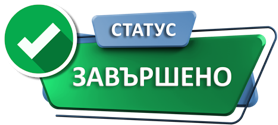
# Упражнения: Полиморфизъм

Можете да проверите решенията си в **Judge системата**: <https://judge.softuni.org/Contests/Practice/Index/4068>

Състояние на настоящия учебен материал:



## Математически операции

**Важно**: Трябва да имате публичен клас StartUp в namespace \_01\_MathOperations.

Създайте клас **MathOperations**, който има 3 пъти метода Add(). Add() трябва да има следните варианти:

* **Add(int, int): int**
* **Add(double, double, double): double**
* **Add(decimal, decimal, decimal): decimal**

Трябва да можете да използвате класа по следния начин:

Text

Description automatically generated

### Примери

|  |
| --- |
| **Изход** |
| 5  11  9.9 |

### Решение

Създайте клас **MathOperations**, който изглежда по следния начин:

Graphical user interface, text, application

Description automatically generated

## Животни

**Важно**: Трябва да имате **публичен** клас **StartUp** в namespace **\_02\_Animals**.

Създайте клас **Animal**, който има следните **полета** и респективните **свойства**:

* **name - string**
* **favouriteFood - string**

Animal има един виртуален метод ExplainSelf()**: string**, койтовръща информация в следния формат:

* **"I am {име на животното} and my favourite food is {любима храна}"**

Добавете още **два класа** - **Cat** and **Dog**, които наследяват **Animal. Презапишете** метода ExplainSelf(),като добавите конкретния звук на животното на нов ред:

* **MEEOW** за котка
* **DJAAF** за куче

Трябва да можете да използвате класа по следния начин:

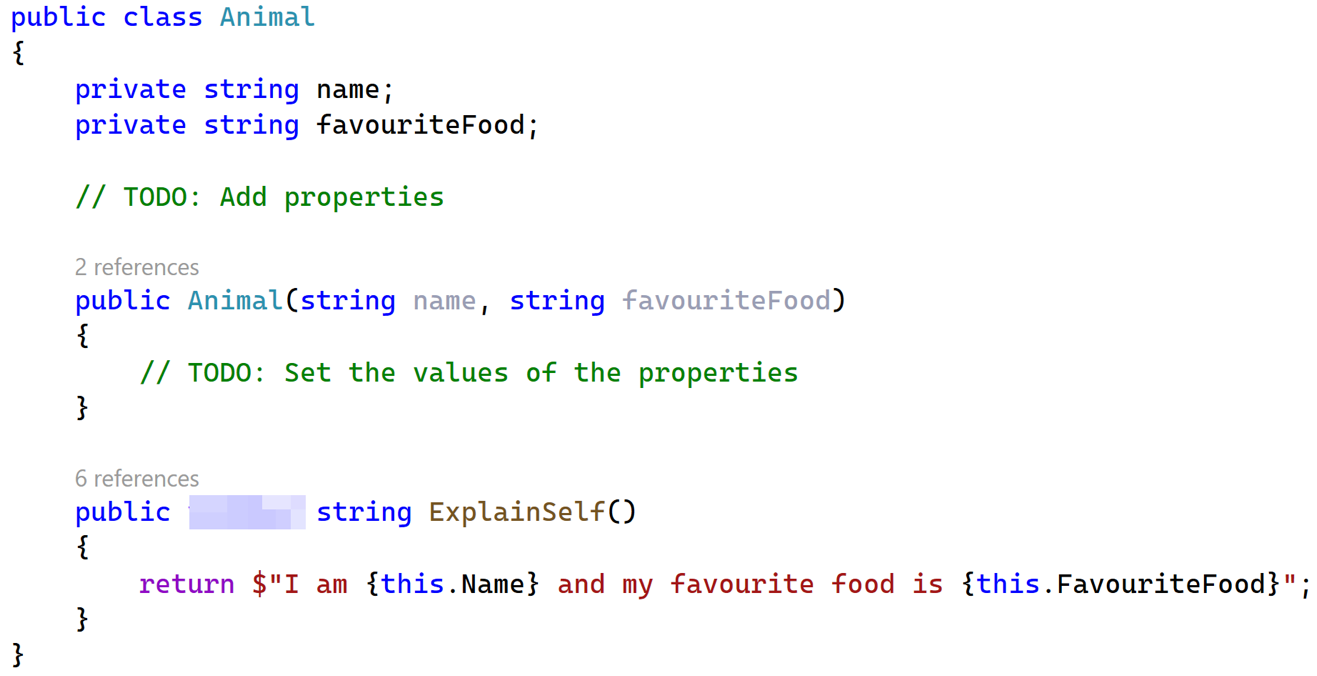
Text

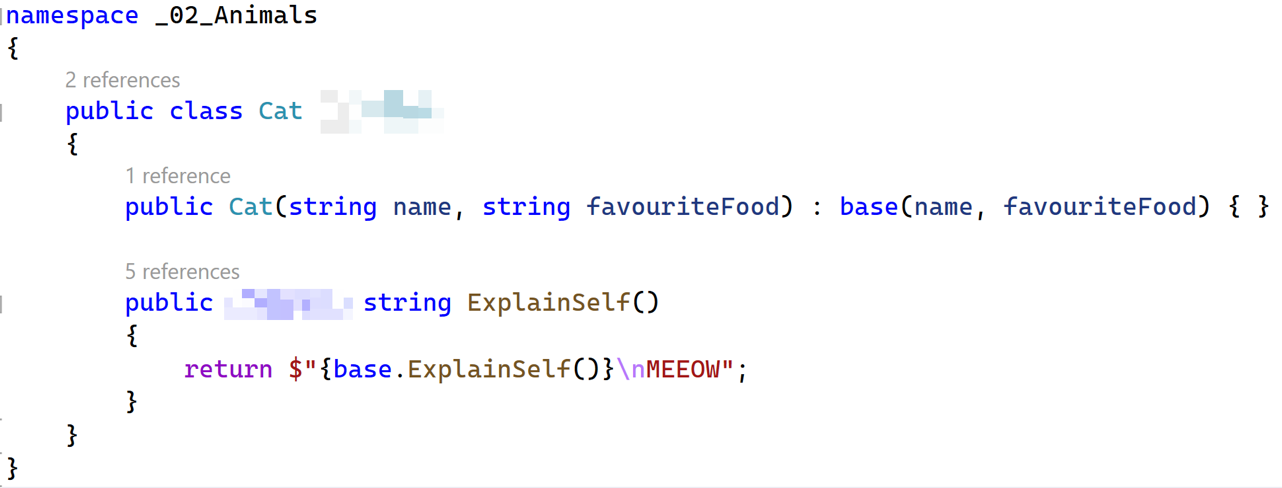
Description automatically generated

### Примери

|  |
| --- |
| **Изход** |
| I am Pesho and my favourite food is Whiskas  MEEOW  I am Gosho and my favourite food is Meat  DJAAF |

### Решение





## Фигури

**Важно**: Трябва да имате **публичен** клас **StartUp** в namespace **\_03\_Shapes**.

Създайте йерархия, започваща с **абстрактен** клас **Shape**:

* **Абстрактни методи:**
  + CalculatePerimeter(): double
  + CalculateArea(): double
* **Виртуални методи**:
  + Draw(): string
* Методът трябва да отпечатва текста "**Drawing**". Класовете-наследници ще допълнят съдържанието на резултата.

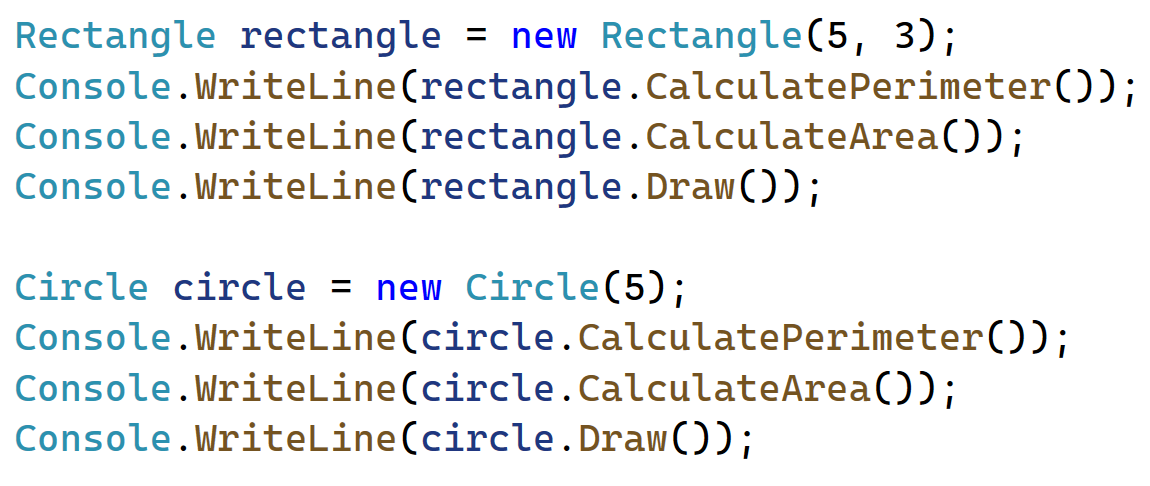
Разширете класа **Shape** с два дъщерни класа:

* **Rectangle**
* **Circle**

Всеки от тях трябва да има:

* **Полета:** 
  + **height** (височина)и **width** (ширина) за **Rectangle** (правоъгълник)
  + **radius** (радиус)за **Circle** (кръг)
* **Капсулация** за тези полета
* Публичен **конструктор**
* Методи за изчисляване на **периметър** и **лице**
* **Override** на метода Draw() за рисуване. Той трябва да **допълва** съобщението от **базовия клас**, като добави името не текущия клас (**Rectangle** или **Circle** респективно).

Трябва да можете да използвате класа по следния начин:



|  |
| --- |
| **Изход** |
| 16  15  Drawing Rectangle  31.41592653589793  78.53981633974483  Drawing Circle |