|  |  |  |
| --- | --- | --- |
| **Дата** | **Содержание выполняемой работы** | **Подпись**  **руководителя** |
| 23.03.2021 | **Практическая работа № 2**  Тема**:** Принципы объектно-ориентированного программирования.  Наименование работы**:** Принципы объектно-ориентированного программирования. Библиотека FCL.  Цель**:** Ознакомиться с принципами объектно-ориентированного программирования. Сформировать навыки создания программ с использованием классов библиотеки FCL.  Задание1.  Изучил предложенную теорию.  Все основанные на объектах языки (C#, Java, С++, Smalltalk, Visual Basic и т.п.) должны отвечать трем основным принципам объектно-ориентированного программирования (ООП), которые перечислены ниже:  1. Инкапсуляция  2. Наследование  3. Полиморфизм  Инкапсуляция — это механизм программирования, объединяющий вместе код и данные, которыми он манипулирует, исключая как вмешательство извне, так и неправильное использование данных. В объектно-ориентированном языке данные и код могут быть объединены в совершенно автономный черный ящик. Внутри такого ящика находятся все необходимые данные и код. Когда код и данные связываются вместе подобным образом, создается объект. Иными словами, объект — это элемент, поддерживающий инкапсуляцию. Т.е. инкапсуляция представляет собой способности языка скрывать излишние детали реализации от пользователя объекта.  Следующий принцип ООП — наследование — касается способности языка позволять строить новые определения классов на основе определений существующих классов. По сути, наследование позволяет расширять поведение базового (или родительского) класса, наследуя основную функциональность в производном подклассе (также именуемом дочерним классом). Т.е. наследование представляет собой процесс, в ходе которого один объект приобретает свойства другого объекта. Это очень важный процесс, поскольку он обеспечивает принцип иерархической классификации. Если вдуматься, то большая часть знаний поддается систематизации благодаря иерархической классификации по нисходящей.  Последний принцип ООП — полиморфизм. Он обозначает способность языка трактовать связанные объекты в сходной манере. В частности, этот принцип ООП позволяет базовому классу определять набор членов (формально называемый полиморфным интерфейсом), которые доступны всем наследникам. Полиморфный интерфейс класса конструируется с использованием любого количества виртуальных или абстрактных членов.  Библиотека базовых классов(FCL)  В состав .NET Framework входит библиотека базовых классов .NET - Framework Class Library (NFCL). Она содержит определения более чем 20000 классов и процедур и предоставляет разработчикам при создании приложения набор стандартных функций. Применение библиотеки базовых классов не зависит от среды разработки и языка программирования — ее функции одинаковы везде.  Библиотека базовых классов содержит следующие категории функций:  - представления базовых типов;  - представления информации о загруженных типах;  - обработки исключений;  - ввода/вывода; - управления потоками;  - проверки безопасности;  - доступа к данным;  - графические функции;  - функции для работы с XML и SOAP.  Задание 2.  Создал программы по образцу в соответствии с вариантом.  Задание 2.1.  Создал приложение для вычисления значения функции. Листинг кода задания 2.1 представлен в Приложении 1.  Результат выполнения программы представлен на Рисунке 1  Безымянный.png  Рисунок 1 – результат выполнения программы  Задание 2.2.  Создал приложение для вычисления арифметического выражения. Листинг кода задания 2.2 представлен в Приложении 1.  Результат выполнения программы представлен на Рисунке 2  **2.png**  Рисунок 2 – результат выполнения программы  Задание 2.3.  Создал приложение для вывода данных. Вводится с клавиатуры число от 0 до 23. Вывести наименование соответствующей части суток (ночь, утро, день, вечер). Листинг кода задания 2.3 представлен в Приложении 1.  Результат выполнения программы представлен на Рисунке 3  **4.png**  Рисунок 3 – результат выполнения программы  Задание 2.4.  Создал приложение для табуляции функции. Листинг кода задания 2.4 представлен в Приложении 1.  Результат выполнения программы представлен на Рисунке 4    Рисунок 4 – результат выполнения программы  Контрольные вопросы:   1. Поясните принцип инкапсуляции.   Ответ: представляет собой способности языка скрывать излишние детали реализации от пользователя объекта.   1. Поясните принцип полиморфизма.   Ответ: обозначает способность языка трактовать связанные объекты в сходной манере.   1. Поясните принцип наследования.   Ответ: представляет собой процесс, в ходе которого один объект приобретает свойства другого объекта.   1. Объясните, для чего служит библиотека базовых классов.   Ответ: Она содержит определения более чем 20000 классов и процедур и предоставляет разработчикам при создании приложения набор стандартных функций.   1. Поясните суть пространства имен.   Ответ: Языки с поддержкой пространств имён определяют [правила](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C), указывающие, к какому пространству имён принадлежит идентификатор   1. Укажите назначение библиотеки Math.   Ответ: этот класс содержит два статических поля, задающих константы E и PI, а также 23 статических метода |  |