

## **Практическая работа 6**

### **Составление тестовых заданий**

**Цель занятия:** Получить практический опыт создания тестов и управление ими

#### **Перечень оборудования и программного обеспечения**

Персональный компьютер  
Microsoft Office (Word, Visio)  
Microsoft Visual Studio 2015

#### **Краткие теоретические сведения**

Создание и управление тестами является основной частью процесса для группы тестирования, которая позволяет организовать их работу. В зависимости от целей тестирования существуют различные типы тестов, которые можно использовать — ручные тесты и автоматические тесты.

С помощью Visual Studio 2010 можно создавать тестовые случаи для ручных тестов, которые содержат как шаги действий, так и проверочные шаги теста. Для добавления тестовых случаев в план тестирования можно использовать Microsoft Test Manager.

Также можно создать автоматические тесты с помощью Visual Studio. Автоматические тесты позволяют более эффективно выполнить проверку приложения. Автоматические тесты выполняют шаги теста и определяют пройден тест или завершен с ошибкой. Поэтому повышается скорость и периодичность тестирования по сравнению с ручными тестами. Автоматические тесты позволяют быстро проверить правильность работы приложения после внесения изменений в его код.

Позднее можно преобразовать ручные тестовые случаи в автоматические тесты путем связывания автоматизации с тестовым случаем, который содержит шаги ручного теста. Затем можно выполнить эти автоматические тесты с помощью Microsoft Test Manager и просмотреть их результаты в ходе выполнения плана тестирования.

#### **Пример создания модульного теста**

В меню Файл выберите пункт Создать, а затем команду Проект. Отобразится диалоговое окно Новый проект. Создайте консольное приложение.

В консольное приложение после class Program добавьте код метода Circle:

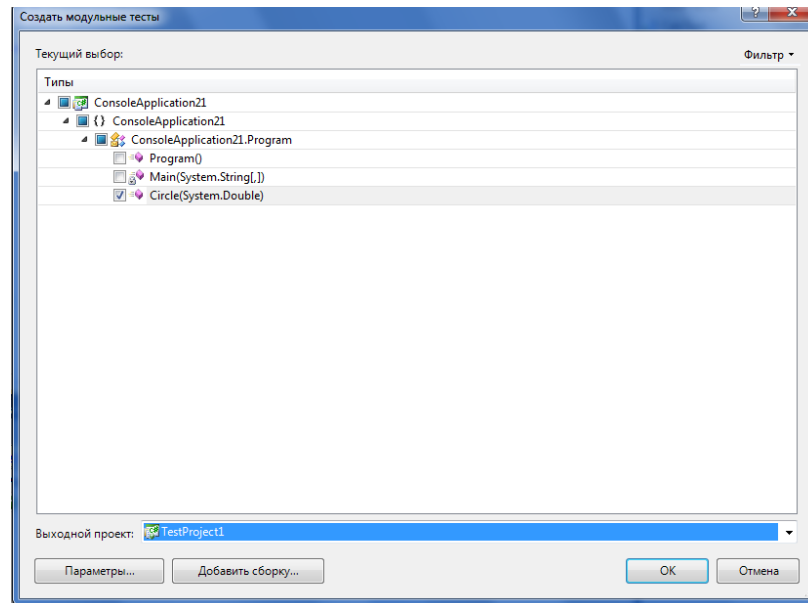
```
public static double Circle(double radius)
```

```

{
double dlna = Math.PI*Math.Pow(radius,2.0);
return dlna ;
}

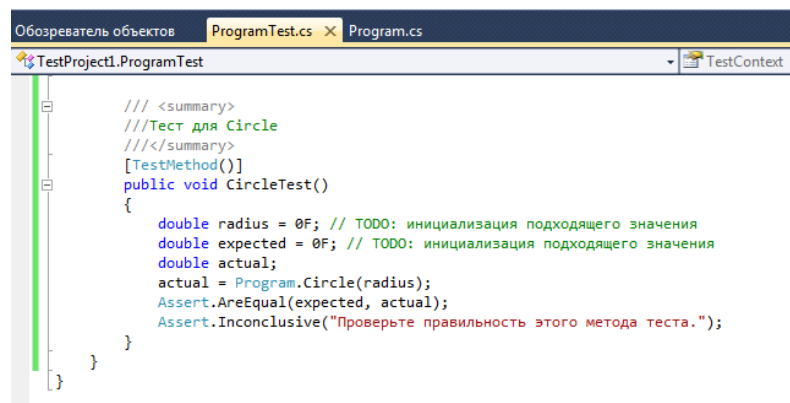
```

Нажмите правую кнопку мыши и выберите пункт меню Создать модульные тесты.



В окне должен появиться метод, для которого будет создан модульный тест.

На экране появится код тестового модуля TestProject1:



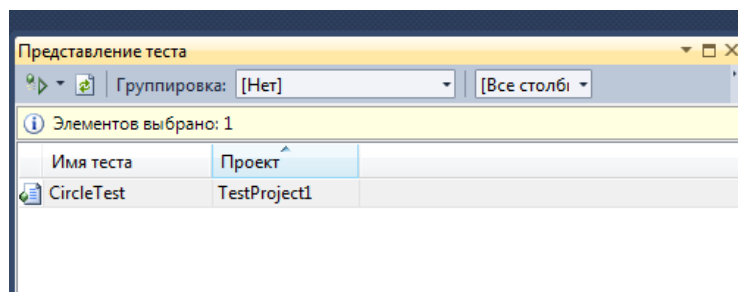
Добавьте в код значения radius и expected для проверки метода.

```

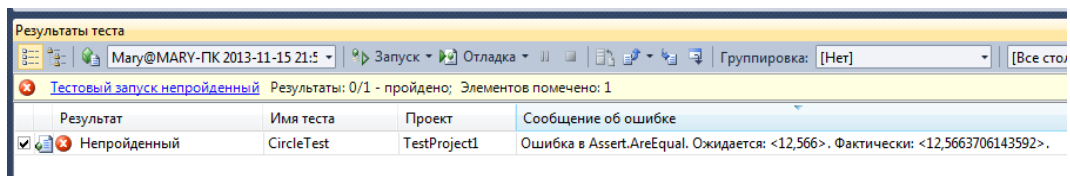
public void CircleTest()
{
    double radius = 2; // TODO: инициализация
    double expected = 12.566; // TODO: иници
    double actual;
    actual = Program.Circle(radius);
    Assert.AreEqual(expected, actual);
    //Assert.Inconclusive("Проверьте правиль

```

В меню Тест на вкладке Окна выберите Представление теста.

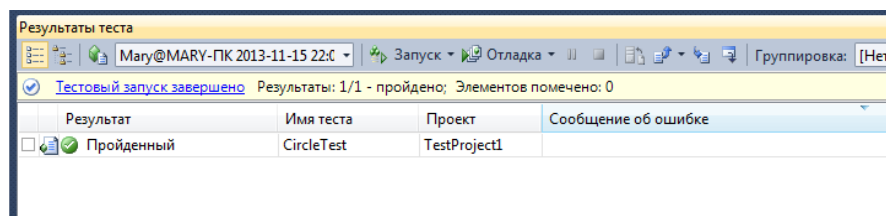


Нажмите кнопку  для запуска теста метода CircleTest.



Как видно из примера, тест не пройден, т.к. точность результата указана не верно.

Исправьте точность и нажмите кнопку обновить в окне Представление теста.



При необходимости проведите отладку теста.

## Задания

- 1 Изучить теоретические сведения и задание к работе
- 2 В соответствии с вариантом задания разработать отлаженный модуль.

## Порядок выполнения работы

### Содержание отчета

1. Название работы
2. Цель работы
3. Перечень оборудования и программного обеспечения
4. Порядок выполнения

5. Листинг отлаженного модуля в соответствии с вариантом задания.
6. Вывод.

### **Варианты заданий**

Варианты заданий представлены в практической работе 1.

### **Используемая литература**

1 Рудаков А.В. Технология разработки программных продуктов: учеб. пособие для студ. учреждений сред. проф. образования – 11-е изд., стер. – М.: Издательский центр «Академия», 2017. – 208 с.

2 Федорова Г.Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности: учеб. пособие для студ. учреждений сред. проф. образования / Г.Н. Федорова.– М.: КУРС : ИНФРА – М, 2017. – 334 с

3 Федорова Г.Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности: Учебное пособие для СПО.- М.: КУРС, 2018. – 333 с.– ЭБС Знаниум

4 <http://msdn.microsoft.com/ru-ru/library/67ef8sbd.aspx>