

Лабораторная работа № 3

Наследование и полиморфизм

Время выдачи работы: 6-я неделя.

Срок защиты работы: 9-я неделя (до 6.04.2013).

1. ЦЕЛИ РАБОТЫ

Приобрести навыки использования механизмов наследования и полиморфизма при реализации объектно-ориентированной модели.

2. ВОПРОСЫ ДЛЯ ИЗУЧЕНИЯ

Принципы SOLID [3, гл. 8-12].

Наследование в C# и .NET Framework 4.5. Виртуальные и абстрактные методы. Переопределение виртуальных методов [2, гл. 3, раздел «Inheritance»)].

Класс `System.Object`. Методы, наследуемые от `System.Object`. [1], [2, гл. 3 «Inheritance» и «The object Type»].

Интерфейсы в C# и .NET Framework 4.5: определение интерфейса, наследование интерфейса, реализация интерфейсов в классе [2, гл. 3 «Interfaces»].

Механизмы форматирования и преобразования объектов в строку [2, гл. 6 «Formatting and Parsing»].

Механизмы сравнения и упорядочения объектов [2, гл. 6 «Equality Comparison», «Order Comparison»].

3. ЗАДАНИЕ

3.1. Варианты №№ 1, 6. «Служба доставки»

В качестве маркетингового хода менеджер службы доставки решил расширить систему скидок, предоставляемых клиентам компании. Теперь доступны следующие скидки:

- 5 %, если стоимость заказа превышает 1 000;
- 10 %, если стоимость заказа превышает 1 500;
- 7 %, если клиент – «привилегированный»;

- 15 %, если стоимость заказа превышает 1 500, и клиент – «привилегированный»;
- 12 %, если заказ был сформирован в период с 25 декабря по 7 января;
- 4 % на доставку товара с артикулом «UT-75X».

Необходимо модифицировать модель, реализованную в лабораторной работе № 2, так, чтобы полученное решение позволяло:

- определять, какие из доступных скидок могут быть применены к заказу;
- указывать скидку, предоставленную клиенту в рамках обработки заказа;
- определять общую стоимость заказа с учетом предоставленной скидки.

При этом решение должно обеспечивать гибкость для системы скидок, т. е. добавление новых видов скидок и удаление устаревших не должно приводить к переписыванию существующего кода обработки заказов.

В консольном приложении требуется продемонстрировать использование полученной модели.

Общий порядок работы с приложением остается, как и был реализован в лабораторной работе № 2. В конце формирования заказа приложение должно выводить список применимых к данному заказу скидок и запрашивать у менеджера, какая из этих скидок должна быть предоставлена. После окончания формирования заказа приложение должно выводить его общую стоимость и стоимость с учетом скидки (если она предоставляется).

3.2. Варианты №№ 2, 7. «Управление задачами»

Компания, использующая приложение «Управление задачами», выполняет проекты по заказу других компаний. Разные контракты предполагают разные условия выплат неустойки, связанной с превышением сроков выполнения задач, причем для разных задач даже в рамках одного проекта могут действовать разные правила.

На настоящий момент встречались следующие условия:

- за каждый день просрочки начисляется пеня 1% от стоимости задачи, но не более 25%;
- за каждый день просрочки начисляется пеня 2% от стоимости задачи, но не более 40%;

- пеня начисляется фиксированного размера, вне зависимости от того, насколько дней превышен срок;
- за каждые 2 дня просрочки начисляется пеня 2% от стоимости задачи;
- за каждый день просрочки начисляется пеня фиксированного размера, причем, при превышении срока 3 дня и менее неустойка не выплачивается (*например*, если фиксированная ставка равна 1 000 руб., то за превышение в 1-3 дня пеня не начисляется, за превышение в 4 дня начисляется пеня 4 000 руб. и т. д.).

Необходимо модифицировать модель, реализованную в лабораторной работе № 2, так, чтобы полученное решение позволяло:

- указывать правила начисления пени для задачи;
- определять общую стоимость задачи с учетом неустойки.

При этом решение должно обеспечивать гибкость системы правил начисления пени, т. е. добавление новых правил и удаление устаревших не должно приводить к переписыванию существующего кода работы с проектами.

В консольном приложении требуется продемонстрировать использование полученной модели.

Общий порядок работы с приложением остается, как и был реализован в лабораторной работе № 2. При вводе атрибутов каждой задачи приложение должно выводить список действующих правил начисления пени и запрашивать у менеджера, какое из них должно действовать для задачи. После окончания формирования проекта приложение должно выводить его общую стоимость.

3.3. Варианты №№ 3, 8. «Учебная программа»

В Университете для получения разных квалификационных степеней разные требования:

- за все время обучения студент должен прослушать определенное количество спецкурсов, а так же набрать в сумме некоторое количество кредитных единиц;
- студент должен прослушать заданное количество спецкурсов, причем в это число должны входить указанные обязательные спецкурсы (задан списан регистрационных номеров);

- студент в сумме должен набрать определенное количество кредитных единиц, а так же прослушать указанные курсы (задан списан регистрационных номеров);
- учебная программа должна быть составлена в 2013 году или ранее; за все время обучения студент должен набрать в сумме некоторое количество кредитных единиц;
- за все время обучения студент должен набрать в сумме некоторое количество кредитных единиц; все прослушанные курсы должны завершаться экзаменом;
- учебная программа должна быть составлена в 2014 году или ранее; студент должен прослушать определенное количество спецкурсов, отработать не менее заданного количества часов практических занятий и выполнить не менее заданного количества курсовых работ.

Необходимо модифицировать модель, реализованную в лабораторной работе № 2, так, чтобы полученное решение позволяло:

- указывать правила получения квалификационной степени;
- определять допустимость составленной программы по правилам, установленным для квалификационной степени.

При этом решение должно быть таким, что добавление новых правил получения степени и удаление устаревших не должно приводить к переписыванию существующего кода обработки учебных программ.

В консольном приложении требуется продемонстрировать использование полученной модели.

Общий порядок работы с приложением остается, как и был реализован в лабораторной работе № 2. Необходимо расширить набор предустановленных квалификационных степеней до 15-20, при этом должны быть задействованы разные правила получения степени.

3.4. Варианты №№ 4, 9. «Начисление зарплаты»

Компания начисляет заработную плату своим сотрудникам по различным правилам (во всех схемах оплаты должен учитываться подоходный налог):

- штатные сотрудники получают заработную плату по правилам, описанным в лабораторной работе № 2;
- некоторые штатные сотрудники получают базовую заработную плату с учетом переработок и работы в выходные, а так же за заключенные контракты 5% комиссионных от суммы контрактов;
- заработная плата начисляется за фактически отработанное время (вне зависимости от того, превышена ли норма рабочего времени в день, и была ли работа в выходные дни) с учетом должности и разряда сотрудника; профсоюзные отчисления составляют 1,5%;
- заработная плата начисляется за фактически отработанное время с учетом только должности сотрудника; профсоюзные отчисления составляют 0,5%;
- заработная плата начисляется за фактически отработанное время с учетом только должности сотрудника, за заключенные контракты 7% комиссионных от суммы контрактов; профсоюзные отчисления составляют 1%;
- сотрудник получает только комиссионные в размере 10% от суммы заключенных контрактов; профсоюзные отчисления не производятся.

Необходимо модифицировать модель, реализованную в лабораторной работе № 2, так, чтобы полученное решение позволяло:

- указывать контракты, заключенные за период, учитываемый в таблице (интерес представляют следующие атрибуты сущности «Контракт»: номер, дата, клиент, сумма, комментарий);
- указывать схему начисления заработной платы для каждого сотрудника;
- определять величину зарплаты за рабочее время, указанное в таблице.

При этом решение должно быть таким, что добавление новых схем начисления заработной платы не должно приводить к переписыванию существующего кода.

В консольном приложении требуется продемонстрировать использование полученной модели.

Общий порядок работы с приложением остается, как и был реализован в лабораторной работе № 2. При вводе данных о сотруднике, приложение также должно позволять выбрать схему начисления зарплаты.

3.5. Варианты №№ 5, 10. «Качество работы»

В целях обеспечения более детального мониторинга качества работы смен рабочих, предприятие начинает переработку системы присвоения категорий качества. Измененный вариант выглядит так:

- **1-я категория:** не более 3% брака; 80% изделий изготовлены за установленное нормативом время; для остальных изделий среднее время переработки не превышает 5 минут (3 минуты для изделий с артикулом «SCOF-B1», 4 минуты для изделий «SCOY-ME2»);
- **2-я категория:** не более 5% брака; 75% изделий изготовлены за установленное нормативом время; для остальных изделий среднее время переработки не превышает 5 минут (4 минуты для изделий «WVP-B2»);
- **3-я категория:** не более 6% брака; 75% изделий изготовлены за установленное нормативом время; для остальных изделий среднее время переработки не превышает 6 минут (4 минуты для изделий «WVP-B1»);
- **4-я категория:** не более 8% брака; 60% изделий изготовлены за установленное нормативом время; для остальных изделий среднее время переработки не превышает 6 минут (все изделия «WVTGN-MRL» изготовлены за нормативное время);
- **5-я категория:** в остальных случаях.

Необходимо модифицировать модель, реализованную в лабораторной работе № 2, так, чтобы полученное решение учитывало новые правила присвоения категорий качества партии. При этом решение должно быть таким, что добавление новых схем категоризации партий (а это будет происходить часто) не должно приводить к переписыванию существующего кода.

В консольном приложении требуется продемонстрировать использование полученной модели. Общий порядок работы с приложением остается, как и был реализован в лабораторной работе № 2. Необходимо расширить набор предустановленных номенклатур изделий до 10-13.

4. ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Что такое наследование? Для чего наследование может использоваться?

2. Что такое полиморфизм? Для чего полиморфизм может использоваться?
3. Сформулируйте принцип открытости/закрытости.
4. Сформулируйте принцип подстановки Лискоу.
5. Опишите синтаксис наследования в C#.
6. Назовите основные особенности виртуальных методов. Для чего используются виртуальные методы.
7. Опишите синтаксис определения виртуальных методов и их переопределения в производных классах на языке C#.
8. Что такое абстрактный класс и абстрактный метод? В каких ситуациях применяются абстрактные классы, абстрактные методы?
9. Перечислите методы, наследуемые всеми классами от класса `System.Object`.
10. Каково назначение метода `Object.Equals`? Назовите, в каких случаях требуется его переопределение. Приведите общую схему его переопределения.
11. Каково назначение метода `Object.GetHashCode`? Назовите, в каких случаях требуется его переопределение. Приведите общую схему его переопределения.
12. Каково назначение метода `Object.ToString`? Назовите, в каких случаях требуется его переопределение.
13. Что такое интерфейс? Для чего используются интерфейсы?
14. Как объявить и реализовать интерфейс на языке C#?
- 15.* Что такое явная реализация интерфейса (Explicit Interface Method Implementation, EIMI) в C#? Для чего этот механизм может использоваться?
16. Опишите стандартные механизмы BCL, предназначенные для получения строкового представления объекта.
17. Опишите стандартные механизмы BCL, предназначенные для определения равенства объектов.
18. Опишите стандартные механизмы BCL, предназначенные для задания отношения порядка на объектах.

5. СПИСОК ОСНОВНОЙ ЛИТЕРАТУРЫ

1. .NET Framework 4.5. — Библиотека MSDN. — <http://msdn.microsoft.com/ru-ru/library/w0x726c2.aspx>
2. Joseph Albahari, Ben Albahari. C# 5.0 in a Nutshell. The Defenitive Reference, 5th Edition. — O'Reilly, 2012
3. Мартин Р., Мартин М. Принципы, паттерны и методики гибкой разработки на языке C#. — М.: Символ-Плюс, 2011