

№ 3 Наследование, интерфейсы, абстрактные классы, виртуальные методы, перегрузка и исключения

Задание

- 1) Определить иерархию классов (в соответствии с вариантом) и реализовать классы. Можете расширить иерархию.
- 2) В проекте должны быть **интерфейс или/и абстрактный класс(ы)**.
- 3) Создайте один герметизированный (**бесплодный**) класс.
- 4) Каждый разрабатываемый класс (минимум один) должен содержать:
 - ✓ **переопределение метода** (override) WriteToFile () – метод записи значений полей класса в файл - в производных классах (наследуется от абстрактного класса или от интерфейса)
 - ✓ **переопределение метода** ToString() – реализовать на основе лямбда-выражения
 - ✓ в одном из классов переопределите все методы, унаследованные от Object.
 - ✓ в одном из методов класса верните кортеж и выполните распаковку кортежа.
- 5) При возникновении ошибок должны выбрасываться **исключения**. Использовать стандартные классы исключений. Выполните обработку **исключений**, используя try, catch и finally.
- 6) Для одного из классов **перегрузить** арифметические операции :+, - , *, / >, <, ==, != (например изменение размеров кнопки, рейтинга бойца, студента, потенциала техники и т.п.). Сделать проверку арифметических выражений на переполнение с использованием **checked**. Сравнение объектов выполнять на основе хэш-кода.
- 7) Один из классов необходимо сделать по паттерну **Singleton**.
- 8) Один из классов сделайте **partial** и разместите его в разных файлах.
- 9) Создайте класс Memento (реализует шаблон поведения **Memento**), который хранит информацию о состоянии о объекта выбранного вами класса (например: кнопки, бойца и т.п.): состояние и два метода set и get. Добавьте в сохраняемый класс методы (создает Memento и сохраняет состояние объекта) и новый класс Restorer (восстанавливает сохраненное состояние). В мейне продемонстрировать работу.
- 10) Написать демонстрационную программу, в которой создаются объекты различных классов, выполняются арифметические операции над ними, выполняются исключения. Поработать с объектами **через ссылки на абстрактные классы и интерфейсы**. В этом случае для идентификации типов объектов использовать **операторы is (с шаблоном и без) или as**.

Используйте документацию <http://msdn.microsoft.com/ru-ru/library/67ef8sbd.aspx>

Далее приведен перечень классов:

Вариант	Задание
1	Классы – фигура(абстрактный), прямоугольник, действия (интерфейс с методами click, move), кнопка (бесп);
2	Классы – фигура(абстрактный), круг, действия (интерфейс с методами checked, unchecked) Cradiobutton (бесп);
3	Классы – фигура, прямоугольник, управление (интерфейс с методами show, input) Ctextbox (бесп);
4	Классы – служащий (абстрактный), обучающийся (интерфейс), токарь (бесп), студент-заочник;
5	Классы – боец (абстрактный), лучник, действия (интерфейс), друид (бесп) и т.д.;
6	Классы – транспортное средство (абстрактный), машина, действия (интерфейс), человек, трансформер (бесп);
7	Классы – боец (абстрактный), охотник, действия (интерфейс), шаман (бесп);
8	Классы – товар (абстрактный), действия (интерфейс), сканер , принтер, CMFP (бесп. и сканирует, и печатает)
9	Классы – техника (абстрактный), самолет, действия (интерфейс) танк (бесп);
10	Классы – Icheckbutton (интерфейс с методами checked, unchecked), список, список строк, CCheckList (бесп);
11	Классы – ПО (абстракт), текстовый процессор, CWord (бесп.), действия (интерфейс), CConficker;
12	Классы – язык программирования (абстр.), операции (интерфейс), C, C++, JAVA, Scala
13	Классы – фигура, прямоугольник, управление (интерфейс с методами move, close, open) Cwindow, CMessageBox (бесп);
14	Классы – фигура (абстр), прямоугольник, элемент управления (интерфейс с методами close, resize), CBitmap, CPictureBox (бесп);
15	Классы – фигура (абстр), круг, IResize (интерфейс с методами smaller, bigger), лупа (бесп);

Вопросы

1. Кому доступны переменные с модификатором `protected`?
2. Наследуются ли переменные с модификатором `private`?
3. Какие методы наследуются от `Object`, как они используются?
4. `As, is` – что это, как применяется?
5. Поддерживает ли `C#` множественное наследование?
6. Можно ли запретить наследование от класса?
7. Можно ли разрешить наследование класса, но запретить перекрытие метода?
8. Что такое абстрактный класс?
9. В каком случае вы обязаны объявить класс абстрактным?
10. В чем разница между абстрактными и виртуальными классами? Между виртуальными и абстрактными методами?
11. Какие компоненты класса могут быть виртуальными?
12. Что такое интерфейс?
13. Как работать с объектом через унаследованный интерфейс?
14. Почему нельзя указать модификатор видимости для методов интерфейса?
15. Можно ли наследовать от нескольких интерфейсов?
16. Назовите отличия между интерфейсом и абстрактным классом.
17. Приведите пример явной реализации интерфейса.
18. Перечислите все правила перегрузки операторов в `C#`.
19. Приведите пример оператора приведения типа.
20. Для чего используются слова `checked` и `unchecked`?
21. Какой синтаксис нужно использовать в `C#` для отлова любого возможного исключения?
22. Будет ли выполнен блок `finally`, если не было сгенерировано исключение?
23. Можно ли выполнить несколько блоков `catch` для одного оператора `try`?
24. Назовите стандартные классы исключений и область их использования в `.NET`.
25. `Dispose()`, `Finalize()` – что это за методы, как используются в `.NET`?
26. Для чего в `.NET` используется конструкция `using(...){...}`? Причем тут `IDisposable`?
27. В чем отличие структуры и класса?
28. Приведите пример определения и использования перечисления.