

Лабораторная работа №23

Документирование кода

1 Цель работы

1.1 Изучить процесс документирования программного кода.

2 Литература

2.1 <https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/xml/doc/how-to-use-the-xml-documentation-features>

3 Подготовка к работе

3.1 Повторить теоретический материал (см. п.2).

3.2 Изучить описание лабораторной работы.

4 Основное оборудование

4.1 Персональный компьютер.

5 Задание

5.1 Написание самодокументируемого кода

5.1.1 Создать новый проект типа библиотека классов, назвать его LabWorkLibrary. Переименовать стандартный класс Class1 в Maths (статический класс, в котором будет храниться набор статических открытых методов и констант, использующихся в математических выражениях). Добавить в класс следующие элементы:

- метод для вычисления суммы двух чисел;
- метод для вычисления разности двух чисел;
- метод для вычисления произведения двух чисел;
- метод для вычисления частного двух чисел;
- метод для вычисления площади прямоугольника;
- константу BinaryFactor = 1024 (константа со значением 2^{10}).

Названия методов должны быть глаголами или начинаться с глаголов.

Пересобрать решение и проверить, что в папке bin\Debug\net появился файл библиотеки (.dll).

5.1.2 Создать новое решение с консольным проектом LabWork23 (здесь будет выполняться тестирование созданной библиотеки).

Подключить к проекту ссылку на созданную библиотеку, указав путь к ней в разделе «Обзор» в менеджере ссылок (меню Проект — Добавить ссылку на проект — Обзор... и выбрать требуемый проект с библиотекой).

В коде проекта подключить библиотеку, используя using ИмяБиблиотеки.

Использовать методы и константу библиотеки в консольном приложении LabWork23.

5.2 Документирование описания элементов классов

5.2.1 Добавить в код библиотеки XML-комментарии ко всем элементам класса Maths (для генерации шаблона комментария требуется нажать /// над элементами класса), заполнить в сгенерированном шаблоне тэг summary (указать, что делает метод, например: *выводит текст на экран в отформатированном виде*).

5.2.2 Для того, чтобы XML-комментарии отображались, требуется открыть свойства проекта библиотеки классов: «Свойства» – «Сборка» – «Выходные данные» поставить флажок «Файл документации». Изучить содержимое папки bin\Debug (должен появиться xml-файл с документацией).

5.2.3 Пересобрать проект с библиотекой и консольное решение для того, чтобы проверить, отображаются ли комментарии. Для просмотра всех комментариев в проекте, к которому подключена библиотека, использовать следующие способы:

- открыть ее в обозревателе объектов, дважды нажав на библиотеку в обозревателе решений,
- навести курсор на методы библиотеки в консольном приложении.

5.3 Документирование параметров методов классов

5.3.1 Заполнить в коде библиотеки XML-комментарий методов описанием того, что за параметры требуется передать пользователям (тэг param) и что возвращают методы (тэг returns).

Например: *параметры: делимое и делитель, результат: частное.*

5.3.2 Пересобрать проект с библиотекой и консольное решение для того, чтобы проверить изменения в отображении комментариев в консольном решении.

5.4 Документирование информации об исключениях

5.4.1 Добавить генерацию исключений в методы класса Maths:

- метод для вычисления частного двух чисел должен генерировать исключение при попытке деления на 0;
- метод для вычисления площади прямоугольника должен генерировать исключение при некорректных входных данных.

Для генерации исключения используется throw:

throw new ТипИсключения("Сообщение пользователю");

5.4.2 Добавить в методы со сгенерированными исключениями XML-комментарий с информацией об исключении:

///<exception cref="тип">причина</exception>

Вместо *тип* указать тип исключения (например: ArgumentException).

Вместо *причина* указать описание ситуации, в которой возникает исключение.

5.4.3 Пересобрать проект с библиотекой и консольное решение для того, чтобы проверить изменения в отображении комментариев в обозревателе решений.

5.5 Документирование пользовательских типов

5.5.1 Добавить в библиотеку классов в отдельных файлах:

- перечисление Role (Роль пользователя) со значениями Покупатель, Менеджер, Администратор,
- класс User (Пользователь) с открытыми свойствами логин, пароль и роль (типа перечисление Role).

5.5.2 Добавить в код библиотеки XML-комментарии ко всем пользовательским типам данных (классам Maths и User и перечислению Role). Для генерации шаблона комментария требуется нажать /// над пользовательскими типами данных), заполнить сгенерированный шаблон.

5.5.3 Пересобрать проект с библиотекой и консольное решение для того, чтобы проверить изменения в отображении комментариев в обозревателе решений.

6 Порядок выполнения работы

6.1 Запустить MS Visual Studio и создать приложение C#. Выполнить все задания из п.5 в одном решении LabWork23. При разработке считать, что пользователь ввел данные требуемого типа, остальные возможные ошибки обрабатывать.

При выполнении заданий использовать минимально возможное количество команд и переменных и выполнять форматирование и рефакторинг кода. Все XML-комментарии писать в том же стиле, что и стандартные XML-комментарии.

6.2 Ответить на контрольные вопросы.

7 Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

8 Контрольные вопросы

8.1 Что такое «XML-документация»?

8.2 Как сгенерировать XML-комментарий?

8.3 Какие действия нужно выполнить, чтобы XML-документация была видна при подключении библиотеки в стороннем решении?

8.4 Что пишется в разделе summary?

8.5 Что пишется в разделе param?

8.6 Что пишется в разделе returns?