# Упражнения: Полиморфизъм

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

## Математически операции

**Важно**: Трябва да имате публичен клас StartUp в namespace Operations.

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 MathOperation, който изглежда по следния начин:

Graphical user interface, text, application

Description automatically generated

## Животни

**Важно**: Трябва да имате публичен клас **StartUp** в namespace Animals.

Създайте клас Animal, който има следните полета:

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

Animal има един виртуален метод:

* ExplainSelf()**: string**

Добавете още **два класа** - **Cat** and **Dog. Презапишете** метода ExplainSelf(),като добавите конкретния звук на животното на нов ред: “**MEEOW**” или “**DJAAF**”.

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 |

### Насоки за решение

Graphical user interface, text, application, email

Description automatically generated



## Фигури

**Важно**: Трябва да имате публичен клас StartUp в namespace Shapes.

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

* **Абстрактни методи:**
  + CalculatePerimeter(): doulbe
  + CalculateArea(): double
* **Виртуални методи**:
  + Draw(): string

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

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

Всеки от тези класове трябва да съдържа:

* **Полета:** 
  + **height** (височина)и **width** (ширина) за **Rectangle** (правоъгълник)
  + **radius** (радиус)за **Circle** (кръг)
* **Капсулация** за тези полета
* Публичен **конструктор**, който приема тези полета
* Методи за изчисляване на **периметър** и **лице** – имплементация от интерфейса
* **Override** на метода Draw() за рисуване:
  + **Rectangle**: отпечатва “Drawing Rectangle”
  + **Circle**: отпечатва “Drawing Circle”