Задание №2

public class PasswordStrengthChekerTests

{

[TestMethod]

public void GetPasswordStrength\_AllCahrs\_5Points()

{

string password = "P2ssw0rd#";

int expected = 5;

int actual = PasswordStrengthCheker.GetPasswordStrength(password);

Assert.AreEqual(expected, actual);

}

[TestMethod]

public void GetPasswordStrength\_less8lenght()

{

// arrange

string password = "P2ssw0r";

int expected = 3;

// act

int actual = PasswordStrengthCheker.GetPasswordStrength(password);

// assert

Assert.AreEqual(expected, actual);

}

[TestMethod]

public void GetPasswordStrength\_ConteinsNumber\_0\_4Points()

{

// Arrange

string password = "Passw0rd";

// верхний регистр 1, за длину строки 1, за нижний регистр 1

// число 1

int expected = 5;

// Act

int actual = PasswordStrengthCheker.GetPasswordStrength(password);

// Assert

Assert.AreEqual(expected, actual);

}

[TestMethod]

public void GetPasswordStrength\_ConteinsNumber\_1\_4Points()

{

// Arrange

string password = "Passw1rd";

// верхний регистр 1, за длину строки 1, за нижний регистр 1

// число 1

int expected = 5;

// Act

int actual = PasswordStrengthCheker.GetPasswordStrength(password);

// Assert

Assert.AreEqual(expected, actual);

}

// Tests for special chars

[TestMethod]

public void GetPasswordStrength\_ContainsSpecialChar\_at\_5Points()

{

// Arrange

string password = "Passw0rd@";

// верхний регистр 1, за длину строки 1, за нижний регистр 1

// число 1, специальный символ 1

int expected = 5;

// Act

int actual = PasswordStrengthCheker.GetPasswordStrength(password);

// Assert

Assert.AreEqual(expected, actual);

}

[TestMethod]

public void GetPasswordStrength\_ContainsSpecialChar\_Hash\_5Points()

{

string password = "Passw0rd#";

int expected = 5;

int actual = PasswordStrengthCheker.GetPasswordStrength(password);

Assert.AreEqual(expected, actual);

}

[TestMethod]

public void GetPasswordStrength\_ContainsSpecialChar\_Excl\_5Points()

{

string password = "Passw0rd!";

int expected = 5;

int actual = PasswordStrengthCheker.GetPasswordStrength(password);

Assert.AreEqual(expected, actual);

}

[TestMethod]

public void GetPasswordStrength\_ContainsSpecialChar\_Doll\_5Points()

{

string password = "Passw0rd$";

int expected = 5;

int actual = PasswordStrengthCheker.GetPasswordStrength(password);

Assert.AreEqual(expected, actual);

}

}

