

Практическое занятие № 8

Тема практического занятия: Полиморфизм, его реализация.

Цель практического занятия: получить понимание концепции полиморфизма и узнать, как он используется в объектно-ориентированном программировании.

В результате выполнения данной работы обучающийся должен уметь:

1. Применять наследование и переопределение методов для реализации полиморфизма.
2. Оптимизировать код с использованием полиморфизма, используя виртуальные методы (функции).
3. Анализировать преимущества и ограничения полиморфизма и применять его в соответствии с требованиями конкретной задачи.
4. Разрабатывать код с применением полиморфизма в различных ситуациях.

знать:

1. Концепцию полиморфизма и его роль в объектно-ориентированном программировании.
2. Как полиморфизм может быть использован для создания гибкого и расширяемого кода.

Перечень оборудования, необходимого для выполнения задания:

- Автоматизированные рабочие места по количеству обучающихся (процессор Intel Core i7 или аналогичный, БП 700 Вт, 32 Гб ОЗУ, SSD 512 Гб, HDD 4 ТБ SATA 7200 rpm, RTX 3060 12GB);
 - Автоматизированное рабочее место преподавателя (процессор Intel Core i7 или аналогичный, БП 700 Вт, 32 Гб ОЗУ, SSD 512 Гб, HDD 4 ТБ SATA 7200 rpm, RTX 3060 12GB);
 - Монитор 34", изогнутый, 3440x1440, 6 ms, 178°/178°, 300 cd/m², 20M:1, HDMI, DP, USB, регулировка по высоте (1 монитор на одно рабочее место).
 - Интерактивная панель 86" с OPS ПК
- Программное обеспечение:
- Microsoft Visual Studio.

Общие теоретические сведения:

Полиморфизм - это концепция или принцип объектно-ориентированного программирования, который позволяет объектам разного типа обрабатываться таким образом, как будто они являются объектами одного общего типа. Это означает, что одна и та же операция может иметь различную реализацию для разных типов объектов, исходя из их конкретных свойств и методов.

Полиморфизм позволяет программисту использовать абстрактные или общие типы данных для создания более гибкого и расширяемого кода. С

помощью полиморфизма можно работать с различными объектами через общий интерфейс, не завися от их конкретного типа.

Одним из наиболее распространенных примеров полиморфизма является возможность вызывать один и тот же метод для разных объектов, и получать различные результаты в зависимости от типа объекта. Это достигается с помощью наследования и переопределения методов в классах-наследниках.

Полиморфизм является одним из важных принципов объектно-ориентированного программирования и позволяет создавать более гибкие, расширяемые и поддерживаемые приложения.

Для реализации полиморфизма используются виртуальные методы и переопределение (перекрытие) методов.

Виртуальные методы - это методы, объявленные в базовом классе и предназначенные для переопределения в производных классах.

Виртуальные методы позволяют достичь полиморфизма, то есть обеспечить различное поведение для объектов разных классов, даже если они используют указатель или ссылку на базовый класс.

Когда виртуальный метод вызывается через указатель или ссылку на базовый класс, вызывается метод, определенный для конкретного класса объекта, а не метод базового класса.

Виртуальные методы позволяют достичь динамического полиморфизма, то есть возможности вызвать правильный метод во время выполнения, исходя из типа объекта, а не типа указателя или ссылки.

Задание:

1. Доработайте библиотеку классов, разработанную во время практического занятия №7, добавив в каждый класс метод, выдающий информацию об городе.
2. Заголовок метода в каждом классе должен быть одинаковым, но реализация – разной. В базовом классе необходимо выдать название города, его площадь и число человек на 1 км². В классе «Столица» необходимо выдавать название города, адрес правительства и число человек с учетом приезжих. В классе «Портовый город» необходимо выдавать название города и его площадь (за вычетом площади порта).
3. Разработайте приложение, которое позволит вводить информацию о произвольном числе городов разных видов, добавлять их в общий список, а затем выдать информацию из общего списка со сведениями о городах.

Указания по технике безопасности:

Инструкция по технике безопасности при работе в лаборатории, оборудованной компьютерной техникой.

Технология выполнения работы (этапы, последовательность действий):

1. Доработайте библиотеку классов в соответствии с заданием. При добавлении новых методов обеспечьте возможность переопределения методов: правильно выберите, какие из методов должны быть виртуальными (переопределяемыми), а какие – переопределяющими.

Пример виртуального метода:

```
/// <summary>
/// Информация о комнате
/// </summary>
/// <returns>Возвращает строку</returns>
Ссылка: 0
public virtual string Info()
{
    return "Комната площадью " + RoomArea() + " кв.м";
}
```

Примеры переопределяющих методов:

```
/// <summary>
/// Метод возвращает информацию о комнате
/// </summary>
/// <returns>Возвращается строка с информацией</returns>
ссылка: 1
public override string Info()
{
    return "Жилая комната площадью " + RoomArea() + " кв.м, с " + numWin + " окнами";
}
```

```
/// <summary>
/// Метод возвращает информацию об офисе
/// </summary>
/// <returns>Возвращается строка с информацией</returns>
ссылка: 1
public override string Info()
{
    return "Офис на " + NumWorkplaces() + " рабочих мест";
}
```

2. Пересоберите библиотеку, убедитесь в успешности сборки. Чтобы пересобрать библиотеку, выберите пункт меню «Сборка» и строку «Пересобрать решение».
3. Создайте проект для приложения. Можно за основу взять приложение, разработанное во время практического занятия №7.
4. Разработайте интерфейс приложения так, чтобы он позволял вводить данные о трех типов городов и добавлять их в список, а также выводить общую информацию из списка. Пример интерфейса для приложения с другим классом:

Помещение	Жилое помещение	Офис	Список помещений
Длина: 5	Длина: 6	Длина: 11	Комната площадью 12 кв.м Жилая комната площадью 12 кв.м, с 2 окнами Офис на 5 рабочих мест Комната площадью 20 кв.м Жилая комната площадью 24 кв.м, с 3 окнами Офис на 24 рабочих мест Офис на 29 рабочих мест
Ширина: 4	Ширина: 4	Ширина: 12	
	Число окон: 3	Число розеток: 60	
Добавить	Добавить	Добавить	Выдать

5. Добавьте и реализуйте все необходимые обработчики событий.
6. Убедитесь в правильности работы приложения. Для этого запустите приложение с помощью пункта меню «Пуск» и подставляйте в качестве исходный различные данные. Убедитесь, что для объектов, принадлежащих разным классам, выполняются разные методы.

Требование к отчету:

1. Во всех трех классах присутствует метод, позволяющий выдавать информацию о городе. Заголовок метода (тип возвращаемого значения, название, параметры) всех трех методов совпадают.
2. Правильно определено, какие из методов будут виртуальными (переопределяемыми), а какие – переопределяющими.
3. В приложении реализована возможность ввода пользователем всех необходимых данных.
4. В приложении формируется список городов.
5. В приложении корректно реализована выдача содержимого списка: при формировании информационной строки вызываются нужные методы, выдается информация о всех ранее добавленных в список городах.

Контрольные вопросы:

1. Что такое полиморфизм?
2. Что такое виртуальные методы, как они обозначаются?
3. Что такое переопределяющие методы, как они обозначаются?
4. Как реализован полиморфизм в данном задании?
5. Каковы преимущества использования полиморфизма?

Основные и дополнительные источники, электронные ресурсы:

1. Подбельский, В. В. Язык C#. Базовый курс: учебное пособие / В. В. Подбельский. - 2-е изд., перераб. и доп. - Москва: Финансы и статистика, 2022. - 408 с. - ISBN 978-5-00184-079-4. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1913989>
2. Гуриков, С. Р. Основы алгоритмизации и программирования на Python: учебное пособие / С.Р. Гуриков. — Москва: ИНФРА-М, 2023. — 343 с. — (Среднее профессиональное образование). - ISBN 978-5-16-016906-4. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1927269>.