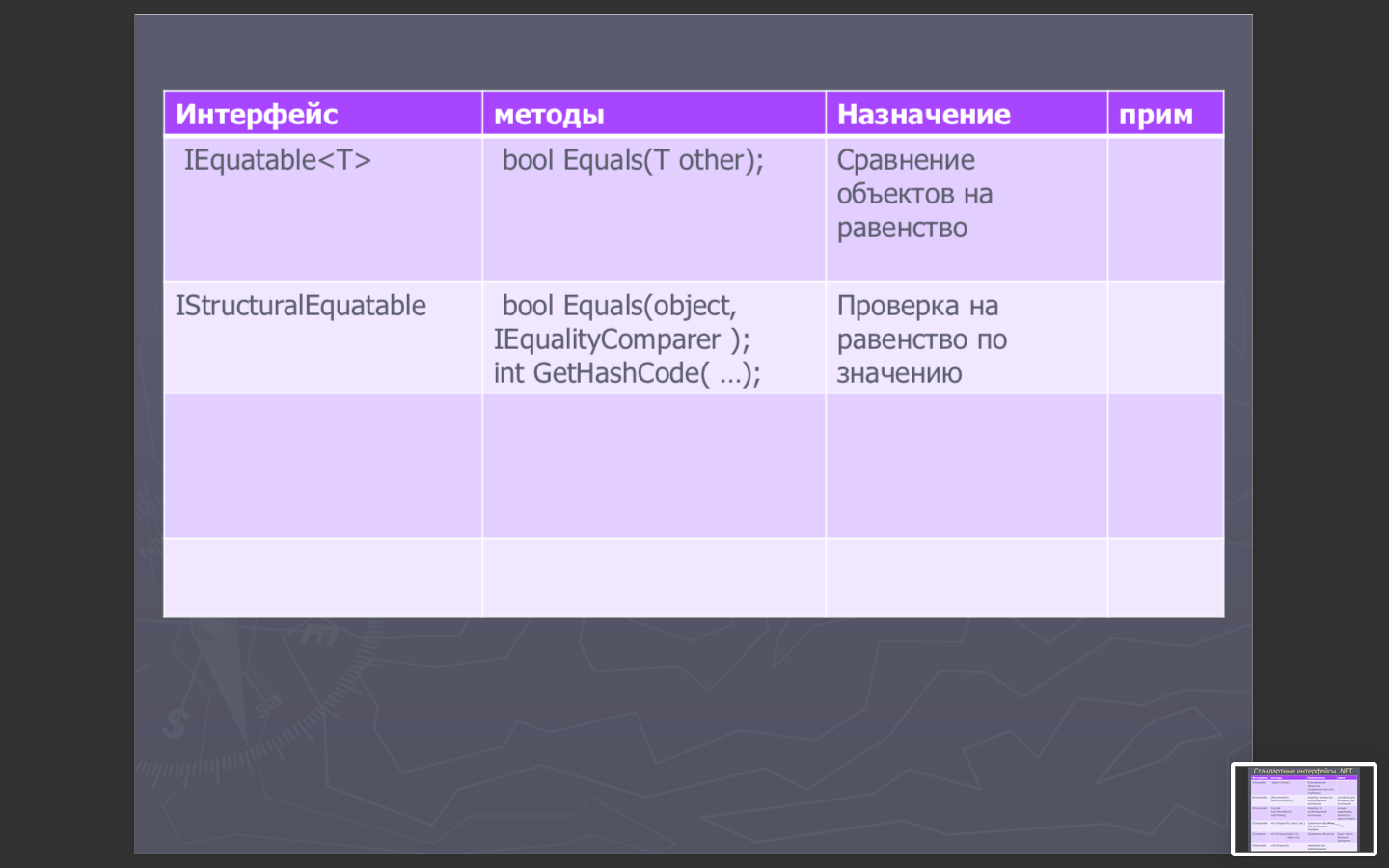
Т.е. структуры отличаются от классов тем, как они сохраняются в памяти и как к ним осуществляется доступ (классы — это ссылочные типы, размещаемые в куче, структуры — типы значений, размещаемые в [стеке](https://professorweb.ru/my/csharp/charp_theory/level12/12_6.php)), а также некоторыми свойствами (например, структуры не поддерживают наследование). Структуры не могут содержать явных конструкторов без параметров. инициализировать поля структуры напрямую при их объявлении,

1. Что может и чего не может быть в структуре?
2. Что такое перечисление? Приведите пример определения и использования  перечисления

Объявление перечисления происходит с помощью оператора **enum**. Далее идет название перечисления, после которого указывается тип перечисления - он обязательно должен представлять целочисленный тип (byte, int, short, long). Если тип явным образом не указан, то по умолчанию используется тип int. Затем идет список элементов перечисления через запятую:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | enum Days  {      Monday,      Tuesday,      Wednesday,      Thursday,      Friday,      Saturday,      Sunday  }    enum Time : byte  {      Morning,      Afternoon,      Evening,      Night  } |

1. Перечислите и поясните стандартные интерфейсы .Net



1. Как используется IComparable?

Он имеет всего один метод:

|  |  |
| --- | --- |
| 1  2  3  4 | public interface IComparable  {      int CompareTo(object o);  } |

Метод CompareTo предназначен для сравнения текущего объекта с объектом, который передается в качестве параметра object o. На выходе он возвращает целое число, которое может иметь одно из трех значений:

* Меньше нуля. Значит, текущий объект должен находиться перед объектом, который передается в качестве параметра
* Равен нулю. Значит, оба объекта равны
* Больше нуля. Значит, текущий объект должен находиться после объекта, передаваемого в качестве параметра

1. Как используется IClonable?

**Cloneable**:

|  |  |
| --- | --- |
| 1  2  3  4 | public interface ICloneable  {      object Clone();  } |

Реализация интерфейса в классе Person могла бы выглядеть следующим образом:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | class Person : ICloneable  {      public string Name { get; set; }      public int Age { get; set; }      public object Clone()      {          return new Person { Name = this.Name, Age = this.Age };      }  } |

Теперь все нормально копируется, изменения в свойствах p2 не сказываются на свойствах в p1.

1. Что такое полиморфизм? Перечислите его формы. Приведите примеры
2. Зачем в классе определяют виртуальные методы?

При наследовании нередко возникает необходимость при наследовании изменить в классе-наследнике функционал метода, который был унаследован от базового класса. В этом случае класс-наследник может переопределять методы и свойства базового класса.

Те методы и свойства, которые мы хотим сделать доступными для переопределения, в базовом классе помечается модификатором **virtual**. Такие методы и свойства называют виртуальными.

1. Как сделать запрет переопределения методов?

Также можно запретить переопределение методов и свойств. В этом случае их надо объявлять с модификатором **sealed**. При создании методов с модификатором sealed надо учитывать, что sealed применяется в паре с override, то есть только в переопределяемых методах.