Модульное тестирование проверки пароля

1. Задачи: использовать модульное тестирование для проверки пароля на соответствие заданным требованиям:

|  |  |  |
| --- | --- | --- |
| Требование | Пароль | Результат |
| Количество символов (от 8 до 20) | abcEDF01)!  Put | True  False |
| Встречаются цифры | 12345678aBc()  Odin | True  False |
| Встречаются спецсимволы (#$%^&\_) | Java  Donba1$$ | True  False |
| Встречаются прописные | Qwert4  FistingIs300$ | True  False |
| Встречаются строчные | asswecan  qazWER!@43 | True  False |

2. Цель: научиться работать с модульным тестированием.

3. Описание: на языке С# был написана программа по проверке пароля и произведен её тест, который находятся тут: https://github.com/Chezemy/Panyushkin\_Pavel\_195.git

4. Теория: для программы были созданы модульные тесты (нажатие правой кнопки мыши по методу или классу, создание модульных тестов).

Assert-коллекция вспомогательных классов для тестирования различных условий в модульных тестах. Если проверяемое значение ложно, создаётся исключение.

Assert.AreEqual(значение1, значение2) – Проверяет указанные значения на равенство и создаёт исключение, если два значения не равны.

Assert.IsFalse(значение) – Проверяет, является ли указанное условие ложным, и создаёт исключение, если условие истинно.

[TestClass()] – атрибут тестового класса.

[TestMethod()] – атрибут метода теста.

Метод validatePassword вызывается из метода Main с передачей пароля(строки) как аргумента. Пример:

validatePassword(qwerQW12!@);

Char.IsLower – проверка, является ли символ строчным;

Char.IsUpper – проверка, является ли символ прописным;

Char.IsDigit – проверка, является ли символ числом;

Intersect() – проверка на пересечение;

Чтобы создать тесты, необходимо нажать пкм по классу или методу и нажать “создание модульных тестов”

Чтобы посмотреть результаты тестов, необходимо нажать:

Тест » Обозреватель тестов;

5. Результат: https://github.com/Chezemy/Panyushkin\_Pavel\_195.git